Advanced Micro Devices (AMD) 22V10 Programmable Array Logic (PAL) Development Team Oral History Panel

Warren Miller, Mitch Richman, and Bill Sievers

Moderated and edited by:
David Laws

Recorded: July 19, 2012
Mountain View, California

CHM Reference number: X6559.2013

Laws: Today is Thursday, 19th of July 2012. We are here today at the Computer History Museum in Mountain View, California to record a video oral history with some of the key contributors to the 22V10 PAL device development project. The 22V10 was an important product developed by Advanced Micro Devices (AMD) in the middle 1980s. My name is David Laws. I was managing director of the group at the time that the product was introduced. I am looking forward to reliving the experiences of those years. They were challenging, exciting times at the company. AMD was founded in 1969 and today we will be talking about the 1984 timeframe, at which time AMD was about a quarter of a billion dollar company. It had just enjoyed a 60% growth in revenue over the prior year. The goal of the company was to achieve 900 million dollars in sales in the next fiscal year. So by that time, it was a substantial factor in the marketplace and was emerging from purely a second-source manufacturer of integrated circuits to a company dependent on its own proprietary devices, devices that it had developed itself. The 22V10 was one of those proprietary products. I'd like to start this morning with an introduction to the members of the panel. I will ask each of you to give a brief background on your early childhood, how you became interested in science and technology, and perhaps what you studied in college. Let's start with Bill Sievers.

Sievers: I grew up in the Bay area in Sausalito and went to junior college at the College of Marin. I then went to UC Berkeley, got a degree in electrical engineering in 1962, and took my first job with RCA in New Jersey. After less than a year there, they notified me that all the recruits were going to get laid off because the union said we had to go before anybody else. They got excited about that and decided they'd fly us out to non-union plants for RCA. So, I ended up in Florida at the RCA Palm Beach Gardens plant, which was working on the Spectra 70 computer. This was very fortunate because I got to design digital circuits. But at the time they weren't integrated circuits, they were designed on printed circuit boards with resistors, transistors, and diodes. After I'd done that for about three years, the Fairchild salesman who was selling us silicon transistors came in and said "Here's a little TO-5 can. It's got two gates in it." I quickly saw that this was a major change in technology, and asked him for an application. , —I came to an interview with Henry Blume at Fairchild. They hired me. I drove back to California in my VW with the family and started at Fairchild in 1965.

Laws: '65, those were exciting times at Fairchild.

Sievers: Yes, they were. I think the first product I worked on at Fairchild was Low Power DTL. H.T. Chua, I think, was helping me with learning the design process and how to design integrated circuits. But Henry Blume was my major mentor.

Laws: Was there anything in your early school years or was there a mentor or someone that turned you on to science and engineering?

Sievers: Well my father was an engineer. But he was a power engineer. He worked for Bechtel. And things like going to Chicago and visiting the Museum of Science and Industry really got me excited. I was always good at mathematics, so that helps you stay on the path for an engineering degree.

Laws: That's good PR. We should use this video for promoting the museum.

Sievers: I guess so, yeah.

Laws: Warren, what's your background. How did you end up in Silicon Valley?

Miller: I grew up in Northern California. My father worked for the railroad. So he was an engineer, but a different kind of engineer. He did civil engineering, building bridges, and helping do the surveying and stuff like that. And so, I guess naturally I got interested in math and science. The space program was really big when I was growing up so that was really exciting, too. So science and technology just seemed like a natural place to go. I'd always enjoyed taking stuff apart and putting it back together and hopefully having the right number parts go in the right place. So, I was interested in it. I think in high school was the point at which I had to make a choice. I was interested in math and good at math. And so, I was thinking about focusing on math, but there was an electronics class in high school that I remember. My teacher was Robert O'Lincoln. And he was an ex-Navy guy that had worked on electronics in the Navy. And the electronics class there was pretty hands on. He'd build these circuits with relays, and there'd be a bug in them. And we'd have to kind of debug them and stuff like that. So, that's where I got more interested in computers and electronics. So when I went to school at Berkeley, just like Bill, again I was looking at the math versus electronics split. I decided that all you would do as a math person was teach math. And so, it seemed like an infinite loop. And so, I got more interested in computers and computer science and took classes in computer architecture and compilers and integrated circuits. I was interested in all of it so when I was looking for a job, Advanced Micro Devices and the product planning and applications group seemed like a perfect fit because I could talk to the circuit designers about my circuit design stuff. I could talk to customers about architecture, and then work on the logic design of the devices. I was working in the bipolar applications group initially on bit-slice microprocessors, bipolar technology for microprocessors. That was in John Mick's organization. And then we started second sourcing the bipolar PALs, [Programmable Array Logic devices built by Monolithic Memories Inc.] and I moved into that group.

Laws: So your first job in the industry was with AMD.

Miller: First job was at AMD, yeah. There were three or four of us that were hired right out of school into the product planning and applications department.

Laws: And what year was that?

Miller: That would have been 1978. The group was just growing so fast, they couldn't hire enough experienced engineers, so they went to the colleges and hired there. But like Bill was saying, there was a good mentoring system there. So, we learned a lot on the job. That was a real plus.

Laws: Mitch.

Richman: I'm Mitch Richman. I actually come from New York. And my dad is also an engineer, a mechanical engineer. He moved out to the LA area to get into the aerospace engineering business in the early '60s. So, I grew up in LA. To say that my dad is a pack rat is a tremendous understatement. Work benches with just stuff piled up everywhere-- So, I was one of those that never lacked for things to take apart and put back together in a myriad of different ways. When it came time to go to college, I was actually a physics major. When I discovered that they wouldn't let the undergraduates play with the particle accelerator, I switched over and became an electrical engineering guy, also at Berkeley. And one of the things that was really interesting is in our digital IC class, we had--- I can't remember who it was, but somebody who had a connection to Intel and was able to get us some of the very first 4004 four-bit microprocessors. This was hot stuff, right? I mean there was nothing like it. And so, that's really what turned me on to the chip business was, look at this thing. It's cool. In '78, I got my first job in the industry at Fairchild in Mountain View working on bipolar PROMs and, I guess you could call it, programmable logic. They were second sourcing the Signetics 82S100. I spent three and a half years there in various engineering roles and came over to AMD in a marketing role in 1981.

Laws: In 1981. Okay. Any other mentors in your background, Mitch? You said your father was a major influence on your choice of career.

Richman: Well, I'll tell you. The guy I worked for at Fairchild, a gentleman by the name of Murray Wilson, was just one of those guys who-- Fairchild was a very sink or swim situation. They threw you in. there was no training. And it was drink from the fire hose, and oh by the way you are now responsible for this. And if it fails, it's your problem. I'd been there five weeks, had no idea of what I was doing about anything, and they sent me to Singapore to do a technology transfer. It was, go do this. Here's some material, figure it out and don't come back until it's done. Okay. I spent three months there on what I later realized could have been done in three weeks if I had any idea of what I was doing. So, Fairchild was a great learning thing. And Murray Wilson was one of those guys who really helped guide you through some of the more dramatic pitfalls. At AMD I worked for a couple of really great people, Bob Lutz [who we'll talk about, great engineering guy. You bounced a lot of things off Bob, often at a high decibel level. But it worked out great. There were some great people.

Laws: The product area that we're talking about will be obscure to people in 200 years time when they look at this video. Could one of you give us a little bit of background on the evolution of programmable logic? Mitch, could you start off with that.

Richman: Okay.

Laws: I'll ask the others to chime in as they feel appropriate.

Richman: Well, you know if you go back into the '70s everything was fixed and dedicated. ROMs were the big thing. The memory configurations were fixed. The idea of a programmable memory started coming out in the '70s. Before that there had been core memory. But the idea that a customer could actually configure something themselves was radical. The idea that the customer could actually have some influence on the function of a chip, other than convincing the product planning guy at the manufacturer that he wanted it to do this, was a radical concept. So people found they could use PROMs, bipolar programmable read only memories, as simple logic things by using the programmable OR array. Signetics came out with the first true programmable logic device about '75. '76?

Sievers: Not true. But only by a month.

Richman: Only by a month.

Laws: The year, Bill, was when?

Sievers: 1973.

Laws: This was the FPLA from Signetics. The museum has recorded an oral history of Napoleone Cavlan and Ron Cline giving the background on that product.

Richman: Interesting product. Revolutionary in the basic concept, but fairly flawed over time in its implementation.

Miller: Napoleone might disagree with you on that.

Richman: I'm sure. Napoleone and I disagreed on many things. Then in-- giving credit where credit was due, in-- what was it, about '77, '78? John Birkner and H.T. Chua and at Monolithic Memories came out with the PAL, Programmable Array Logic, which simplified the AND-OR array that Signetics was using. They basically made the OR array fixed. So you had a programmable AND, and a fixed OR array, and that allowed the device to be significantly faster. But even there it was fairly limited. So they had a whole spectrum of different products. This one's got four outputs. This one's got six. This one's got eight. This one's got two product terms. So there was a wide range of those types of devices that you had to design, manufacture, and stock. When AMD got into the business, we started by second sourcing the most popular of the Monolithic Memory PAL devices. And that's where the idea of the 22V10 started to emerge because, making all these different flavors wasn't the easiest thing for the design guys, certainly it was complex from a manufacturing perspective. And customers had issues with it.

Laws: We'll get into detail on how that happened later. Who was using programmable logic devices in the marketplace? Who were the customers? What where they building? Why did they find them attractive?

Richman: Well, initially the primary customers, and certainly the most notorious, were the minicomputer guys, DEC and Data General. I should have brought my copy of Tracy Kidder's book "The Soul of a New Machine" because that's one of the things that made PALs famous. That was one of the first major sales tools that MMI had. Data General basically testified that these little things, that each had only fifty to a hundred gates, they allowed us to take what would have been a seven-board minicomputer and put it all on one board. That really lit things off.

Laws: So the advantage of the devices was that you could program them to do a specific function in a much smaller space and save cost and save time.

Richman: Right. Instead of using fifty 74LS chips, each having one or two gates, you could use just one of these and you could change things on the fly.

Sievers: Right.

Richman: Oh it didn't work? Well I'm not going to tell my boss that I had an error here. I'll just go, and I'll change it and program another one and drop it in. He'll never know.

Sievers: I think that was key to popularizing them. I think that and PALASM. I know MMI touted PALASM as allowing designers to easily implement their logic, but, the fact that you could change, I think a lot of designers really appreciated that. Like Mitch said, to cover up their mistakes or to improve the design and not have to have the whole circuit board changed.

Miller: There was a competing technology at the time, gate arrays, with which you could do something similar. You could take a lot of devices that would normally be on a board and move them inside of an integrated circuit. But the way you had to do that was build a custom mask set. You had to make a very specific device, and you could define the metal interconnects on the device. But it was done in the fab. It wasn't done on an engineer's desk. And if you made a mistake, it cost a lot of money to have to redo a mask set plus a lot of time. It would take weeks or months to do that. So, you can imagine an engineer who's working on something, if he's made a mistake, that's a real expensive mistake in many different ways. So, if you could actually program a device on your desk instead of having to do a gate array--although you can't fit as much logic on it, but it's much more forgiving technology. And as Mitch said, you can make iterations and changes. You could even change the architecture somewhat halfway through the design and really optimize it for the device. So that was a real plus.

Richman: That turned out to be a big thing because not just fixing the engineer's mistakes, but when the marketing guy comes in and says-- or the architecture guy, we really need it to work like this. Or, could you add this? Instead of going, well sure but it will take six months to do a new gate array. Okay, let me go back to my logic equations, and you could change it. And in a very reasonable amount of time, you could actually change the function and enhance the function of the thing.

Laws: So, an iterative design process was possible.

Miller: Yeah, exactly. And, again, one of the reasons that it became interesting to engineers was that gate arrays did have this perception of being sexy because it was really expensive, and you could put a lot of logic on it. It was the high-end engineers who were doing the gate array stuff. Well here was a way for anybody to do something that was similar to a gate array except you could program it yourself. So, I think one of the ways MMI sold this to engineers was "Now you can do your own gate array. You don't have to pay all the money and you can build your own chip." Before you had to do it with this real expensive technology. Now you could just do it on your desktop. So engineers could build their own device, whereas before they were always stuck with the devices that the manufacturers were providing.

Laws: Bill, I think you were one of the earliest pioneers in user programmable logic technology. Tell us about your work at Intersil on these products.

Sievers: I was hired at Intersil to design PROMs. I started there in 1970. But about 1975, they got the idea that they wanted to make a programmable logic product. They looked around, and they saw that National's mask programmable logic product was fairly popular. So let's copy that. They had the fuse technology with the PROMs. And I worked designing the IM5200 FPLA product. But when it came out, we didn't have any way to program it. So, we had to design our own programmer. George James was my technician and I gave him the task of putting together a programmer for the product. He did an excellent job on it and it also qualified him to be hired away from our company by Data I/O. But he went on there and proved to be a very good contact.

Miller: Yeah, he was key.

Sievers: He subsequently worked on a lot of programmers there. But key to getting any programmable product in the market and used is to have some way to program it. And so, being the first, it was necessary to develop the programmer. But, the problem at Intersil was they were using an old process. They had a fuse technology called AIM, Avalanche Induced Migration. That was good but required a lot of fusing current. And I don't think it allowed for Schottky technology, so ours was a gold doped, bipolar process, which was quite slow. The part was spec'd at one hundred nanoseconds. I think typically it was sixty-five nanoseconds. So it wasn't Earth shattering speed. And about a month after we introduced, Signetics-- came out with their product line. So the IM5200 was the first programmable logic product.

Laws: This oral history will make sure that it goes in the record. How long did you stay at Intersil?

Sievers: I was there until 1980 when Bob Lutz hired me to start programmable logic at AMD.

Laws: And the rest of the time at Intersil you were designing PROMs of various kinds?

Sievers: I also worked as product engineering manager there. In fact, I think that was my last job.

Laws: When you joined AMD, who decided what kind of programmable logic you were going to do?

Sievers: They had decided they were going to second source MMI, but they didn't have a license to do so. So my first job was to search patents for about six months and to try to find out if we didn't need to buy a license. I thought I had enough to challenge the patent, but the legal department didn't agree so they eventually bought a license from MMI. My first hire as designer was Paul Harvey who subsequently was the designer for the 22V10.

Laws: Warren was there any applications involvement in the selection of the MMI family or was it just so obvious that this was the product that the company should build?

Miller: Well the choice was either MMI or the FPLA products. But I think our sales force and FAE's were really good at scoping out which of our existing customers were using what programmable logic devices. And DEC and Data General were huge important customers for us. And they were very big on programmable array logic devices. So I think it was, like a lot of things at AMD, it was pretty much sales driven. And so, it was a pretty obvious choice. So one day John Mick came to me and said we're going to second source these things called PALs. Let's start learning about them and figuring things out. So it wasn't something we did a big market research on and projected sales and decided this was the way to go. But it certainly was a good decision at the time I think.

Laws: Were there any features put into the AMD devices that were not in the MMI to make them more attractive to customers?

Miller: From a circuit standpoint I think you can talk about the testability stuff, right?

Sievers: Yeah, we decided that we wanted to include testability, so that we could prove that all the programming circuitry and the final operating circuitry worked 100% before the customer received the product. I think that was a good choice. It did cost some chip real estate, but it proved that our products were very high yield.

Laws: I think one of the concerns that customers had was that when you buy a hundred parts and you program them, there's a typically a few percent that do not work correctly. What kind of fallout were they seeing with MMI parts? Do you remember, Mitch?

Richman: From a marketing perspective this was a big thing because, we're second sourcing their part, but with MMI and those guys were using Nichrome fuses. They typically had 10 to 15% of the parts fall out in programming. Now, of course, yes you return them to the manufacturer. You get reimbursed. But it's labor. It's lots of issues. And then, even the parts that programmed wouldn't all work in the system. And because of the testability that had been designed in to our parts, we were able to market and differentiate ourselves with PPFY, Post Programming Functional Yield. That was a real tool that the sales guys were able to hammer home that we get 98% programming yield. They get 85%. After you've programmed them, 99.999% of our parts work. After they program theirs, maybe 90% of them work. And that really helped. There were also three flavors of the 20-pin PAL family that we did differently right? You remember the 16H8 and the 16HD8 and 16LD8.

Miller: Right, with metal mask options we were able to do to do some additional parts,

Richman: The H was a good part.

Miller: The H was a good part, okay. But again, that's kind of an indication of the AMD culture. It was a very much seat of your pants kind of thing. You could pretty quickly come up with part ideas and things like that, and it was a very streamlined process to get them done. But talking about the post programming functional yield thing, it was important to understand how purchasing decisions got made at some of the big companies. There were groups within some of these companies that were only responsible for programming devices and testing them. The design engineers who actually designed them hardly ever worried about the manufacturing side of it. But I remember at DEC and DG there were these gatekeeper guys in the test and programming groups. And if you couldn't sell them that we were going to make their life easier, they wouldn't approve. They wouldn't allow their designers to use those devices. So the cool thing about the testability was that it really appealed to those gatekeeper guys. And once you got qualified within the company, then the design engineers could use them. And so that was a cool way to get in from the side door, as opposed to the poor design engineers having to argue with these gatekeeper guys that I want to use the AMD devices.

Laws: So you built some marketing advantages into the device so that even if you were somewhat later in the game you could still gain a viable share of the business.

Miller: Yeah, yeah.

Laws: Bill, you started in 1980, when did this first second source family get into the marketplace? Do you recall?

Sievers: Maybe as fast as a year, but it could have been two years. I don't really recall.

Laws: You were using an existing technology that had been proven for the PROM products, the platinum silicide fuse?

Sievers: That was absolutely an advantage that I'll have to acknowledge credit for. I shared an office with George Brown who had worked on PROMs and got PROMs to a level of a high yield and knew the technology cold. He coached me how to do the design of the PAL, the programmable part of it. And the fact that we were using the proven platinum silicide technology everything was there ready to make the PAL successful technologically. That was quite smooth.

Laws: So, you had this product family out there getting good acceptance, and somebody decided that we wanted to try to do a proprietary part based on the PAL architecture. Warren did that come out of applications

Miller: The nice thing about having the second source MMI products was it allowed us to get in to customers and talk to them about what they were using the 20-pin devices for. And then as you mentioned, AMD was really trying to move away from second source to proprietary products because the margins were much better and you could make a lot more profit. So it was pretty clear that we wanted to do a 24-pin product that could differentiate us, that could be a proprietary device. We talked with a lot of key customers. The AMD sales force and FAE force were really important there because they could identify customers that had challenging problems that could really feed into the product definition process. K.C. Murphy was the head of the field applications engineering team. That's a technical team. They're engineers who are calling directly on customers, understanding customer problems, and then advocating for AMD product solutions. But K.C. viewed them as the eyes and ears of AMD in terms of providing input to us on the product planning side. So we'd talk to engineers at DEC and Data General. Josh Rosen, for example, I spent quite a bit of time with. He would actually show us schematics.

Laws: Who is Josh?

Miller: He was a design engineer [at Data General] -- he's mentioned in Tracey Kidder's book "The Soul of a New Machine" as one of the engineers that used the PAL devices. He was really open. He would pull out his schematics and show us how he was using the PAL devices. We noticed that he was tying PAL outputs together, which you normally wouldn't do. But he had to do that to get more product terms because some of the things he was working on were too complex to generate the logic at single output. And he needed to do it fast. So he couldn't do multiple passes of logic, which you could do if you could go slower. So there were things like that in talking to customers that made it clear that there were some opportunities for additional capabilities in a 24-pin device. And that's when we started talking internally about what those features might be and talking with specific field application engineers. I can remember Rob Anderson from the north, and I think it was Drew Dutton in the Texas area, Sherman Lee down south. We would talk with these guys, and get ideas from them, and understand what customers were doing with their devices. So that was a real plus with AMD having that good customer reach for us.

Laws: So you had this understanding of the marketplace. How do you turn all of those inputs into a product definition?

Miller: Some of that just takes some time of noodling around and talking about ideas and stuff. So pretty quickly we decided that additional product terms were going to be really important. A product term is kind of the measure of how much logic you can do in a device. The more product terms you have, the more logic capabilities, and the more decisions you can make inside the device. Josh, in particular, was pushing the envelope. So it was pretty clear that if we could do sixteen product terms that would help in a lot of applications. Ten outputs was really important too. A lot of functions would be for eight outputs, but you typically need one or two more. You'd need to check if there was a zero, or if there was a carry, or you'd have some kind of pattern matching that you're looking for. So having ten outputs was pretty clear as an important number. And then in talking with the circuit design guys you always have to make trade offs. We couldn't have sixteen product terms all the way across, it just wouldn't fit in the device.

Laws: Because of the number of outputs and inputs required?

Miller: Just because of the amount of logic. You couldn't even put it in a package.

Sievers: The die size was constrained by the package.

Miller: So we started thinking about a variable product term distribution because not everything needs sixteen outputs. You might need a couple of them, especially if you're building counters or things like that where some of the bits are more complicated than others. So we went to a variable product term distribution, start out with eight on a couple outputs and then go to ten and twelve and fourteen and sixteen. So that seemed like a really good fit for what the customers were looking for. But then we ran into another problem. And that problem was they need variable product terms, but the way the PALs up until that point were architected is you had to define which outputs were going to be registered [have a flip-flop at the output] and which ones were going to be non-registered. So then you have to build a family, some of them with four registered outputs, and some with six and some with none. But because we had the variable product term distribution, that was an additional complexity. We had to then decide which ones to register. Do you register the sixteen product term outputs? Or do you register the eight output product terms? It looked like we were going to have to have this massive family with all these different choices of registered and non-registered outputs. The 20-pin PALs did something similar. There were a dozen different flavors, like Mitch was talking about. But this would have ballooned into a much larger number of flavors. And so we scratched our head about that problem for a while. Had a couple suggestions for how we might trim the family down to only be about a dozen products.

Laws: You said we. Who is the "we"?

Miller: Yeah, yeah good point. So, Brad Kitson, who worked for me on the program in applications, and Bill, and Paul Harvey-- Paul Harvey who was a circuit designer working for Bill. I had this famous white board in my office that we would just get on there and kind of draw some stuff around. I remember one day that we had a partial drawing of different selections for families, and it was on the white board. And I was driving in from home that morning trying to figure out "How do we organize this thing?" I said wouldn't it be great if we could just program the output instead of having to dedicate it to be a registered or non-registered output. What if we could program it, and choose to be a register or not? We could just have one device and then the user could figure out whether they want a sixteen product term output registered, or have it be non-registered. And so, I came in and Bill and Brad and myself started modifying the white board drawings there, trying to figure out how we could make the output programmable. Brad said let's just go talk to Paul. And so Paul came upstairs. You guys were downstairs at that point I think. Anyway, he came to the office and we all started drawing on the white board. And Paul showed us some ways we could add fuses then to make the outputs programmable. And so we were really excited at that point. We were worried that it would cost extra silicon real estate, or if it would make the part slower. What you couldn't do in those devices was make them slower. Speed was really critical. Luckily there were some key speed points. It had to be fifteen nanoseconds, or twenty, or twenty-five. A part that was twenty-two nanoseconds versus twenty-five wasn't that big of a deal. So luckily Paul found some ways to add the flexibility without making the part slower or at least he said he thought he could. That's when we felt like we had the right combination, that we had the functionality that the customers were looking for at the speed they were looking for and in a single device instead of inside of a box.

Laws: Bill you were watching this process. With all these desirable features, were you not concerned about how big the chip was going to be?

Sievers: That's always a constraint. I was very lucky to come upon Paul. His resume came across my desk. I think he was working for Harris but he wasn't long out of school from MIT. He was a very bright, free spirit kind of guy. He'd skate to work on his in-line skates, or ride his bike. He was very creative, excited about circuit design. And we were lucky to catch him because he was definitely key to the design of the 22V10, much more than myself.

Laws: So this is a classic process here. You had the customer input. You had the design input. You had the applications group and iterating backwards and forwards around these requirements.

Miller: Well it worked for us. And again looking back on it, it's just kind of amazing how much seat of the pants we did it. How do you estimate the sales for a completely new product like this? And how would you justify it to management? Bob Lutz was the managing director at that time. I remember the review meeting where we were going to go in and show him our spec for this 24-pin PAL family. And he was expecting to have, just like the 20-pin PALs, another dozen products. And so we went in to tell him about our 24-pin PAL family. It was a PAL family all right, but it was a family of one. And so here's the product line guy who was expecting to have to have a dozen different products to generate a certain amount of revenue and everything. And so, pretty quickly the gears turned in his head and he figured out if I can build just one product to serve this market instead of a dozen, I'd much rather do that.

Laws: True.

Miller: So he was a pretty easy sell. That was one of the fun meetings to present a product.

Laws: What was the process for getting a product approved in AMD? Obviously, you would have to come up with an objective specification. Marketing has to come up with some estimates of the market size. Mitch, why don't you take us through this?

Richman: There was a very detailed PDP, Product Development Plan, that had to have everything from here's what the thing is, a detailed analysis from design in terms of what the die size was going to be and the schedule and costing. Marketing had to make up our numbers of what the sales were going to be and margins, and so on. And after this change came up, we had to go back and rewrite the PDP because it was so significant a change [from the original family concept],

Laws: So the part was originally approved as a multi-device 24-pin family.

Richman: It was a family of eight or ten or fifteen 24-pin PALs, and that were second sourced to the MMI products. Remember this was when AMD was doing the asparagus products. Remember the asparagus campaign? Oh yeah the farmer grows the stuff he can easily sell and then he plants the asparagus which takes--

Miller and Sievers: Three years.

Richman: Three or four years to mature, but then you make more profit. This was such a dramatic change. We had to go and completely rewrite it and go back and represent it at the quarterly PDP meeting. And I'll say that one sticks out in my mind because that was only my second one of those meetings. And so I'm standing up there presenting this. And people are kind of quizzical about what the heck did you guys come up with? And at the end it went over really well, but Jerry Sanders came up to me—"Did you used to be a car salesman?" I'm not quite sure if that was a compliment or not but--

Miller: From Jerry it probably was.

Laws: He bought a lot of cars, so he knew a good car salesman when he saw it.

Miller: That's right.

Richman: But the reception to the changes were overwhelmingly positive, tempered by the question - "Are you sure you can build it?"

Miller: Well that was another thing that was interesting. I mean we were in the process of technology exchange with MMI. And I remember a meeting with MMI where we had to go and tell them what our plans were for the 24-pin device. And so, we told them about it and we were hoping that they would consider second sourcing it. I guess kind of hoping. But I remember H.T. Chua saying that they couldn't build it because our process was superior to theirs. It would have busted out of the package for them. So that was interesting to us in that it just proved we did have a superior process at that time to MMI. So we were going to be able to put more functionality into a 24-pin package then they could.

Laws: Into a die size that was manufacturable.

Miller: Yeah.

Laws: Bill, please tell us a little bit more about the technology at the time? What kind of product dimensions, how big a die could you build, what tools did you have available to design with?

Sievers: Well I was pretty much constrained by the power and the die size that a particular plastic package which we were using the, what the limits were for that. Exact numbers I don't recall, but those were the constraints. If you met those, then you had a product.

Miller: Yeah because power dictated by the package, too. Otherwise it wouldn't help.

Richman: Here you go. [Shows a document] Here was your new product status report on that part from 1982. Up in the upper left hand corner it talks about the die size.

Sievers: Okay, twenty-two thousand mil squared.

Richman: Which as I recall we had to go and have the package redesigned to allow a die that big to fit into that thing. They had to make a new lead frame for the plastic package.

Sievers: But it says it's an IMOX 2s product with sinkers, which allowed you to get a lower output level and one-thousand ohm implant resistors, which probably allowed you to get smaller physical size resistors.

Laws: IMOX, [was an acronym for] Ion-Implanted Oxide Isolation.

Richman: But the interesting thing was from a marketing perspective, that we were able to market this as being up to about a thousand logic gates where the 20-pin family was fifty to a hundred and the MMI 24-pin family was two hundred to four hundred. And we could go out and say here this thing is going to do everything that all these do with twice as much logic and all this other flexibility. Once you could convince people that you could actually make such a thing, it was not a hard sell to a design engineer.

Laws: Mitch I you just set Bill up for the next chapter in this story. Talk about the design, Bill. What did it take? You said Paul Harvey was assigned as a designer. You oversaw his work. How long does it take to do the electrical design of a product like this?

Sievers: I think typically it takes a year to-- or at that time, to bring out a product. And I didn't do any homework on exactly how long this took. But we had a number of problems as represented by an excellent group of cartoons that were drawn by a test engineer that was assigned to this product. He was definitely frustrated. His drawings were an outlet for him to describe what we went through. It was painful. I would say one of the more painful products that were put out.

Laws: Why was that? What was it about the product that made it difficult?

Sievers: Well it was- a new process. I think the IMOX 2S, and Paul was relatively new to designing. And we didn't have all the modern tools that they have nowadays to check design. So, you couldn't put your schematic in and your layout in and have them agree with one another.

Miller: Were they still putting blocks [drawings of pieces of the design] out on the cafeteria floor?

Sievers: You had to put down the blocks and--

Miller: That was the way that they checked.

Sievers: Check by hand.

Miller: Yeah.

Sievers: Every contact, every resistor value.

Laws: You weren't cutting Mylars at that point were you?

Sievers: No, we were drawing them with Calma systems. But you still had to check the Calma plots and use multiple colored layers to make sure they all overlapped in the right places. The initial problem was there were some extra vias that were put in between Vcc and ground.

Richman: Ouch.

Sievers: Which measured as a twenty-two ohm resistor.

Miller: That's why it was the 22V10?

Laws: Okay, so this device had ten outputs. That's probably where the ten came from. And the V came from versatile?

Miller: Right.

Laws: What was the twenty-two?

Miller: Number of inputs.

Laws: Twenty-two potential inputs.

Miller: Actually, so I think at MMI when they numbered the original PAL family it was the inputs and then a letter designator, and then outputs. I think there was a similar taxonomy for tubes where there's a number, a letter, and a number. And that's kind of where they got that from.

Richman: And we were able to do more inputs because they [MMI] also used a dedicated programming pin, in order to supply the high voltage, so they had to tie up an entire pin. Bill and his guys came up with a way of generating that internally and routing the programming stuff separately. So you didn't have to waste a pin.

Sievers: I don't think they had an extra pin. But, they definitely had a little different way of programming.

Miller: Yeah, yeah. Programming was, as Bill was saying, a key point., But some of our sales guys were a little confused because we were second sourcing MMI devices but they programmed differently. So it wasn't a direct second source, customers had to do the programming differently because our technology was different from MMI.

Laws: They required different voltage and current levels than the MMI part.

Sievers: And when you had a new device, you had to convince [third-party programmer] companies like Data I/O (they were the most important supplier) that it was going to be in their interest to support it. That they were going to sell enough of them.

Richman: There was a special adapter.

Sievers: Adapter for their Model Ten programmer. You had to convince them that they should do that. It helped that we had George James there.

Miller: Exactly.

Richman: And don't forget Roland Fritz Didn't he do the evaluation of all of the programmers as they came in?

Sievers: It was later that Roland was involved with programmers.

Laws: So the other people in the creation of the product were who. There's Paul Harvey as the design engineer. Somebody had to write the test programs. There has to be a product engineer who manages the product through a lot of these processes. Can you tell us about some of these people and what their contributions were?

Sievers: The test engineering was done by Jerry Katz's group, which didn't report to me at the time. I think I didn't hire a test engineering manager, Jeff Leong until later. Gary Knight, who was also the cartoonist, was the test engineer that was assigned to write the test program. And they had a lot of their own problems that delayed the product, too. The product engineer was Mike Nelson. His supervisor was Roland Fritz.

Miller: Yeah.

Sievers: And he and Paul debugged and got the part finally working.

Laws: The design of the mask, who did that? Was there a particular layout person?

Sievers: Mae Yee was the 22V10 mask designer.

Richman: Pauline Seals was the process person.

Sievers: Pauline Seals. I'm glad you have that name.

Richman: Well she was the process engineering person assigned to it. I don't know that she was the actual mask designer.

Sievers: No the mask designer-- entirely different function.

Laws: So there are about five people involved in this process.

Sievers: Yes.

Laws: A couple of design people including part of your time, a product engineer, a test engineer, a borrowed mask designer, a process engineer. They all have to work through these issues.

Sievers: Right.

Laws: What's the toughest issue you remember, Bill?

Sievers: Getting first working silicon, just the pain of it. And the spotlight was really on us. I mean these guys had whipped up the enthusiasm for the product. And in a way it was helpful to the design group because we didn't have the visibility to the outside world. So were always in isolation doing our tedious work, whereas they come in and tell us about the customers that want this stuff. And so that just added the pressure. When the thing wouldn't work the first time, we're disappointing all these people. But, it was an exciting time. It was one of the best times in my career, I think, to work with a group of people that were going out, talking to customers, seeing what they wanted then we actually could do this very challenging product for them.

Laws: I think this is about the time that I came in, must have been about '82-- 1982 or so.

Richman: Late '82.

Laws: Late '82, and we split off a separate business unit for programmable logic devices. You were working for Bob Lutz, I believe at the time.

Sievers: Right.

Laws: My job was to keep the folks off your back so you could get the job done, and buy pizzas and beer occasionally, I seem to remember.

Sievers: You were one of the better people I worked for in understanding what people were going through and making it easy for us to get resources and things.

Laws: Thanks. Mitch, what was your perception of what was going on during this time?

Richman: Well it was interesting because, as was mentioned, they'd come up with a great product. It was fairly easy for us to whip up a lot of enthusiasm in the sales force. Yes, you had to get them over the hump of, you're selling something different, you've got to design this in as opposed to just dropping something in to somebody else's socket. Yes there's a different programming adapter, but look at all of the benefits. And the customer's going to be locked in. So the sales guys and the FAEs were very bought in. Originally, this was supposed to be a very late '82, first quarter '83 sampling. And due to a number of the issues that Bill's talked about, it slid out a few times. And from a marketing perspective, it was interesting every quarter, going in front of the area sales managers and saying, it's coming soon. But we're not quite there yet. So that was an interesting challenge because there was a lot of demand. By the time we were able to start sampling this in mid to late '83 I think it was, there was a lot of pent up demand. And there was clamoring. There wasn't any programming stuff in the field yet so for the initial samples, we had to get the information from the customer. They'd give us their JEDEC programming file, they'd have run it through our PALASM software. And we would program it at the factory for them. And then somebody, I remember-- I think you mentioned you hand carried some to HP.

Laws: Yes, HP had a big product development for the Spectrum series computers at the time. They were using PALs to emulate the architecture and test out their software. And they wanted hundreds of parts and we could deliver two or three a week as I remember. I would hand carry them up to the HP plant in Cupertino. Interesting times!

Richman: And we did the same thing with Data General. These things didn't get shipped through regular channels. When we had half a dozen to give to Data General somebody got on an airplane and carried them. So, it was really interesting. We put together this whole aggressive marketing program with the family of one. Dave at that time was certainly very actively involved in the marketing. At AMD at that time, if you had a really hot product, instead of just a plain data sheet, they would let you put a glossy wrapper on it. This was fancy stuff. [Holds up a data sheet]

Laws: The picture on the front? What's the story behind the Swiss Army knife?

Richman: You should tell the story because that was you. You're the one who came up with this, the world's most versatile logic tool.

Laws: And I think you had some knives made up as samples.

Richman: Once we came up with that we even went ahead and made real Swiss Army knives with the 22V10 logo on it. I think we bought thousands of these. We gave them out at the sale. Those were the days when you could carry knives on airplanes. So we gave one to every sales guy, every field applications engineer, passed them out in the factory. It's how come I have a few thirty years later. We did whole big-- I'm not even sure what you'd call it. But that was for the World Wide Seminar series. We had a big chunk of the 1983 World Wide Seminar was dedicated to programmable logic. And we spent a lot of that on the 22V10 and all of the wonderful things it could do. So, we had people whipped up into a frenzy for you, Bill.

Laws: Warren, you were busy creating other tools for the customer to use. I believe there was an AMD version of-- we talked about PALASM a few times. That's Programmable Array Logic Assembler, which is a program the customer uses to translate his logic design into the fuse pattern that will be used to program the part. Can you tell us a bit about what you were working on?

Miller: Because this is a user customizable device, you have to allow the user to define what's going to go in it. And the way they did it was to write logic equations on a computer. And the computer would take those logic equations and turn them into, what Mitch talked about, a JEDEC file, which is basically a bunch of ones and zeros that the Data I/O programmer would then use to actually program the fuses inside the device. So we had to put together software that we called Programmable Logic Programming Language, PLPL. There were a few people within my group that were working on this. Phillip [?] was one of the new hires that we brought in to put that together. And that was kind of new for a semiconductor company to not only just sell the devices, but to have to provide some software for the customer to use. Customers were used to doing that for microprocessors and for bit-slice microprocessors like the AMD2900 family and stuff like that. But this was one of the first times that the applications group had to develop the software. So, it took a while to get that together. But customers were then able to load it onto their IBM PC and then define the product, and then hook the Data I/O machine up to the computer and program it.. But there were some struggles for us to do that. I mean it was the first time that we'd put software together. And it wasn't like we had a proven software development release and test kind of process. So, we just kind of had to do it by the seat of our pants. It was one of those things where looking back, if you realized how much trouble it was going to be, maybe you wouldn't have done it. But we had to. I mean there wasn't any other way for customers to use our devices. So, that was a challenge for us. We just jumped into the pool and figured out how to swim.

Laws: This was an interesting time of transition in the semiconductor business. It was started by the requirement for providing customers with tools to design with microprocessors and micro controllers. Programmable logic was probably the next [product area to demand tools], and nowadays probably every product has an array of software and other tools that the customer needs to use the device.

Miller: Exactly. This was a learning experience for a chip company, for AMD. The bread and butter business was just second sourcing other devices. For those none of the programming or any of the complex technical support activities were important. This was a new device. And so, we had to do a lot of training for customers and training of the FAEs. We had to put together a whole kind of book on how to use these devices with not only the data sheets, but examples of how to program different functions into it. And then the document that Mitch showed, this is something that you would provide training, not only for our FAEs, but for customers. Why is the 22V10 a better device to use? And what are some of the things you can do with it? How do you do a state machine? How do you do asynchronous logic? All these were things that we now had to teach customers how to do it. A Karnaugh map, how many of those customers had forgotten Karnaugh maps from their college days, or maybe hadn't even used them. So, that was the other thing that our group had to do, not only do the development of the product and work with engineering, but also think about what are customers going to need to be able to use the device effectively, the software, the training. And then, get the FAEs-- the field application engineers trained so that when a customer asked them a question about how come PLPL isn't working for this device, they could help answer the question. Some of that was self-serving. We didn't want to take the hundreds of calls from the customers. We wanted the FAEs to be able to do it. So, like you said, these were some of the changes that were going on in the industry that were particularly tough, I think, for AMD. This was the first time that we were at the forefront of this kind of technology.

Richman: There was also a real marketing and sales issue with PLPL because the customers had to have a tool. And the sales guys, of course, wanted to give this away. And AMD was adamant that you couldn't give it away because we didn't want to kill the ABEL programming language from Data I/O. And there was another one called COUPLE, which was from Bob Osan [spelling?]. As those were emerging there was a real question as to whether we wanted to be in the software business long term. Well, if you give these tools away, aside from the fact that they cost money to develop, if you give them away, you're going to kill these other guys. And then you are in the business long term because there's no other alternative. And I remember getting a lot of those calls from sales guys. And after I said no, I got to pass them to you. And you would get to say no.

Laws: Do you remember Bill how many design iterations were there on the 22V10? Did you go on to redesign for tighter process rules, or other ways of making the part cheaper, and more attractive, or faster?

Sievers: I know there were a number of iterations. Z, designates the first mask set and I recall a W at some point. But specifically why we made each iteration, I don't recall.

Richman: The big one that I remember was in '84, you did the new mask set for the San Antonio fab.

Sievers: Oh, well that's just for a different manufacturing plant.

Richman: Well, except that was what saved our bottom line because we were making these things in Fab 5 in Sunnyvale. It was very limited as to what we could make. And once you did the redesign and we got it into San Antonio, all of a sudden we were able to start cranking out tens of thousands of parts. So, that one I remember.

Sievers: Better memory that me. But I don't remember any major design changes, per se, other than correcting the initial mistakes. I think-- and perhaps trying to speed up the product to a new process or something.

Richman: Yeah, there was the-- late '80-- mid to late '85, I'm trying to remember what I found from looking through some of these old notes. An IMOX 3 version that was going to get us to the A and the B speed grades.

Sievers: Yeah, so they were mostly for process changes.

Laws: And speed improvements, better yields to certain speed levels, rather than massive changes in the product.

Sievers: The first product was 15 nanoseconds. Did we make 15? I don't recall.

Richman: The first one was 35.

Sievers: Yeah 35.

Richman: And then the A was the 25. And I think the B was the 15. I have the advantage of being able to look through the annual sales conference books and see what we were telling the sales guys, and therefore the public, on these things each year.

Miller: Yeah, so the preliminary data sheet says 35 nanoseconds for standard and 25 for A.

Richman: We couldn't make the A part out of Sunnyvale. It was like a one percent yield thing.

Laws: So, Bill what products did you work on after the 22V10? Do you remember?

Sievers: I think that in 1987 we merged with MMI. And we started our investigation into implementing a lot of these things in CMOS, and eventually did implement in CMOS. But before that, I think it was more speeding up the products we had, maybe adding an 18P8 device, a few variations like that. But nothing quite as significant as the 22V10.--

Laws: You didn't create another radically new architecture?

Richman: There were some iterations that the product planning guys came up with. There was the 21VT8. Most of these things, I think, didn't actually ever get built. That was a dual-banked one with multiple clocks. A 21VR10, I have no idea what that one was. A 23S8 was the one that we did build with the internal buried state registers, so you could build internal state machines and use it as a very high speed sequencer. And we even a defined family of ECL parts because ECL was very big back in the '80s.

Sievers: Never got off the ground.

Richman: And AMD was doing a whole lot of ECL stuff. And we defined it, but we never built it. Right?

Sievers: We never built it. I was never a believer that we could, but we did have an effort going on it. And it just never got off the ground. They couldn't get past the power problems.

Miller: At that point, I'd left. I think I left-- I think it was maybe in '83, or so. Om Agarwall came in after I left, and I think-- and Brad Kitson was still there defining some of the products. I'd left to go to Step Engineering to help design the processor board for the bit-slice processor development system. So, I actually got to use PALs in a design with a 68000-based microprocessor. So that was fun. But after a few years of doing that and then doing some consulting, I ended up back at MMI doing product planning. And at that point, I was working for Napoleone. And then the merger came along. And so, there was I was back at AMD. And so, that was interesting though because during the time the 22V10 had turned out to be a great product and generated a lot of sales. And so, it was fun to be back at AMD at that time.

Laws: And CMOS versions of it by other manufacturers, Cypress in particular, I remember. And I believe Lattice among others built CMOS versions and generated substantial revenue.

Miller: Well, there was an effort with Seeq not to do-- well, maybe not the 22V10, but the 20 pin devices. So, there was some CMOS stuff going on, as well.

Richman: And AMD had their own CMOS PLD development effort in I think it was late ’85 or early ’86. CMOS had really been coming along. AMD’s processors were getting a lot better. I moved out of the marketing and transferred over into the CMOS PLD engineering group. And we were working on both an E-squared version, and at that time, was the very early stages of what later became Flash memory. And unfortunately, by the time I left in, I think, early ’88, we still hadn’t been able to build a product and get a CMOS PLD product out to the market. It was a lot more complicated than everybody made it out to be. CMOS, in those days, had a tremendous number of both technological, and manufacturing, issues.

Laws: And you were compounding that challenge by adding an erasable memory element of some kind.

Miller: Well, that was one of the interesting things about AMD. I mean it’s an example of your strength turning into your weakness. The bipolar memory group within AMD was a very, very strong group. And very protective, I think, of its people and it’s technology. And I can remember some of the discussions about CMOS versus bipolar. And I think because the company was divided up by technology lines like that, it was very difficult to see and develop the CMOS side of things. The bipolar group was very strong, but very-- they built their walls. And nobody was going to attack. So, if the company, maybe, would have been organized a little more forgivingly, not having so much of the technology organization, maybe the move to CMOS could have happened quicker and more effectively. But that was a really strong group. There were some amazing people in that organization and did some really difficult things.

Sievers: We had zero experience with CMOS, so it would have been like starting from scratch. I don’t think it was so much we resisted it. We saw it coming.

Laws: There was certainly some resistance from management in terms of crossing that boundary. The knowledge of the market and product design and all the issues were in the bipolar group, and all the process related to CMOS was another building somewhere else, and it was very difficult to organize it to make this happen.

Richman: When I went over there, they had “Oh no, no, we’re not going to go talk to those bipolar PLD guys about the marketing and the forecasts. We’ve got somebody here who’s going to do CMOS PAL marketing.” And it was a complete division.

Laws: I think Warren made a good point there that one of the strengths of AMD in the early days was that it comprised a bunch of independent, very smart groups that could achieve something with a relatively low level of resources. But once things got more complex, then AMD couldn’t afford to fund all these different businesses to the same degree and had to make some tough decisions as to how the company should be organized and where it should focus.

Sievers: We did make that transition in the late ‘80s. It was gradual. And we did it by bringing in a few people who had CMOS experience.

Miller: It wasn’t just us, though. MMI, I think, had similar kinds of issues.

Laws: I think most companies did.

Miller: Yeah so, MMI was really strong in bipolar. And they tried to do a CMOS deal with-- was it Cypress. And they just couldn't get it off the ground.

Laws: How long did you stay with AMD, Bill, after the merger with MMI?

Sievers:. It was really a culture change and we were embedded-- I think we were about $7 million (in revenue), something like that, when we merged. And they were 50-- I don't remember the numbers that well. But we were much smaller. And so, we went right into their building. John Bourgoin was the manager for our group. And we were sort of competing with the old MMI group as to who was going to build the next PAL, what technology we were going to use. We managed to kind of work it out, but it was never the old AMD way of doing things. And I think that was in 1987. I stayed until 1997. And towards the end there, I moved into a role called customer and engineering service. That was engineering to take care of customer complaints, returns, things like that. And to investigate new tools for the engineering group for-- like design tools. And that was exciting for me. I really saw that this was way to become more productive in design, to get designs out quicker, and to provide a more satisfying environment for designers because it was terribly frustrating to have to deal with errors in the cycles through the fab and all that sort of thing throughout your career before that. But I think nowadays, it should be much more productive.

Laws: With 2 billion transistors on a chip, it has to be.

Sievers: Right, it's impossible to physically look at every contact, measure every resistor.

Laws: What did you go onto? Let's see you were at AMD for 17 years?

Sievers: 17 years, I retired in '97. I did a little bit of consulting, but not much after that, and I've been happy in retirement ever since.

Laws: Warren, you came back to AMD for a while, and then what did you go onto after that?

Miller: You can run, but you can't hide from AMD. So, yeah so, I was in the programmable logic group after the merger, but pretty quickly after that I moved into the networking group, and also the product planning group. Let's see, I think Steve Dines was leaving from the strategic development organization. And so, I took over for Steve, that was director of strategic development, which is what product planning and applications had kind of morphed into, but with a less shooting from the hip kind of approach, one more of looking at markets and participating in standards bodies and that kind of stuff. So, I did that for a little bit and then took over engineering for some of the networking products. I was director of engineering for the fiber distributed data interface [FDDI] devices. That's where I got a much better feeling for the blood and sweat that Bill and his team and the test people and process and product engineering people put into these devices. So, that was a real learning experience, and to really understand how that all worked. But after doing that for a while, I went to distribution. I worked at Marshall Industries for a while. I had never really seen the sales side of things. I'd always interfaced with the sales people, but I felt like that was one of the areas I really hadn't had much experience with. So, I worked with Rob Rodin and his team over there. And that was great to really see how the sales side of things worked, especially in the distribution side. I was managing the FAE team and I was putting together some of their demand creation programs. How do you make it easy to sell products from a wide variety of suppliers into the marketplace? So, learned a lot there, but then basically decided to go back into consulting after doing that for several years. And, I'm still doing consulting right now, some of it for programmable logic companies, some for software IP companies. I've got a blog. There's a new website "All Programmable Planet" [http://www.programmableplanet.com/]. And so, I'm a blogger on programmable logic. The last couple months, I've been interviewing people that started out in the programmable logic business, Steve Trimberger at Xilinx. And I just got done with that. Then I'm going to have an interview with some Altera guys. My blog is basically what's going to happen in the next five years.

Laws: What is the state of the programmable logic market today? What are people using them for? What kind of complexities are they working with? It's changed dramatically since the 22V10.

Miller: Oh yes, it's changed a lot. So, today you can put thousands, if not tens of thousands of the equivalent of the 22V10 into a single device. And talk about the software that's required now, in many respects, the software is really much more complex than the hardware that goes into the devices. I mean the hardware architecture inside of field programmable gate arrays now is pretty simple in comparison to the software. But you can build a complete system on a chip now. A 22V10 used to be a piece on a board. And there were a dozen boards in the system. All that stuff goes onto a single device now. You could have multiple processors inside, lots of memory. You have high speed SERDES IOs you can hook directly to Ethernet into all sorts of high speed serial interfaces. So, complete systems are going onto these devices now, where with the 22V10, you could build a little state machine in a tiny subsystem, but now a complete design goes into it. And where one guy could define a 22V10 in an afternoon, now it can take a dozen people a year to design some of these really big devices. But it's really changed the way electronics is developed. And no longer do you have to wait for the semiconductor company to build these devices. You can build them yourself. There used to be a lot of companies out there that were building application specific products for communications or for networking or for video. And you see less and less of those products now because customers can basically design those themselves. The programmable logic devices are big enough.

Laws: And there are libraries of preconfigured functions that can be dropped into the design.

Miller: Intellectual property, IP, cores is what they're called. And so, you can drop in a processor core. You can drop in an Ethernet MAC. You can drop in video processing. You can put in DSP functions. And ideally, they all work nicely together and have the same bus, and the timing works out and everything. And so, design now is less writing equations, which is what you did with the 22V10, and now basically selecting a library of components, plunking it down on the device, and hooking them all up, and then doing a little bit of logic to make sure the bus interface works, and it interfaces to the outside world. And you're kind of done.

Sievers: What has become of companies like Data I/O that supported these smaller products?

Miller: Well there's still some of these small products around. And so, there's still the need for programmers on-- especially on the CMOS side of things. But some of those companies have morphed into doing more services and more software kinds of things. Like I said, the new devices require a whole lot more software support.

Laws: Who are the big suppliers of programmable logic devices today?

Miller: Xilinx and Altera are the big ones together with some smaller companies, Lattice, Actel (who was purchased by Microsemi) are probably the big four right now. But the big trend now is doing products that have actual microcontroller units, processor units that aren't defined by the customer, that are fixed by Xilinx or by Altera. And then a lot of programmable logic strewn around it all on a single device. So, you basically have a CPU that you can program to do your functions similar to a PC. But then the programmable logic can do special instructions and can do additional logic and stuff like that. So, the big trend is more and more of these big fixed function things in the middle with programmable logic all around it. The stuff that used to be multiple devices on a single board all coming together into one device.

Richman: But a lot of programmable logic around it. I mean you can put and ARM processor and a million gates of programmable logic that can be defined to do anything you want. And there's your entire system on a single chip. It's changed a lot.

Miller: Like you said, there are whole industries that have developed that are new from when the 22V10 came out. So, there are companies that just build intellectual property cores, just build the stuff that a designer's going to buy to then put into their programmable logic device. There's software that you buy specifically for synthesizing your logic or doing digital signal processing algorithms or video processing algorithms, those kinds of things. So, there's similar to the way there's a software industry built around the PC, there's this intellectual property industry that's built around the programmable logic devices that provide these functions for users to use, and then algorithms and software to--

Laws: So, the [programmable logic] industry—that in the mid '80s was maybe a 50 million dollar business worldwide, is probably at least a 5 billion dollar business today, if you take all those companies building chips, plus all the support services involved.

Miller: Yeah, yeah, I think that's correct.

Laws: Mitch, what did you go onto?

Richman: Well, as I said, I moved from the bipolar marketing into the CMOS PLD engineering group, and then I think it was about '88, I looked at some opportunities and decided to try something besides programmable logic for a while. I ended up at a small company that was trying to get into the PC chipset business with high speed cache controllers and memories for the '386 and the brand new '486. And that was a very interesting two years of a completely different management style. Imagine the Intel or AMD constructive confrontation style of management taken to the extreme. Some of us might refer to it as management by shouting and yelling. I might never say that. After a couple years of that I actually got sucked back in to programmable logic and went to Xilinx and spent most of the nineties at Xilinx working on everything from-- well initially we started off working on the 4000 series FPGA (Field Programmable Gate Array) family. This had, roughly the same kind of impact on the FPGA world that the 22V10 did on the PAL world. It was tremendously more powerful, dramatically more flexible. And it became, in the span of about three years, a very, very big seller. I don't want to call it dominant because Altera and some of the other folks would object to that, but let's be honest. In those days it was the dominant product family. I also continued to work on a variety of higher density and higher performance programmable logic devices. I left Xilinx towards the end of the '90s. Got out before the bust of 2000. And since then, I do some consulting work, not nearly as aggressively as it sounds like you do, on strategic marketing and some product planning and development for a variety of companies and I am happily semi-retired. It's nice, I can take on interesting projects that yes this looks good –" this is fun. I want to do this. No I can't do that one." I'm going to go spend three months in Europe so thank you very much for this opportunity.

Laws: When you left Xilinx in the late '90's, about what was the level of complexity of parts that were mainstream then as compared to the 22V10, for example?

Richman: Well, at that time, we'd gotten up from-- when I started there, the XC3000 family-- the original FPGAs were in that 8 by 8 matrix. You had 64 logic cells, which was why when they first started coming out with that in the '84, '85 time frame, an AMD marketing guy looking at it as a competitive threat saw a very low gate density, and very poor speed performance because of the CMOS technology that was available at the time. This is going to take at least five years to morph into any type of a competitive threat, which turned out to be fairly accurate. In 1990, they'd gotten up to about three thousand, four thousand gates. By the middle of the '90s, we were shipping products in the fifty to seventy thousand gate range. And by the end of the '90s, I think we had the Vertex 2 or Vertex 3 family at somewhere in the neighborhood of a half a million gates. I mean we're talking about two orders of magnitude change in the semiconductor industry in a decade. It was just a tremendous period of change, driven largely by process technology. I mean there was just no way to put a half a million programmable logic gates on a chip in 1990. So, the technology advanced. The design techniques and tools advanced. And we spent a lot of money at Xilinx on software development because that had definitely gone from the, okay somebody can write things to we need to do these things in C and VHDL and Verilog. And it all needs to be compiled down. So, it was a very interesting period in the programmable logic business. That's also that period that saw it basically fall from-- in the '80s there were what, twelve, fifteen companies doing programmable logic? Until, by the time you got to the end of the '90s, you had Xilinx, and Altera, and a few other folks.

Laws: And a new architecture every two or three years that could never really break through the lead that those two companies had built.

Richman: And there continue to be some really-- as Warren said, really interesting technology and architectural developments in programmable logic that people come up with. I mean some of the guys that we've all worked with have gone off and started companies trying to do it, and it's just-- the support level that it takes in terms of software, hardware, the entire environment you have to surround a chip with these days. It's really hard to try to break in against these two guys that together now control close to 80% of that market.

Laws: As for my career after running AMD's PAL business, for a couple of years I served as vice president of business development working for Tony Holbrook. Then in 1990 I accepted an offer from Altera as vice president of marketing. The company had just completed a 2 million dollar sales year. I stayed about 4 years, by when we had reached one hundred million in sales, and went on to run a couple of other companies. So, I also lived through much of the programmable logic growth that we've discussed here. There was one other another interesting venture in the programmable device business. I consulted with IMP on a programmable analog device. They could build a neat customized Op Amp for about five dollars. Unfortunately, with off-the-shelf devices selling for 10 cents it didn't go very far. But it was an interesting technological exercise. Are there any other comments any of you would like to make about your experience at AMD, the culture there, what you found in the outside world, and why AMD was a fun place to work, Bill?

Sievers: I just want to thank you, David for inviting us. It was a pleasure getting to meet these gentlemen again after 15 years or more. But I was very happy and lucky to have been at the right place at the right time to get involved with the semiconductor industry when it got off the ground, and very satisfying career, and got to work with a lot of very sharp people like these two gentlemen. I am just very happy I was involved in the industry.

Laws: Exciting times. Warren? Any last comments you'd like to make?

Miller: Well yeah, it was an exciting time. A lot of us that started right out of school didn't know what we were getting ourselves into, and we just kind of did it, just jumped right in.

Laws: There weren't too many textbooks to tell you how do go about it.

Miller: Yeah, so we were figuring things out. And AMD as an environment was really supportive. There were some hoops you had to jump through, the product development committee meetings, and the forms that we'd have to fill out and everything, but after a while you kind of figured out it was rubber-stamping stuff. And you knew what numbers you had to put down in order to get the approval. And what were they going to do? Come back in five years and say, well gee your projections were off by 50%. So, we kind of knew that if we talked to customers, and we understood what they really needed, and then worked with the technology guys and made the right trade-offs that we'd come up with something that would be pretty good. And the industry was growing so fast then. It was really forgiving. You could stumble a couple of times, and then you'd have time to make it right. So yeah, it was an exciting time to work. You can't help but look back on it and think of the opportunities that were missed. Here AMD, we had this great product and great family, and we let Xilinx and Altera kind of take over the market. And I was at MMI when we were second sourcing the Xilinx devices, and I saw the same kind of thing. It's just a classic business example of a company that's really good at building buggy whips just kind of missing out on the next wave of technology innovation. I remember Mitch looking at the Xilinx parts the same way and saying, gee somebody's going to implement logic using SRAM. I had done error detection and correction stuff for memories, so I knew that the FPGA's [on-board] configuration memories were going to lose their state. And was anybody really going to be comfortable with hardware, which is supposed to be hard, implementing it with memory.

Sievers: Having to reload the thing. Every time you turn the power on!

Miller: Exactly. Having to reload the thing. So I'm sure there were a bunch of us that kind of had our blinders on and just didn't kind of see where it was going to end up going. It was because we were so focused on doing what we were good at. And within AMD, like you said, we were kind of in those directorates that were really good at focusing on what they're doing. But what I think we missed out on was the ability to do some objective looking at the future and seeing what some of these technologies really might turn into. And if you look at AMD today, you could almost say that it's still got that kind of mindset. There's a certain class of things that AMD is trying to do, but it seems like there are opportunities that just get missed because you get so focused on having to catch up with Intel or that kind of thing.

Laws: These classic changes in the business model are very tough to recognize. When I went first to Altera, I was surprised at the realization that for the first five years of the business we were going to be selling mostly software. As remember we enjoyed 95% margin on the tools, and 2% margin on the chips. It took about ten years before the silicon grew into a big business. Mitch, in my experience you usually like to have the last word. What are you going to say to us?

Miller: That's why we sat you here.

Richman: Well, a few interesting things. Coming to AMD from Fairchild was an interesting culture shock. I mean Fairchild was very insular, and they'd just been bought by Schlumberger about eight months before, which really was a tremendous change. But coming to this very flamboyant company-- remember in '81, '82 to the company spent spend two million dollars on a Christmas party. And in looking through the notes in 1983, we had a 22V10 design contest. I don't know if you remember that, Dave. We gave away a Corvette. We didn't do those kinds of things at most other companies. And so it was really an interesting experience. A lot of great people. I'll say management-wise it was interesting because in my four or five years on the PLD marketing side, I went through five managing directors.

Miller: That wasn't your fault, though.

Richman: I kept making them successful.

Miller: Exactly, and they moved on to bigger and better things.

Richman: I started at AMD and two months later Ralph Cognac who was at that time our director of marketing for bipolar memory, and had brought in an entirely new marketing group. There had been a complete turnover in bipolar memory marketing in the last twelve months. So in November of '81, he hired me, an engineering guy from Fairchild, Steve Donovan who was a sales guy who had just come back from Abu Dhabi, and a new college grad as a junior marketing guy. And eight weeks later he's gone. We're like "okay how hard can this marketing stuff be?" But it was really an interesting company to work for. Let's put it this way, no company that Jerry Sanders was running was going to be dull.

Laws: That's right.

Richman: And in terms of laying a foundation AMD may not have been able-- we may have not been able - to see the future and read all the tea leaves properly. But you certainly got a lot of practice at reading tea leaves and seeing how a lot of different pieces went together. The directorates may have been very pipelined, but within the directorates there was a tremendous amount of communication between design, product, test engineering, product planning, marketing, the process people, the fab people, production control. And that exposure, for me, was really enlightening and certainly I think laid a very solid foundation for later all of the business unit management and P&L management that I was doing at Xilinx. So I really enjoyed it and it was a really interesting group of folks.

Laws: Thank you all very much for coming all this way to join us today- Bill from Arizona and Warren from Washington, and to Mitch for getting up early to participate in this session.

Richman: Well thank you for putting it all together David.

Laws: You're welcome.

END OF INTERVIEW

