Oral History of Ping Chao

Interviewed by:
Paul Estrada

Recorded: January 30, 2008
Mountain View, California

CHM Reference number: X4442.2008

Paul Estrada: Hello, this is Paul Estrada and I'm going to do an interview, a personal history of Ping Chao. We'll go ahead and get started. Ping, probably a good way to get started is if you tell us a little bit about your background, where you were born, and where you grew up.

Ping Chao: Okay. I was born in Taiwan in 1951 and so I grew up there and went to college there. I got my Bachelor's of Science degree in EE/CS [electrical engineering and computer science] in National Chiao-Tung University, which is a top notch engineering college in Taiwan specializing in electrical engineering—communications, electronics, physics, computer science, and so forth. I graduated there in 1974. Then I had one year of military service in Taiwan and spent a year there. After that, I came here to the [United] States. I came to the States in 1976. I went to study at U.C. Berkeley and in about a year and a half finished my Master of Science degree, again in EE/CS. Actually, my Master program was in computer architecture, which is not quite directly related to what I end up doing eventually in the EDA [Electronic Design Automation] industry, but I have my general training in EE and CS—basically in computer architecture and so forth. After that, I've been working here in Silicon Valley—at various companies—and got involved in EDA for most of my career.

Estrada: When you went to school in Taiwan, you went to university in EE. When did it become obvious to you that you were going to go that route and go into electrical engineering?

Chao: Actually, the system in Taiwan back then was that when you start college, you're already decided what your major was, so pretty much in my senior year in high school I'd already decided to go into my preference. Obviously you have to go through tests and admission process, so my number one preference was EE and CS. So yes, in my senior year I decided to get into that. That ended up working fine for me. At the time—as you can imagine around the time of 1974 when I graduated—what's interesting—again this is in Taiwan, it wasn't in the US—I'd never heard about a microprocessor. In 1974 that terminology was unknown in the leading engineering college in Taiwan. So when I came here in 1976 and all this—UNIX and Berkeley and microprocessing—was almost a shock to me. When I came to the States to learn all that stuff going on. Imagine that back then. Things really happened very quickly, and the rest of the world might not even catch up with it. Unlike today, back then there was no Internet and so forth. Literally there was a culture shock when I came to Berkeley.

Estrada: So when you came to Berkeley had you already decided you were coming to the US? Or did you look at where should I go to graduate to school, decide you wanted to go to UC Berkeley, and that's what got you over here?

Chao: I wanted to come here for a number of reasons, and some of them are very personal reasons. It's just the overall package of the attractiveness of not just Berkeley but coming to the US—you know, the environment—like I said, the whole package. I was looking forward to come here. I applied for many schools. Berkeley was my top choice. I was lucky enough to get in, and so I enjoyed that time there.

Estrada: You got an MS [masters in science] degree from there?

Chao: Yes.

Estrada: Did you go on to do any Ph.D. work?

Chao: No, no, I finished my degree, my Master's degree. It was about the time I tried to interview for jobs, and I was able to get a job offer from National Semiconductor. That was the beginning of my career. I started to get exposure to the early days of EDA. The hiring manager, his name is Paul Huang, it turns out he came from the same college in Taiwan. I think he was in the final stage of getting his Ph.D. from Santa Clara University, and he was considered a pioneer in this so-called EDA area. His Ph.D. study had dealt with physical design in IC. His Ph.D. had to do with how to produce a linear shrink—to come up with a linear shrink program to manage the geometry aspect of physical design of integrated circuits. He was hired at National to run their design rule check program. He started to hire people and I'm probably his first hire to get exposed to EDA, He interviewed me and liked my qualifications, so I ended up working for him. He became a mentor ever since. Later on, we started a company together and so forth. He's a very important person in my career, life.

Estrada: So when you took the job at National, was it specifically to work on this Design Rule Check - **DRC** project (IC design tools which automatically detect and report violations of IC layout against sets of layout rules for manufacturability) or was it more general?

Chao: Actually when I was first hired into National, Paul asked me to look into another part of physical verification. It's called layout versus schematic (LVS). Layout-versus-schematic tools automatically detect and report discrepancies between IC layout (done by IC layout designers) and its corresponding circuit schematics (done by IC circuit design engineers). So Paul was into the design rule checking and he asked me to work out layout versus schematic. Back then there was no such package in the commercial world even in academic domain. I was quite excited about that kind of opportunity. Before I started doing that, my warm-up project at National was to develop a new software capability that takes the IC layout data and rasterize it for new generation of plotter from Versatech. They had just introduced a 72-inch wide plotter that needed software to generate rastorized data from IC layout. I was the first one to do that, and it gave me some warm-up opportunity as a young engineer. After that, I got exposed to design rule checking, but my real project was to investigate layout versus schematic.

Estrada: This was all within National. Back then there were no independent companies trying to solve this problem that you knew of? I'm guessing there must have been other big IDMs [integrated device manufacturers] that were doing similar work inside, no?

Chao: Actually there was independent software company called NCA that offers commercial DRC software for design house like National. National Semiconductor bought the software license from NCA. Back then it was still the time when the compute model was based on IBM mainframe. There was a timeshare company called ISD, also in the Valley. They provided the timeshare service for running software applications on their IBM mainframes. The National Semiconductor design team rented time from ISD to run DRC applications from NCA. From the National standpoint Paul Huang was like “Mr. DRC” [design rule checking], and he ran that for all National designs. That was the model then.

Estrada: …and National at the time was using NCA's design rule checking?

Chao: That's right.

Estrada: …but they didn't offer layout versus schematic (LVS) and that was your project?

Chao: Back then there was no layout versus schematic capability. The design was relatively simple—you're talking tens of thousands of transistors maybe, and they were often repeated. For instance, in memory design, a big part of chip is array of repetitive circuits, called memory cells. You don't check the whole chip. You can check it cell by cell and so forth. So a lot of the checking was manual, but very quickly you find out that you need to have tools to do the checking automatically. I think during my tenure at National a vendor came out with a layout-versus-schematic tool. That vendor was called Phoenix Data Systems. I think they were based on the east coast—in upstate New York. They had some capabilities in this area, so I think Phoenix Data came out with the first commercial LVS offering for design teams. I believe National also bought their software later on. I think my contribution was that I developed much easier to use LVS after I left National, as part of the Dracula package, which ended up being the most successful IC physical design verification package in the industry. The LVS technology I developed was able to pinpoint open and short errors down to the transistor and net level. I think the Phoenix Data Systems capability would basically just tell you there was something wrong with your layout. Imagine you have, even back then, tens of thousands of nets and transistors, so just knowing something was wrong was not that useful. So I think my work became the most practical layout-versus-schematic capability. Basically it would report the opens and shorts down to the net and transistor level, where you really need it. A tool that says pass or no pass is not as useful as a tool which can pinpoint the error.

Estrada: How long were you working on LVS program, and did you work on it within National?

Chao: No, at National I would mostly investigate. National’s design rule package was based on NCA mostly. It's hard for me to develop LVS [layout versus schematic] as a standalone capability. It has to be part of the total physical design verification package. Because there's infrastructure code, such as common database, design rule and command language and other things needed in order to provide LVS. LVS needs to be built on top of a whole design verification tool package. So given that National was based on the NCA (outside vendor) package, I could only investigate LVS. The real development came later when we developed the Dracula package after both Paul and I left National. It was initially developed for our next employer, which was Gould SEL. It was a minicomputer project within the conglomerate called Gould. At Gould SEL, Paul architected Dracula and developed the DRC, while I developed Dracula LVS.

Estrada: How did you and Paul end up moving over to Gould?

Chao: What happened was that Gould SEL actually was a minicomputer vendor— building minicomputers for industrial use, not for the scientific applications. Back then the most famous minicomputer vendor was Digital. So they were competing with each other but in different spaces—with Gould focused on minicomputers market for industrial applications, like factory automation.

Estrada: These were VAX machines?

Chao: Almost, but architecturally they were not like VAX machines. They decided to invest in CPU chipset design. They set up a design team in Silicon Valley, and the head of that design team was Glen Antle. He was hired by SEL to head the design team. Glen hired Paul and me away from National, because he needed to set up his own EDA team alongside with his design team. Paul and I left National around 1980 to join the SEL CPU chipset design team. That's when the original Dracula code was started. Dracula code was started sort of accidentally. Because Gould SEL built their own minicomputer, they mandated that the software needed to run on their computer and the commercial solutions from NCA and Phoenix Data Systems did not run on their computer. Glen tried to negotiate with the vendors and ask them to port their code to the SEL platform, and they did not agree. Given that, there was no choice. So Glen said, "Paul and Ping, can you guys build something on this computer?" We said, "Yeah," and that's how it got started.

The funny thing was that this was accidental. The SEL machine came out about the same time when the Digital VAX machine became very popular in the engineering community. VAX architecture had a virtual memory. SEL did not have a virtual memory, so we were constrained to develop the original software architecture of Dracula under a very limited hardware resource. And that actually turned out to be a good thing. The key performance advantage we had was that we were very, very efficient in terms of the hardware resources—memory and disk and so forth. So that's an interesting twist of Dracula.

Estrada: When you start the development of these programs, was the term electronic design automation in use—EDA?

Chao: No, it wasn't. It was called CAD - computer aided design, and, actually, Dracula wasn't the name either. Dracula was the new marketing name later on when we started our own company ECAD. Like I said, Dracula was originally architected and developed within Gould SEL. Now another interesting twist was that after awhile Gould decided to terminate the CPU chipset project. They stopped the development, and Paul and I and almost everybody else in CPU chipset project team were laid off from Gould. That was when we started a company together with Glen Antle. So, when we started ECAD, we already had the foundation code of Dracula technology. We negotiated with Gould SEL to get the rights to the source code and the perpetually exclusive marketing rights to the technology. Shortly after, we launched the commercial version to the market. Dracula was used as a new name to launch the product

Estrada: When did you officially start ECAD?

Chao: 1982 we started ECAD.

Estrada: It sounded like the three of you, Glen, Paul and you started ECAD. Your thought was there's got to be a market to sell this independent of being tied to hardware and independent of being captive inside of an IDM?

Chao: Yes. By the way when we started ECAD, there were two other founders. Besides Glen, myself and Paul there were another two: one is called Wen-Jai Hsieh and another one is Lap Man Yam. So, total four of us on the technical team developed Dracula. The initial funding was by Gould themselves. They decided they would fund us initially, and as a condition we had to make the software run on their platform.

I think they were looking for some applications on their hardware. We agreed to that, and in return we got the rights including the marketing rights. Then Glen tried to raise some venture capital funds initially and it was not very successful. We then began an initial marketing and sales effort. Very soon, we started to generate income and became profitable. Our business grew very quickly, Big part of that was because Dracula technology had some compelling advantages over the tools from competitors –NCA and Phoenix Data Systems. Once we became profitable, we were able to raise VC funding. So we started in 1982. I think by the time we had some investors on board it was almost 1985.

Estrada: And you were a small company still at that point?

Chao: We were a small company by the time the venture capital was on board. Like I said, we were profitable, we probably had a reasonable sales and marketing effort, all the application engineers, and it was just a matter of developing and growing the market.

Estrada: So you were one of probably a relatively few independent EDA companies at that point?

Chao: Yes. That was an interesting time in EDA, because around early 1980 the EDA industries really started to form based on the advancement in hardware computing technology and growing need in design automation. The initial push was to develop the specialized workstation for schematic capture, simulation, and so forth. That was the first wave of Daisy, Mentor, and Valid. They were based on some interesting EDA application and used specialized hardware to run EDA applications. They had some initial success. ECAD then was perhaps the first pure software company in early EDA industry. I wanted to mention that Gould SEL asked us to develop software on a very weird platform. Later on we found out it was a major advantage for us, not just performance-wise, because we believed the software should run well on a fairly limited computer architecture – common across many different computer platforms. So it was easily portable to the Digital VAX machines, Prime Computers, Apollos, and so forth. We were the first pure software EDA company.

Also, we had the concept of software portability very early on, so we were able to take advantage of that and proliferate very quickly to all the hardware platforms, including Daisy, Mentor, and Valid, then Digital VAX, and all the workstations. Of course later on, quickly toward the second half of the 1980s, the hardware platform was starting to standardize on UNIX workstation, such as Sun Microsystems, and so forth. Similarly, we were able to port Dracula to all the UNIX workstations very quickly. Another interesting player then was SDA. They started out about the same time, but it took them a bit longer to come up with their product. They were into the UNIX workstation based integrated IC physical design solution, which was complementary to what ECAD was doing with the physical design verification. One addressed physical design and one addressed the physical design verification side. Interestingly, the whole industry evolved along these two lines: design and verification. First, it was physical design and verification and then came logical design and verification. Starting with ECAD and SDA, the EDA industry evolved over time to more of a software-centric business model based on UNIX platforms. I think interestingly ECAD perhaps was, if not the first, very early in software only EDA business model, and in software portability.

Estrada: Early on in ECAD what were some of the biggest technical hurdles that you had? Obviously you're developing for constrained hardware, so that was probably one of them. Were there other technical hurdles in terms of algorithms…?

Chao: Well one was portability. Because of portability, you can automatically benefit from the quick advancement on the hardware engine which was happening then. So we would literally take advantage of that very quickly. Customers were switching or buying new hardware and immediately the Dracula and all other ECAD software solution were available for them. That's important. Also software innovation. I think Paul Huang was quite unique to come up with a very novel approach in terms of processing IC layout geometry. At the time the physical design have perhaps represented the biggest challenges. It's a massive amount of data you have to process, so it's a compute intensive application in EDA. You have to deal with a huge amount of data, and so he came out with the so-called trapezoidal data model to represent that physical data, and some band-scan as opposed to line-scan algorithm approach. So there is some fundamental software innovation algorithm wise and data model wise. Dracula algorithm even to this day is still considered quite good after what, how many years? I mean it was a long time. Then later on, Dracula's biggest hurdle was to be able to go into so-called hierarchical mode of processing. During my tenure there and Paul Huang's tenure there we were not into that. We knew it was coming and would become important some time down the road as IC layout continued to get more complex. Sure enough it became Dracula's biggest issue—to be able to process the layout in a hierarchical fashion. As far as flat processing I still believe Dracula was about the most optimal solution after 20 some years. Hierarchical processing takes new innovation. It's something that Dracula did not do well on.

Estrada: Which created the eventual opening…?

Chao: That creates opportunity for the next wave of innovation from others, what is it?

Estrada: Caliber?

Chao: Yeah, Caliber [from Mentor Graphics] and so forth. The hierarchical processing is the one big innovation that did not happen.

Estrada: You're talking about early to mid-1980s with flat verification, and Caliber really started to catch on probably mid-1990s and later 1990s?

Chao: Right, right, yes.

Estrada: But Dracula is still used to this day.

Chao: It still is when it comes to DRC/LVS processing in flat-mode. If you compare flat to flat, it's perhaps still the front runner—or among the front runners, so that's quite amazing.

Estrada: So early in EDA—for context for people that are not familiar with the industry—really what you were starting to do for the first time was use hardware as a basis for creating the next generation of hardware. It started becoming something which you can get a snowballing effect from. As the hardware improved you could create better design tools on that hardware to create yet the next generation of hardware.

Chao: That's right. Yeah, that's quite interesting.

Estrada: You mentioned the parallels between ECAD and SDA. At some point, as the two of us know, they got together. What was the story behind ECAD and SDA getting together?

Chao: ECAD went public in 1987. At the time SDA was the first one to come out with an integrated physical design solution running software on a UNIX-based platform. They were the pioneers doing that. At the same time ECAD had an internal technology that was a similar solution called SINBAD. SDA started to market their tools aggressively and have very good success, particularly with their custom layout tools, and some success with their cell-based tools—place-and-route tools. But most importantly, they introduced integrated design framework. Many customers started to complain that the most design tools did not communicate with each other. SDA was able to come up with a framework notion to address that issue and had a lot of following because of that. If you look at their financials, it was very similar to ECAD—although ECAD went public (IPO) first—in terms of revenue, the number of employees, the growth rate, the profitability, and so forth. The founder of SDA, Jim Solomon was at National when Paul and I were there. We actually also both had a connection to Berkeley and to Richard Newton who was at Berkeley. So, we had some collaboration in ideas and so forth. We knew a bit about their ideas. They had this idea about building a total solution to address physical design and all that, but this is at the same time we were competing. Pretty soon we realized we are going to step into each other's turf, because they were also designing DRC and LVS. Of course that was not their main effort. So we had a lead on the design verification side, and they had the lead on the design creation side. We both believed in the software only EDA business model, the software portability model, and all that. I think also culturally we are quite compatible. We both valued engineering talent and so forth. At the same time we were competing. By the time SDA got ready to go public there was the 1987 stock market crash. SDA had to abort their IPO. So there were all kinds of discussions about merger acquisition, and the bankers, proposals, counter proposals—not just between ECAD and SDA but among others—Mentor, Daisy, and all the others. Mentor was getting big. Eventually ECAD and SDA were the best strategic fit because our visions were much more compatible in terms of software only EDA business model, the UNIX-based solution, and all that. So we ended up working together. Of course a very important factor, I mean I don't want to downplay this, a very important factor was Joe Costello. He was a big factor that attracted the two sides together. We knew we were going to have a young leader and take the business forward. So that's how the two companies came together in 1988 to become Cadence, I believe. The momentum, with the effort of the management team under Joe, it was just unstoppable at the time. Macro-environment also favored Cadence, because there's a strong drive toward the UNIX-based workstation solution. That put the big three—Daisy, Mentor, and Valid—with their proprietary hardware platform, on the defensive. Cadence [the new company that resulted from the ECAD and SDA merger] cashed in on that very, very aggressively.

Estrada: What was your role in ECAD when ECAD went public?

Chao: Actually a couple of years right before the ECAD IPO, I started to drive and run this new effort, which was to develop a second product line for ECAD. It is also the competing product to SDA. The product was called SINBAD, which was the integrated physical design, place-and-route, all that kind of stuff. I was involved in that.

Estrada: How did you feel about joining forces with someone that you were trying to replace?

Chao: I was fine. What happened was—and this is why the merger worked well—when SDA and ECAD came together there was a sincere effort to integrate not just people, which is most important, and integrate technology. The surviving technology actually adopted much of SINBAD's technology. For instance the symbolic design compaction algorithms, but we did obviously use the more mature framework technology from SDA as a base. If you look, many applications actually came from SINBAD. That was part of the merger process, that people and the technology were fairly evaluated and integrated together. That was quite good. So I had no problem at all.

Estrada: Some of the technologies on the implementation side were DF2 [design framework 2], CDBA [Cadence database], Virtuoso, and Composer, all of which are still the mainstays in the industry today?

Chao: Yes, exactly.

Estrada: So 20 years…

Chao: Some 20 years, that's right, that's right.

Estrada: So one other question on this area and then I want to move forward with your career: You mentioned Joe Costello being pivotal here. How did he enter the picture in terms of ECAD and SDA?

Chao: Joe was a rising star in SDA right before the merger. So you know, he was like the heir apparent at the time already to Jim Solomon. We knew as part of this merger discussion, that he was going to be the CEO. Technically Glen Antle was the CEO of Cadence in the first couple of months after merger, something like that, but Glen was fully supportive. I would say this guy was going to lead the joined company together. Both sides and the employees were totally supportive. Like I said, he was a factor because he was convincing enough. I give credit to Joe. EDA ended up as industry that could be looked at as an example of how M&A [mergers and acquisitions] work. It is commonplace in every industry now, but back then perhaps the most active M&A was in EDA. Joe was able to drive that process with many others' help. We thought this was a very successful merger and later on a couple more successful ones followed. That's what built Cadence. Then it became a model for the EDA industry and in many other high technology industries.

Estrada: Alright, so now we have Cadence formed and off to conquer the world with unified design and verification—I believe on an independent UNIX platform. They were really competing with, at the time, Daisy and Mentor…

Chao: Daisy and Mentor were still around. They started basically weakening.

Estrada: …especially since a lot of their business was based on hardware as part of the revenue stream, and that was dying.

Chao: That's right.

Chao: Yes, exactly. That's a major factor. In my mind there are three important components to technical advancements in EDA. One is computation and hardware technology advancement. It either works for you or works against you. That's a very important factor—the overall computing environment and so forth. Next obviously is software innovation from time to time. The third component is the advancement over time about the design methodologies. For instance, hierarchical design methodology is an important one. There are others, like starting to go into high-level design as a methodology, which also is an important factor in how EDA technology evolved. At the time, it was pretty obvious that proprietary hardware was becoming a strategic problem for some of the EDA players.

Estrada: So Cadence is formed. How long did you end up staying with Cadence?

Chao: Cadence was formed and I stayed on after the merger for about two years. After the merger Paul Huang was basically the head of the engineering in the combined company. It took about two years to get the integration totally in place, people-wise, technology-wise, and customer base transition-wise, all that. I think I'm Paul's right-hand person to manage that effort. So after that, we say what's our next dream? Paul and I left about the same time to get into our next venture—about the end of 1989 or the beginning of 1990.

Estrada: From a methodology standpoint, there was a cusp just starting to come out there with Gateway and its high level functional simulator Verilog coming into Cadence. And then came Synopsys and some rivals for synthesis.

Chao: That's right. So that whole thing happened about the time we left Cadence.

Estrada: So what did you guys have in your head about what to do next or did you already know?

Chao: First of all, my philosophy personally, is that we don't want to do something competing with our previous venture or employer. So we've got to do something different. We looked at what kind of challenge and opportunity was presented by Moore's Law and, therefore so-called design productivity gap. That gap basically describes the widening gap between our ability to design and the full potential provided by continuous advancement of process technology. Ability to design includes both design creation and design verification. Particularly, its been a big challenge on the design verification side. It seems like people can create design. But unless you can verify it, the design is useless. So that prompted us to look at that space. At the same time also, we say, "we've been religious about pure software from the beginning. It's software. It's software." It's sort of fun to challenge ourselves and maybe there's some hardware play in EDA. Nothing can always stay constant. Maybe there's some hardware solution out there. We know at the time, given limitation of UNIX workstation's performance, there were just a lot of things you cannot do with it. So we decided to look into then rapidly improving FPGA [Field-programmable gate array] hardware technology. We said, "Maybe we can use FPGA hardware technology, which is programmable, to address this design verification problem with potentially one million time speed up compared to traditional simulation." That's the beginning of this so-called Logic Emulation technology in EDA. The FPGA then could give you limited capacity. Perhaps equivalent of

1,000 logic gates per FPGA chip. So you need to put a large array of FPGA chips together to give you sufficient capacity to emulate a state of art ASIC [Application specific integrated circuit] design.

Estrada: At this point, you're talking about emulating a design more at a logic level, not at a physical level, which you had done with Dracula.

Chao: Exactly. So this new venture sort of moved from the physical design EDA (from our first venture) to the logical design EDA. We also swung from the software EDA business model to the hardware EDA business model. So we say, "Let's have fun with it and see what happens." It was, indeed, very challenging.

Estrada: Did you consider other potential areas at the time that you gave serious consideration and then picked this? Or this was something that you thought of and thought this is what we want to do?

Chao: Actually, no. First we thought, "Let's go there and look." We had another founder, who had actually also previously worked on Cadence on the physical design tool. His name is Nang-Ping Chen, Dr. Chen. To address technical challenge of putting together a large array of FPGAs, Dr. Chen came up with this crossbar architecture. What was the best partition architecture that gives the maximum efficiency in terms of utilization of the gates and to manage the timing clock—clock skew problems, hold time problems. We thought we had come up with something pretty smart. We also believed it was scalable. Okay, this we know. We also believed that FPGA technology was going to improve very quickly. So it's important that you're not stuck with old generation of FPGAs. We said, "This architecture is going to be scalable to future higher density FPGAs." And of course, it is going to be portable to any new FPGA family from any vendors. So we said, "Okay, regardless of how hardware is going to change, what's the solution software-wise and architecture-wise?" So that's how it got started. This time, though, we did get funding. We did have a VC on board and all the right startup setup—a more classical setup with advisors on board and all that.

Estrada: Your earlier success in ECAD as well as some of the other successes laid the foundation for the venture community to be able to trust that there would be an outcome…

Chao: That's right.

Estrada: …and you guys had a track record by then. "Give money to people that don't need the money", right? You were in a better position all around.

Chao: That's right. Exactly. That's what happened to the initial setup of this thing. We were able to recruit some very good people and had all the checkpoints checked in terms of a reasonable startup at the time. It turns out it was a different experience than ECAD altogether, but that's how things happened. You probably know pretty soon we ran into competition, which was Quickturn.

Estrada: Just to clarify, the name of this company was--

Chao: It was called PiE Design Systems.

Estrada: And the name comes from?

Chao: If I remember right—usually I work on the name for things like this, like I chose the name ECAD. It's kind of a fun thing. Paul and I often times chat about his philosophy and so forth. Everybody knew about this even at Cadence when Paul was there. His philosophy was the job in high tech was that we need to continue to polish. That's what the P stands for. After a while, he said you need to break out of the pattern and innovate. It's like a step function. The step function is to innovate. You need to then capitalize on the innovation and do all the polishing. So he has a step function. So P and I stands for polish, innovate. Keep on going in order to grow. Eventually you will enjoy. That's what P-I-E stands for.

Estrada: Polish, innovate, enjoy.

Chao: That's what PiE stands for. That's the second venture. So we said "okay, we do it all over again, P and I." Anyway, pretty soon we ran into serious competition with Quickturn. They also were well funded. They were actually in the marketplace already about a year or so ahead of us.

Estrada: Did you know of them when you started the company? Or you didn't know of them?

Chao: We knew them. We knew that they had more of a first generation of system architecture—not crossbar. So we think, "Ha! We are smarter. We have a crossbar solution." We really looked it up inside out and analyzed it and said, "This is such a beauty. This is going to absolutely wipe them out," not knowing that they also came up with something new and similar which was based on patented technology Quickturn just acquired from Mentor Graphics. Actually, independently Mentor Graphics had a senior guy called Mike Butts. He was their fellow or something and he had studied this thing for sometime. He came up with a good idea. Mentor Graphics didn't build one, but they patented it instead. So we actually were totally surprised when Quickturn filed patent lawsuit against PiE, right after they acquired patent from Mentor. We thought we had something really original. We did not bother to patent the thing early enough. But we think we've got something interesting, a very elegant engineering solution. At that time, we both got customer traction, both sides. We had our customers and they had their customers. For instance, they had Sun Microsystems, we had Motorola, and so forth. It was just fighting in the marketplace and in the courtroom at the same time. We don't know what was going to happen. But at the time, Phil Kaufman was their CEO—the famous Phil Kaufman of the Kaufman Award. It came about that Kaufman had a heart attack or something. So he died in the middle of this, so Quickturn hired a new CEO. I think that could have been the event that sort of broke the ice. Then both sides started to look at this on a more rational level instead of an emotional level. Later on, the two boards kind of got a chance to look at this and say, "Let's try to work together." That's how the two companies merged.

Estrada: Who was the CEO? Was that Keith Lobo or was that before Keith?

Chao: Keith Lobo, yeah.

Estrada: And your CEO at the time was?

Chao: Was Glen.

Estrada: So Glen came and joined you?

Chao: Glen joined us when lawsuit happened. Initially Paul was CEO, and once the lawsuit started, he basically focused on the lawsuit and engineering, so Glen would run the business.

Estrada: What were you doing at the time?

Chao: I was actually running marketing.

Estrada: You were marketing?

Chao: At PiE.

Estrada: You said you founded it in 1990. How long until you had the product out the door, and when did the merger with Quickturn happen?

Chao: You know, I'll have to guess. I guess about 1992, the product. The merger happened about the end of 1993 or something. Quickturn went public in 1995.

Estrada: You were a combined company before they went public?

Chao: Yeah.

Estrada: At the time you were taking in designs, so I mentioned a little bit ago about the next wave of methodology coming out with high-level design—RTL-based design. Originally, were you taking gate level designs then?

Chao: Yes, it was taking gate level designs. Logic emulation basically emulates the functionality of a design, but it had to work correctly in terms of functional timing. To emulate a design, managing hold time was the issue. You can only look at the hold timing at gate level to be accurate. You had to look at every clock cycle and all the functions that have to happen within the clock cycle. So Logic Emulation was based on gate level. However, the industry is clearly moving to high-level design – RTL-based design methodology. Later on I actually ran the project to do the so-called RTL-level of emulation. What it does is to actually provide the RTL level front-end to logic emulation. To debug your design using logic emulation, you have the RTL front-end to help you do it. But underneath, it's gate-level implementation.

Estrada: It was a merger between Quickturn and PiE?

Chao: Yeah.

Estrada: What was your role in the combined company and how long did you stay with them?

Chao: I ran two projects at post-merger Quickturn. Logic Emulation business had been a very high-end EDA business for design projects like high-end microprocessors and graphics chips. Its hardware costs millions of dollars. That basically narrows the market appeal. Besides high hardware cost, another major challenge is ease-of-use of logic emulation. To be able to compile the design [for emulation] quickly and easily, to be able to just open up the box and use the product was almost never heard of. I started an effort to look at addressing these two key issues. One was to look at the lower cost, more affordable system. Another one was the new emulation environment for high-level design and debugging for better ease of use. So we tried to see if we could make it more usable and hopefully more affordable to the mainstream design project team. The projects resulted in limited success. The idea to push logic emulation for adoption by mainstream design team was never quite realized, even to this day. Basically, logic emulation technology has strong appeal only to a certain high-end design market segment, a certain niche. Even today it’s true. Anyway, I stayed on through early 1996.

Estrada: You were running development projects for those two?

Chao: Yes. I’m sort of the general manager for that. So in the beginning of 1996, I first took some break and sort of “retired”, before I started my third venture - Silicon Perspective.

Estrada: So you leave Quickturn in 1996 and you’re going to take a little break. What got you out of that mode? Did you know you were going to go back in and do yet another EDA company?

Chao: Yeah. I kind of knew. It’s like pendulum—swing between software based venture and hardware based venture. This time I said, “I’m going to swing back to software business. I miss the software.”

Estrada: It’s got to be easier than hardware?

Chao: It’s got to be easier. That Quickturn experience is very, very unique. I learned how unpredictable things could turn out to be from the original idea. It’s quite unpredictable in terms of how difficult the technology turned out to be. How costly it is. It has a certain market appeal, but its limited market appeal. The effort of making logic emulation work is overwhelming - for engineering, for sales, for marketing, for customers…it’s just overwhelming.

Estrada: So you swung back to the software side and you also jumped from verification over to the design side, right?

Chao: That's right. Just like semiconductor industry, EDA has been driven by Moore's Law – continuous advancement in process technology and ever increasing design complexity. Early days of EDA supported a relatively simple design methodology with two areas of focus: design and verification. Later on, EDA evolved to support a more refined design methodology, like a two-by-two matrix: logic design and verification, plus physical design and verification, each of which requires significant effort. Typically these four types of design tasks were carried out independently by different design team members but guided by a common chip level architectural plan. Around 1996, just about the time I left Quickturn, the IC design and EDA community faced a new crisis that the traditional two-by-two design methodology can no longer address the new design challenges. This was because the new process technology had advanced to a point that physical interconnect [on chip electronic wires which had approached 10^{-7} meter thin and getting thinner] became a dominating factor for chip performance (speed and power). The actual physical interconnect can only be determined during layout (physical design) phase. Traditional design methodology which used an over-simplified model to "predict" interconnect during the logic design phase was simply too inaccurate. And IC design became much harder to complete for final sign-off. This was the famous Logical-Physical design disconnect (or design closure) problem. At about the same time, another wave of design methodology change is happening, that the design needs to be hierarchical. With this backdrop in EDA industry, my third venture, Silicon Perspective was about pushing the next generation of design methodology which would "bridge" logical and physical design disconnect and support a hierarchical design solution. That's what the Silicon Perspective venture was all about and it's a swing back to software again.

Estrada: Logical design really got started in the late 1980s or early 1990s. What was pulling them was just the sheer complexity of the amount of functionality and making sure that was all right. Meanwhile, the silicon was getting smaller and smaller in process geometries, so new physical effects were pulling the people that have to make sure that the physical layout was right. It was pulling in the opposite direction, so you needed to bridge the gap between those two worlds.

Chao: Right. I think the market condition was just right to address "design closure" crisis. That's a compelling market requirement. Equally important, I think we tried to build a solution which was very easy to use. EDA always had a bad name for its hard-to-use technology for many, many years. You go to every customer that ease of use is always among the top complaints. EDA becomes a very niche science almost. Because the perception of EDA tools has always been so hard to use, so if you make it easy, make it lightweight and fast turnaround, it's got a lot of appeal to the market. So again, we addressed the critical problem in the market with an easy-to-use package. I think we got a good recipe.

Estrada: What was the genesis of Silicon Perspective itself? Did you go out and decide "I'm going to start a company for this"?

Chao: Actually, this time around, based on my previous experience at PiE, I decided not to start out with a plan and then simply stick to that plan. My experience told me that the plan will change. If we were building the plan too early, there would be too many unnecessary constraints, so it probably will be bad. I'd rather be flexible. I didn't take venture capital money until almost two years later. I think you have to have a pretty solid idea before you take serious investors' money. I didn't want to do that. So at early stage of Silicon Perspective, I'd have a fair amount of flexibility to decide what to do. Actually, the business plan was nonexistent, to be honest with you. I decided to first hire some good people—very talented engineers to begin with—and try different ideas and build prototypes. Then we would take that prototype to test it with the potential customers and see how they rate it and test that way.

To be honest with you, what we initially built had totally nothing to do with the final product. We built a cycle-accurate simulator. That was the original idea. In 1996, we did not actually build this thing aiming at logical-physical design closure. You can see how different it ended up. Later on, we tried this and tried that and talked to some of the design teams in the valley. We also picked up the buzzwords in the marketplace. Design closure was becoming a real big problem. We talked to many, many people, and we understood what others were already trying to accomplish. So we tried to steer clear of similar approaches or strategies. That's why we came up with this [Silicon Perspective's product]. Not that we had a plan first and just do it. We did actually modify the prototype quite a bit along the way. But we said we wanted, no matter what, to make it an very easy–to-use software solution. We believed that if you can build an easy-to-use, lightweight solution, then no matter what it has some appeal to IC designers. How big it could grow depends on how big the market demand is. But it turns out that functionality for bridging logical and physical design is a major requirement. So that market for the Silicon Perspective product grew to be reasonably big. There is a lot of functionality in EDA. That means there are always pockets of opportunities for easier-to-use solutions. That, we said, was almost a golden rule. With luck and the right market conditions, that opportunity may grow to be much bigger.

Estrada: Who were some of the key people early on with you?

Chao: The primary technical person is Mitch Igusa and then Wei-Jin Dai.

Estrada: You had done two previous companies with Paul Huang, well actually three, if you go back to National. Did he have any involvement?

Chao: No. In Silicon Perspective he was a passive investor. He was my board member and a passive investor. He was off to something else. He was no longer doing EDA, but he remained my mentor.

Estrada: This is the first one where you're really the sole leader?

Chao: Yes, this one is mostly my working with Wei-Jin, Mitch and the team. I got to know Wei-Jin, and obviously he was very key to SPC's [Silicon Perspective Corporation] success. I knew him from Cadence/Quickturn days. He was actually on the competing side. I was on the PiE side; he was on the Quickturn side. We were competing with each other. We knew each other from there. I knew Mitch, by the way, from back in Cadence. You know, it's a small industry. I feel that there are many very talented people in the industry, but there are many problems to be solved. To find a good project actually is not easy. There are always endless problems to be solved in EDA, but my definition of a good project is that it will have the potential to have big enough commercial success and somehow the process has to be fun. That's not too easy to find.

Estrada: So you found it again with Silicon Perspective?

Chao: I liked the Silicon Perspective experience particularly. I think it had both a commercial benefit and the fun aspect. Quickturn, I think it was okay. It was not the best project. It's very challenging, don't get me wrong. There were talented people and it was very challenging to solve a tough problem. EDA turned out to be like this: it's fundamentally solving tough problems, but good projects are here and there. It's not always there.

Estrada: With Silicon Perspectives you tried a few things. You ended up with the product called "First Encounter." It became clear over the course of time that this was really the best way to solve the problem.

Chao: That's right.

Estrada: How long did that take?

Chao: I think it happened fairly quickly. When Silicon Perspectives first came out, about the same time there were quite a few other startups coming out to also address this problem - timing enclosure and design closure from a different angle… The best known ones included Magma, and Monterrey Design was another example. Back then we didn't have a lot of marketing effort. I remember that Magma had some kind of PR event and they invited a bunch of people—the users, the press and so forth. I heard at Magma party somehow they were talking about Silicon Perspectives, so I knew we had something good going.

Estrada: When was that?

Chao: It was 1998. I knew we had something, because we would see a lot of exciting customer feedback and communications and referrals. I felt that somehow we struck something, because that's the best way you can validate it's there. In EDA, if you've got something that could really solve designer's problem, you know they would talk about it. The word of mouth is the most powerful marketing in EDA.

Estrada: So the big companies—Synopsys, Cadence, Mentor—at the time they were trying to solve the same problems. Magma was in there. There were several other little companies. At the time, what were you thinking about where you take Silicon Perspectives? What's the natural next step with the company?

Chao: Even at Silicon Perspective, we were looking at the roadmap based on First Encounter as an anchor point to grow to a complete suite of solution based on new generation of EDA technology from there. For instance, we would need to have a router. We would need to have a synthesis, and other things. One way to go from there, we could try—and I did try to team up with other startup company. For instance, to approach the company with up and coming best-in-class routing technology, NanoRoute and others. They were a private company, and it's pretty tricky to get them on board. They knew Silicon Perspective had the momentum, and they were still in the developing stage and trying to get to the commercial stage. We knew the window of opportunity would be like within one year to 18 months to get a complete suite of next generation solution out there. The only other person who had it was perhaps Magma. The big guys didn't really have it. Cadence didn't have it and Synopsys didn't have it. So we thought okay. The other option is we go join one of big guys to make this happen. My real assessment was that it was a lot easier to make it happen within Cadence than within Synopsys. We looked at all the considerations—financial, the state of EDA industry and companies, the potential, and all of that. That's why we ended up with Cadence. And after we merged with Cadence, we started doing all these M&As from Cadence. And the rest is history.

Estrada: So you joined Cadence again, and you knew there was a new group of EDA startups out there that looked like they had promising technology. So as part of Cadence you were able to then go out and get Cadence to acquire that technology and lead the integration there?

Chao: Yes, exactly. Cadence had the currency and it made it much more workable. Actually the window of opportunity—I thought it was about one year to 18 months—so we would quickly move to get a router, the best router, and then the best equivalence checker, and the best synthesis technology, and that's what happened. I think that built a foundation for a new generation of Cadence digital solutions very quickly. So that was good. That was a good experience. I appreciated Cadence's support, particularly Ray Bingham's support to make it happen, because his support was important. We did it in such a way that it was sort of "unCadence." We moved very quickly and tried to run it almost like a startup at the time.

Estrada: So there was a legacy product line there that was a big part of Cadence that you came into and then brought these other companies and their technologies in. You put together a solution that basically replaced a big part of what they had going and was clearly kind of tired in the marketplace and not doing very well. So you basically brought a new generation to the company.

Chao: That's right. That's another part of the transition if you will. I mean any integration of new technology—and even involved some old stuff for that matter—I think the best way to do that is to do it quickly. Quickly is tricky because you have to work with customers to see how that'll happen, but you have to offer sufficient benefit so they would go along with you. A lot of this is managing your own team to get that going quickly. As it turns out, that worked out well.

Estrada: The Cadence acquisition was what year?

Chao: What do you mean?

Estrada: Cadence acquired Silicon Perspective in what year?

Chao: December 2000.

Estrada: You did some other acquisitions and then integrated that. How long did that take?

Chao: I think about 18 months to get there and going. Again, it took Cadence to do it too. You know, there's an ability—a capacity to execute that, and there are a lot of details—not just from making the deal happen standpoint. I think what I realized is that even Cadence could do it. I mean a big company supposedly moves slower, but it can happen. So that's good.

Estrada: So you ran that effort at Cadence for a couple of years and then?

Chao: Until 2004 or something like that. I don't know how long was that period of time, then I was running this so-called design and verification [group].

Estrada: At Cadence?

Chao: Right, when we worked together.

Estrada: Then you decided time for a real retirement?

Chao: I decided, time for a real retirement. That's right, exactly.

Estrada: So you've had no or little involvement in EDA since?

Chao: Oh yeah, a little, a little. Like I said, I think it's hard to find this kind of interesting project now. There's always opportunity, but a project that I think will be both rewarding and fun—it's now so hard to come by. I think industry still has time, a couple of years, to digest the technology. When you and I were at Cadence, between 2000 and 2005, there was a lot of new technology introduced. I think it will take a couple more years to digest all that new technology. I am interested in looking at new frontier in EDA if you will. So I've got some involvement in the so-called DFM space.

Estrada: DFM—design for manufacturability.

Chao: Yeah. You know, process variation and that kind of stuff. Statistical stuff.

Estrada: When you look back over your career, are there any lessons learned or things you would have done differently?

Chao: No, I think it's just experience. It's got good days and bad days. It's okay. I mean I take the package and I'm basically pretty lucky to have that experience.

Estrada: And now I understand you've spent some time back in Taipei?

Chao: Yeah, yeah, I spend a little time in non-profit. This is interesting also, very challenging. It's different stuff, because one of the things which is important and why we're in industry is it's always driven by profit, literally, which is a very powerful kind of driver, and nonprofit is no profit, so…

Estrada: You have to find another driver.

Chao: It's very different, very challenging. So it's like a totally different world in my mind. Some aspects of it is similar, you look at different proposals, find out the management team, people, the process, and leadership…that aspect is somewhat similar. But by and large, it's not as structured. In the EDA domain it's quite structured. We know pretty much the structure, what to expect, and all that. So it's a different experience. It's enjoyable. I like that.

Estrada: Well, I want to thank you for the time for the interview here and for all your contributions to EDA, which in the context of the Computer and History Museum enable new generations of computer architectures and capabilities that affect everyone's life every day. Thank you.

Chao: Thank you.

END OF INTERVIEW

