Stanford eCorner

A Historical Perspective on Semiconductors and Moore's Law (Entire Talk)

Craig Barrett, ***Intel Corporation (Formerly)***

October 21, 2009

Video URL: http://ecorner.stanford.edu/videos/2331/A-Historical-Perspective-on-Semiconductors-and-Moores-Law-Entire-Talk

Intel Corporation legend, former CEO, and Chairman of the Board Craig Barrett discusses his personal career path from a Stanford Associate Professor, to Silicon Valley consultant, to a 35-year career inside one of the globe's most prominent players in technology. His talk concentrates on Moore's Law and the myriad factors in place to ensure its continued progeny.

Transcript

We're honored to have Craig Barrett of Intel here today as our speaker. For 40 years, Intel has grown from a startup that helped put the word "silicon" in "Silicon Valley". So, now it's a household name with its products in over half the households in the entire world. Craig Barrett has been part of Intel for 35 of those 40 years. And I'm proud to say that Craig started his career here at Stanford where he got his bachelor's, master's and PhD in material science. After he graduated, he stayed at Stanford for another 10 years, we couldn't get him to leave, as part of the material science faculty until 1974 when he joined Intel. He became an Intel vice-president in 1984, a senior VP in 1987, an executive VP in 1990, elected to Intel Board of Directors in 1992 and named the company's chief operating officer in 1993 and became its fourth president in its history in 1997 and then CEO in 1998 and chairman of the board in 2005; just an illustrious career for a Stanford alumni. Craig retired from the board in May 2009. And with that, I give you Craig Barrett. It's indeed a pleasure to be back here.

I showed up on this campus 52 years ago as a freshman. And I can tell you a few characteristics about the university at that time. One is there were still trolley tracks from downtown Palo Alto up Palm Drive to the campus. You could drive your car and park it anywhere without a parking sticker at that time. We used to joke and it's probably still true today that you can tell the difference between the faculty cars and the student cars. Student cars were more expensive than the faculty cars. And perhaps even more significantly, the tuition was $250 a quarter when I started. I understand it has increased slightly over this time. I ended up at Stanford a little bit by mistake. My life's desire was to be a forest ranger.

As I grew up in high school, I wanted to be a forest ranger. I only applied to a couple of schools. I applied to Stanford quite frankly on a lark because I knew I couldn't afford to go to Stanford. But I was accepted. Stanford doesn't have a forestry department, as you may know, so I had to major in something else. And so, I chose engineering as it's close to forestry as I could get to at the time. But I didn't know the major in the engineering disciplines. So, I asked a good friend who was in my high school lab class at Carlmont High School, which is just 6 or 7 miles up the river, what he was majoring. And I know he was going to San Jose State and was going to be an engineer. And he said, "Metallurgical engineering." And I said, "That's really cool, Tom.

What's metallurgical engineering?" And he didn't know either. So, my next obvious question was, "How do you spell it? Is that one L or two L's?"And he didn't know that either. But I did choose to write down "metallurgical engineering" on my application to Stanford with two L's, which is the proper spelling. International Nickel Company had a scholarship for someone who chose to major in metallurgical engineering. I was probably the only person in the history of Stanford that said, "I want to major in metallurgical engineering." I got the scholarship so I could pay the $250 of tuition a quarter and come here and then that led to other things. So, that was the first really serious decision I made in my career was how to spell "metallurgical engineering". I came here and I majored in metallurgical engineering. And it turns out that this was near the Sputnik timeframe and every metallurgical engineering department in the world changed its name to material science. So, I actually majored in material science and the pre-decision of the material science in engineering department that keeps us here today, the other MSE department in school of engineering. Unique acronyms for your department names, we wouldn't allow that in a place like Intel.

The second decision that really impacted my career was I actually fell in love with material science. I fell in love with the concept that you could figure out why materials had the properties that they have. Why is steel sometimes soft? Why is it sometimes hard? Why is it sometimes brittle? What gives properties, electronic properties, magnetic properties, optical properties to materials? How do you relate the microstructure to the physical properties of materials? I was intrigued by this. Probably the thing that really got me going in this area was one day that I had an old 1949 Ford convertible. And the owner of this car prior to myself had modified it such that to get in the trunk, there wasn't a key latch. And I think what he got there was a little solenoid at the bottom of the car, you had to press and activate this solenoid and open the trunk and you could get into it. That thing ceased to function and it was the fact that the spring didn't have the right properties inside. I'd be really smart and I got out my little propane torch and I heated this copper alloy spring-up. And I can tell you, it ceased to become a spring very quickly. And I wondered why that had happened in my whole life.

And then I got into material science and was able to figure it out. But I majored in material science, fell in love with it. I had the opportunity to get three degrees here then go to Europe on a NATO postdoc fellowship then come back here and teach for 10 years which was a magnificent time. And when I came back to teaching, and this was 1965 now to '74, something else was happening here and that was, Silicon Valley was being created. It was a semiconductor industry which at that time had worldwide sales of about $1 billion. And so, it was relatively small but it was centered to a large extent here in the valley, Fairchild, National, a few other companies were growing. And one of the things I learned very quickly was in fact, material science was at the core of all of these semiconductor processing and semiconductor technology. It was really microstructure, complex structures you made in creating transistors and properties of materials. So, I did a little bit of consulting with some of these companies. And I can remember spending many long hours at Peterson Building.

It must be over in that direction somewhere, in the x-ray lab there and doing x-ray topography, which is a very simple imaging technique to image defects in silicon wafers and you can map out the entire silicon wafer. People were interested in the introduction of defects into their processing technology and function of thermal cycles and chemicals. I did quite a bit of consulting and met a lot of people in the industry. And one thing led to another and one day someone called me up from Intel, one of the gentlemen who I had met at Fairchild. He was looking for a ceramic engineer which I was not, but we also had no ceramic engineers in the material science department, still pretty much a metallurgical engineering department. And I said, "Well, we don't have any of those that you're looking for, Jean, but how about a frustrated associate professor who would like to do something else with his life than teach?" because my interest had migrated more towards application of technology. So, I ended up at Intel on a leave of absence and ended up there for 35 years. And there are two things that occurred to me at Intel. One was a continuation of the observation I had made when I was at Stanford consulting for the semiconductor industry, which was that material science was at the basis of all of the things going on with semiconductors, the manufacturing, the technology, printing, smaller transistors or lithography, etc. The other thing that I found quite intriguing was in fact, there was this thing called "Moore's Law".

How many of you have heard of Moore's Law? How many of you read the original article in 1965 on Electronics Magazine. That's what I thought; oh - one. Anyway, Gordon Moore, founder of Intel, but he really had written the paper about Moore's Law when he was at Fairchild, was really the head of the R&Deffort there. He just noticed that integrated circuits started in about 1960; it was now 1965. Five years later, you notice a certain trend. And if you follow the number of transistors per integrated circuit over a five- or six-year period, it will look like it was a log, a plot or a linear plot and a log scale. So, it meant that you were doubling every 18 months or so in terms of transistors and transistors related into functionality performance of a bunch of other things. If you read the article, it says very simply, "Hey, I've made this observation. It can possibly last but it's interesting that it has gone on so far and maybe it can last for another five years or so. But nothing continues to double for very long in nature." And I then had the pleasure when I was working in Intel of interacting with Gordon Moore for 35 years.

And every year I would ask him about Moore's Law, and every year he would say it's going to die soon. And we are now 40-some years later and its' still going. And it's pretty cool that it looks like it's going to go for another 15 years or so. So, Moore's Law will probably end up having 60 years or so of history before we have to change that basic transistor structure. It will change; there's no question about that, just as vacuum tubes died when transistors came along. Transistors will die when the next electronic switch comes along. We just don't know what it's like. You're doing a lot of research here at the university and that's a great entrepreneurial effort. Something will have to happen. But Moore's Law is very interesting.

And the pursuit of Moore's Law has been interesting for the last 35 years. I have to take you back to the founding of the semiconductor industry, and it turns out it was founded by a bunch of chemists and physicists and electrical engineers. And I don't want to demean any of those categories of people because they're probably represented here in the audience to some degree. But I can tell you almost unequivocally that most of the folks that I dealt with in the early days of the semiconductor industry had no idea of material science, had no idea of the relationship of microstructured properties and what could happen. So, if you happened to be a material scientist and you knew about dislocations or vacancies and diffusion and equilibrium and all these cool things that we studied in material science, you could be a superstar in the semiconductor industry. Besides that, I got into writing a text book when I was here, "The Principles of Engineering Materials". And so, I got to go to Intel and they thought that I knew something. I didn't know very much but I knew at least a little bit about material science. And so, it was a really fun time. Now, what's Moore's Law and what does it do? It says you double something every 18 months or so.

You double the number of transistors in an integrated circuit. When I went to Intel in 1973, '74, the state-of-the-art circuit was something called "1103", which is a 1K dynamic RAM, 1000-bit. And each bit was basically a capacitor. So, it had about 1000 capacitors and a little decode circuitry, so roughly 1000 transistors or so. And today, we have a couple of billion transistors. So, we have increased the functionality quite a bit. And as I said, you get another 15 years or another six or seven doublings and you can take yourself up well beyond a billion transistors or so functionality. And along with the functionality, you got all sorts of devices. You got dynamic RAMs and static RAMs and CMOS clock chips and CCD-type devices which are really imaging-type devices. And then, you got microcontrollers and microprocessors and all the analog circuitries; an incredible functionality increase over that time period.

But also pretty interesting along that time period is if you scale transistors down, everything good happens. They get faster. They consume less power. You can pack more in them. You get more functionality. So, it's really a self-fulfilling cycle. The other interesting thing is when I joined the industry, the manufacturing plants that we had were considered to be fairly sophisticated. They cost $2 million. Today, our manufacturing plants are pretty sophisticated and they cost several billion dollars. So, we've scaled up the cost of the plant very substantially over that time period, roughly a thousandfold increase in the cost of the manufacturing plant.

But if you look at the underlying cost aspects of this technology, the cost to create a square centimeter of silicon in 1970 is about the same as it cost to create a square centimeter of silicon in 2009. It's a couple of bucks per square centimeter. So 1103, that 1K dynamic RAM I mentioned, they were producing in volume at Intel when I joined it, would sell for $3 or $4. And today, you can make a microprocessor, microcontroller with maybe a hundred or several hundred billion transistors to sell for exactly the same amount. Now, how did that happen? How do you get a thousandfold increase and the cost of the plant and the cost of the output stays the same at absolute dollar? It's pretty easy. First of all, you went from very small wafers to very big wafers. So, you got roughly a ten-fold or fifteen-fold increase in the area that you process in each wafer. The yields went up by a factor of 5 or 10. The throughput or the size of the factory went up by a factor of 10 or so. So, you get three factors of 10, you get a thousandfold increase out of a plant that costs a thousand times as much.

And it translates to absolutely antiinflationary static cost. If you want to relate the cost of processed silicon to something, it's roughly the same cost as real estate in Tokyo. You calculate the end of the square foot of real estate in Tokyo, that's about what a square foot of processed silicon costs you. And that was really when Tokyo was at its high point from a real estate value; it's probably down a little bit. But it's quite an amazing feature that you can have that same cost for that same unit area of silicon. And every 18 months or so, you're able to cram twice as many transistors into it and get twice the functionality into it. And we have now been doing this for 40 years or so. That has been quite an accomplishment, I think, for the industry. Now, I mentioned before that when I joined Intel it was populated by a lot of physicists and chemists and engineers in general. And they were great technologists and they invented all sort of new circuits, new devices.

But frankly, we in the initial stages of the company were not very good at manufacturing devices. We were great technologists and we bring new circuits into the marketplace. And we'd make a few of them and we make some money on the first few that we sold. Then, other people would come along and manufacture them and give us stiff competition. And the only way we were able to survive is continue to go to the next generation of devices, the next generation of technology. And the Japanese-based companies were especially competitive in this respect. They had really learned how to manufacture in volume and were very cost effective and high quality in their manufacturing capability. In fact, 25 years ago in the mid-1980s, the timeframe when the Japanese-based companies NEC, Toshiba, Fujitsu and others were really, if I can use an indelicate phrase, "kicking butt". They were really taking it to the American manufacturers. And companies like Intel were in serious trouble at that time.

In fact, I can remember very specifically a time frame when many noted academics, I don't think any here at Stanford but certainly a lot of Harvard and on the East Coast, were telling us at Intel, "You guys are crazy to be in the manufacturing business. You ought to just design circuits and leave the manufacturing to someone else because you will never compete with the Japanese-based companies." And that was in about 1984 or '85 that this pronouncement was made. It also is when companies like Intel were starting to lose money because we could not adequately manufacture devices. So, our company was, as they said, in serious decision at that point in time. What do we do with our future? Do we follow this academic advice and get out of the manufacturing business and just become a designer of circuits? Or do we in fact try to become a manufacturer? We actually chose the latter approach. And I can remember leading an airplane load after airplane load of our executives and our people to Japan, touring Japanese factories, talking to them about manufacturing technology, seeing what they did on their manufacturing floor. And after about two years of that, we went into a huddle and concluded that they weren't doing anything very sophisticated or anything very secret. They were just applying good engineering principles and engineering discipline, statistical principles and statistically control discipline. And they had set a series of high expectations on what manufacturing lines could produce. So, we came back to the US and committed ourselves to that, trained the entire company on statistical process, control, design of complex experiments, all those sort of things that are good engineering discipline but were lacking on the manufacturing floor.

And by 1988, '89, 1990, we had improved the situation quite dramatically. And probably one of the most proudest moments in my professional career was when, after having visited Japan about 20 times to learn how to manufacture in the mid 1980s, we had constant flocks of Japanese executives to Intel headquarters to see how we were manufacturing in 1990. We had completely turned the table. So, that's when really the company became, I think, noted as a manufacturing powerhouse of sorts. But it was really the application of engineering discipline and engineering principles to the manufacturing floor and it was not the sort of thing that the scientists and engineers automatically bring with them out of school to a manufacturing floor. But we'll get a little bit to that later on. One of the topics that we became known for was something called "Copy Exactly". The fact that we had half a dozen big manufacturing facilities and each one of them had a plant manager and each one of them was really king of their domain or queen of their domain, looking at the audience, they had absolute rule over what went on in their manufacturing plant. It was a little bit like a department head here in the school of engineering; I presume that you have absolute rule over what goes on in your department. Of course.

Of course. At least you think you do. But the problem we had was we have these six manufacturing plants normally running the same technology with different sets of equipment, different processes, different recipes. And if you ever try to move a product from one plant to another, it was next to impossible. None of the plant managers were willing to give up the autonomy or their control. I actually remember a meeting that we had one day. There were 21 senior manufacturing executives of Intel. I was running manufacturing at the time when I called the meeting. And I projected that we would change the way we would do business in the future. Rather than having independent factories, we would have very closely linked factories.

Every factory would look the same. We would run the same recipe, have the same equipment. And I was basically telling the managers who were there that I was taking a lot of their authority away from them. One of them raised his hand and basically said, "You know, this is a different style of management you are suggesting, Craig. Do we get to vote on it?" And having counted the number of participants in the meeting, 21 as they walked in the room, I said, "Yes, we're going to vote but I have 22 votes." And that's how this concept which has given some notoriety called "Copy Exactly", which is every one of our factories today looks the same, has the same paint, the same tiles, the same air conditioning, the same equipments, the recipe and they all operate absolutely identical. It drives a lot of our people nuts because they think that you should be allowed to twist knobs to optimize at the local level as opposed to running a common recipe across the board. But we do it the McDonald's way. If you go to any McDonald's around the world, the french fries all taste the same. If you come to any Intel facility, our products behave the same and manufacture the same. We can easily transmit them from one area to the other.

Along these 20 or 30 years that I've tried to capsulize for you, there were really three things that I thought were very, very important. One was Moore's Law and Moore's Law has been the roadmap for our industry. It has run since 1960 so we're approaching 50 years. It's not a law; it's an empirical observation, but it is the roadmap. And it is so engrained in the industry that, in fact, people are fearful of falling off of Moore's Law as extrapolation on their watch. The senior managers all say, "It's not going to die under my watch." And the new young engineers we get into the company don't know any better and they think it's going to continue forever. For probably the last 30 years, we've been saying Moore's Law is going to last another 10 or 15 years. We're still saying that today. It would be wonderful to come back in 10 or 15 years and talk to you again to see where we are in that respect. But Moore's Law has been the direction dictating the industry forever.

Now, there are a lot of naysayers to Moore's Law and they usually fall under one of two categories. They either fall into the category of financial analysts and the press who haven't a clue about the industry and a clue about the technology. And they all say, "Well, this can't continue because nothing continues to double. Besides, it costs too much to do that." And I told you the thousandfold increase in our manufacturing plants. What these guys don't ever recognize is you get a thousandfold increase out for that extra thousandfold increase in cost. And the other category of naysayers are competitors who don't like to spend money, who would like to just put a manufacturing plant in and then run it like a petrochemical plant or a power plant, that is, amortize its output over 20 years rather than having a two-year cycle of technology. But the leadership companies in the world have adopted Moore's Law as their technology strategy, their technology roadmap. And we do all sorts of things to perpetuate it, including a lot of research in universities like Stanford; they'll drive that fully. The other thing that I think has been constant in the industry has been the engineering effort that's involved to follow Moore's Law. And it's all sorts of engineers.

It's not just material scientists, a bunch of chemists, physicists, mechanical engineers, computer scientists, EEs. To go from a thousand-transistor device to a two billion transistor device is not a trivial effort. We used to design devices with two people. The first microprocessor was designed by basically three people, Hoff, Faggin and Stan Mazor. But today, one of our microprocessors may take a team of 500, 600, 700 engineers to do the physical design and layout. And then, you have additional hundreds of people who are doing simulations and doing the verification deal. So, everything has rather scaled up. But it's all predicated on strong engineering talent, which is why we hire essentially only masters and PhDs into the company who work as engineers. And bachelor's degree really is not sufficiently capable. The third thing that has driven the industry very simply the usage models.

And you are familiar with the usage models. The usage models are PCs and then various varieties of cell phones or smart phones like this BlackBerry device. And the fact that you are selling hundreds of millions of PCs and a billion or so cell phones or handheld devices a year consumes an immense amount of silicon. And I would like to just stop for a minute and personally thank all of you in the audience because you represent the generation that says, "You don't have enough processing power; I want more. You don't have communications capability; I want more. You don't have enough imaging capability; I want more." You are really the life blood driving our industry. So, thank you for that. But it is really that usage characteristics and the invention of the Internet or deployment helped along the way because that's allowed electronic communication to, you can sit behind a terminal and reach a billion people easily. When I was in your position, sitting in this room 40 or 50 years ago, we had mimeograph machines and we had handwritten letters back and forth then we had telephone calls. And that's about all we had.

But everything has changed and that usage model has really driven capability, along with Moore's Law and all the engineering talent that went into that. So, enough of history. Let me position myself to just give you five simple bits of advice and then try to take your questions. These are bits of advice that I've noticed in the industry. And 35 years or 40 years of watching the industry and going through 11 recessions, not as all severe as this last recession or as severe as the dot-com bubble of 2000. But we've had an 11 or so full-blown recessions in the electronics industry in the last 35 years. The first is very simply there is no replacement for sophisticated problem solving methodology in life. And you can describe problem solving methodology in a whole bunch of ways but engineers are good at problem solving because that's what you do. And that perhaps also is the reason why, if you look at the most common educational backgrounds of Fortune 500 CEOs, it's not law, it's not business, it's engineering. So, I think the training that you get in a university like this in problem solving is absolutely fundamental to your success in life, in whatever you choose to do, entrepreneurial life or work for a big company, whatever it is.

Problem solving methodology, whether it's the plan-do-check-act cycle like Walter Shoehart proposed in the 1920s and is still very valid today or just getting people to agree on the definition of the problem they're trying to solve, is one of the most important things you could do in all business world. If you don't believe that's the case, just look at the healthcare debate in Washington, D.C., today and try to get any two of our elected representatives to define the problem they're trying to solve, and you'll find that none of them will be able to define the problem. And if they do define the problem, no two will define the problem the same way. So, problem solving methodology and getting a common definition of where you're trying to go is absolutely critical. By the way, you learn little tricks along the way like if you're faced with a problem in life or in business or in technology, just start asking why. It's impossible to not get to the root cause of something by asking why four or five times. But you have to make sure that you don't get a bullshit response along the way. And people always give you answers that you are related to. Not the fundamentals, not the root cause, but they'll talk about something like why did this happen. And then, if you give me an explanation why that, OK, what caused that to happen in this way? If you ask "why" about five times, you'll do very well.

Another fundamental business tenet, I think, is always changing the rules. Quite often, people feel that they are captive to the environment that they find themselves in, that they have to continue to do things in the same way. But quite often, you can just artificially change the rules. And if you change the rules, that quite often will give you a head jump on the road. I'll give you one very simple recent example of changing the rules. How many of you have an iPod? What did Apple do to make the iPod so successful when there were generic MP3 players around everywhere? They changed the rules by all of a sudden making downloading music legal with iTunes. Now, you could buy the device. You could go to their website. You could download music legally or illegally. But they changed the rules on how MP3 players were sold and they were an incredible success.

And they also had great industrial engineering. One thing that Steve Jobs does fantastic is the user interface and the industrial engineering that goes into that user interface. But they basically changed the rules of the device. Have you heard of a company called Nokia? What did Nokia do to change the rules? This happened many years ago when a company called Motorola was the premier communications company in the world. It was still selling analog telephones, cell phones. And Nokia decided, "No, there's an opportunity here." Nokia had actually done some work in computers digitally literally. And they said, "Probably digital communications is going to be more important in the future than analog communications." Motorola sat on the old analog systems for a long time. Nokia moved over to digital side, changed the rules of the game. And if you looked at their relative positions in the market today, it's really related to that decision they made at that time. And you could go and on.

Kodak, how many of you have heard of Kodak? How many have had a film camera that you use? What's the matter with you two guys? But Kodak, bless their heart, they did recognize that the rules were changing around them with digital photography. But they were really wed to their business model, which was film and paper business model. And they let the environment changed around them. And Kodak today and Fuji as well are around the shadows of themselves because they didn't change the rules; they let the rules change them. Companies like Intel, we've done lesser things along these lights. How many of you have ever seen an "Intel Inside" logo? It's probably the most successful ingredient branding program ever in the history of the universe. It changed the rules. At Intel, we don't interface with anybody who buys our stuff directly, we do it indirectly. So, to deal with them indirectly we had to have an indirect advertising/marketing branding campaign. And as "Intel Inside" worked into computer manufacturing, it changed the rules and gave us a leg up the competition.

We also at one time and computer companies in the US and around the world found themselves to be continually outsourcing their engineering and doing less and less R&Dto move the computer technology ahead. We found that as manufacturers of microprocessors, it was important for us to do their research for them and then give them that research for free. How else could we sell our products if they weren't going to do the research and the engineering in these next-generation computers? So, we created something called "Intel Architecture Labs", which was basically an Intel-financed computer industry research organization doing research for the computer industry as a whole. Lots of examples that you can go through in that area of changing the rules. Anytime you can change the rules ahead of competition or change the rules before they get changed on you will do very good things for you in the marketplace. The third observation I have is no matter how big your company is and how good you think you are, you do not own all the smart engineers in the world. There are lots and lots of smart ideas and smart engineers that don't work for you. So, how do you capture their talents? How do you use them? We've talked about this a lot internally in the early 1990s and said, well, one of the ways you can do that and one of the ways you can use all that smart engineering talent to your advantage is, in fact, to become the world's largest high-tech venture capital company and to fund research and startups in your general computer Internet area that compliment your own work. And Intel is today the largest high-tech venture capital company in the world. We have billions of dollars in our portfolio and hundreds of companies in our portfolio.

And the process of thought there was simply there are a lot of bright engineers out there. We would like them working around our general space, creating technology into the marketplace that complements our own to grow the market so we can grow our sales of microprocessors. Again, it's changing the rules of the game but also recognizing where the talent resides. We also recognize not just Intel but the industry as a whole that universities like Stanford and Berkeley and Michigan and many others are top-flight universities with great researchers. And that's why we put together basically $100 million a year where we fund university research activities to complement our industry, to make sure that Moore's Law continues to move forward. But it's that basic concept that you're not an island, you don't own all the resources, you need to figure out how to make use of those resources to your benefit even if they don't work for you. And if you make intelligent venture capital on this, then you not only get the output of those people but you get the benefit of, in fact, their success that flows back to you as the equity investment you've made. Along those lines I'm always drawn by the power of universities like this type to be disruptive influences on an industry. And the way I usually like to describe this is Intel Corporation has an R&Dbudget of about $6 billion a year, which I think is even bigger than the Stanford School of Engineering's research budget. Microsoft has a budget probably of about $7 or $8 billion.

If there's a Microsoft person in the audience, you can correct me but in that range, slightly more than Intel. And every time I look at those two companies, and I'll pick Microsoft as an example here, and look at the challenges that those companies have with their huge R&Dbudgets, absolutely huge budgets, what are the life-threatening challenges that occur and where do they come from? And the life-threatening challenges to Microsoft have not come from IBM or Hewlett-Packard or SAP, Oracle or any other big company with a big research budget. But the three near-death experiences for Microsoft came from one or two researchers at a university with a bright idea. Netscape, Internet browser, University of Illinois or the Illinois Supercomputing Consortium, whatever it was called, and Yahoo!, Stanford University; Google, Stanford University. Each one of those instances, one or two graduate students, bright engineers with a bright idea, able to challenge a company with a multibillion-dollar research budget. And that's why I think research universities like Stanford are so important. And that's why it's so important for companies to maintain a good relationship with you guys. We don't want you eating our lunch routine basis. We'd like to have access to your ideas before you take them out in the marketplace. But the individual idea is, in fact, really the key strength around the world.

And there is no underestimating the value of this single smart idea. It can take on the largest corporation and bring it to its knees almost overnight. And it doesn't take a big research budget to do that. I want to say one other thing about, if any of you aspire to become an executive in a company someday, start growing the thickness of your skin right now because you will be a constant target for commentary from two directions. One is the press and the other the financial analysts. I don't want to sound defensive but neither one of those categories of people knows anything about our industry but it never hesitates from writing about our industry on a continuing basis. I'll give you two examples. By the way, there is a standard rule in the industry if you're a CEO that you're never as good as the press says you are and you're never as bad as the press says you are. When they're saying good things about you, it's really cool. When they say bad things about you, it does get under your skin a little bit so it helps to have thick skin.

But two really separate examples, my own personal experience. One was in 2000, there was something called "dot-com bubble burst", when evaluations of companies like Intel went from $300 or $400 billion to $100 billion. Microsoft did the same thing. Cisco did the same thing. Every company who lost about 65% to 75% of their capitalization. And dropping hundreds of billions of dollars in the marketplace almost overnight is not a good thing for your shareholders and not a good thing if you're a CEO at that time. But what we did at that point in time is we thought we understood our industry better than the financial analysts who were saying, "You guys are dummies. Don't you understand there's a recession? Don't you understand demand for your products has dropped? You should have big layoffs. You should stop capital spending. You should stop investing in R&D

You should just recoup. Go in a corner and sit there until things get better." We said, "No, no, no. We follow Moore's Law." And what does Moore's Law say? It's new technology every two years and doubling the capacity. Besides that, Gordon Moore, our founder, uttered a famous saying one day which says you can never save your way out of a recession. You can only invest your way out of a recession because the only way to get out of a recession is with new products and new technology. So, how do you get new products and technology? It means you beef up your R&Dbudget. You beef up your capital budget to be able to produce that new technology. So, for 2001, 2002 and 2003 the press and the analysts had an easy target to shoot at. It was Intel and its CEO for this ridiculous fact that we increased our capital spending and our R&Dspending in the worst recession our industry had ever seen. In 2004, everything turned around.

We had one microsecond of glory when they said, "You know, we've been criticizing Intel for the last four years or three years on their stupid decisions. It turns out they were right. So, OK, Intel." Now they turned around and started criticizing us on something else the next day. You have to get used to that and have a thick skin associated with that. The other area that's associated with the dot-com bubble collapse was, well, industry evaluation. And a lot of people in the industry went on acquisition sprees because there was going to be no end. The Internet traffic was going to double every 30 days, which was the common idea at that point in time. So, the telecommunications sector was red-hot. The electronics sector was red-hot. Intel did a series of acquisitions.

We bought about $10 billion worth of companies. Our market value at that time was about $400 billion. That works out, I think, about 2.5% of the market cap when the company went into the acquisitions. It was a de minimize trivial amount if you manage the company and say, "What's the big deal?" But the press and everybody said, "I can't believe this. They've spent $10 billion on acquisitions. And what do they get for it?" They're never pointing out that, in fact, it was 2% of the value of the company and we were requiring technology to make the company successful in the future. Grow a thick skin. It's part of the management deal. Let me conclude very simply with the following, which relates to what you're doing here and relates to international competitors. There are only three things any company or any country can do to be competitive.

And you're part of two of them. What makes companies and economies and countries competitive is smart people, people with good education that can add value to what they do; that's why you're here. Smart ideas, and that's the R&Dbudget. That is creation of ideas for the next generation of products, services, companies, startups. And then, the right environment to put smart people together with smart ideas. And you actually see some of that here in Silicon Valley, although perhaps less and less as we go forward. What happened in Silicon Valley was that you had smart people and smart ideas coming out of Berkeley and Stanford and elsewhere. You had venture capital money here, which was funding startup companies. Venture capitals basically started here at the valley. In fact, Intel was the first, I think, officially founded company with venture capital from a guy who still does work in San Francisco, Arthur Locke.

But you had smart people, smart ideas and the right environment for people to invest in innovation. So, if you're managing a company, smart people and smart ideas and the right environment, you'd be successful. If you're President Obama and managing the country, it's smart people, smart ideas and the right environment that get people to invest in innovation. There's no other formula that works. I think by and large a lot of the high-tech companies have done it right over the years. I'm not so sure about Washington, D.C. With that, I'll open it up to any questions you might have. So, Craig, thank you. As I mentioned and some of our audience knows, part of this audience is our MS&E278 class, the spirit of entrepreneurship which surrounds this class, and they get the first couple of questions. So, I will read them.

You will translate them? I will translate them and then we'll open it up to the rest of the audience. You alluded to, in fact a couple of times in this speech, the importance of educated workforce to Intel. And I know Intel under your tenure had even more sharp focus on education. Now that you're out of Intel, what would the three things you would change about education in America be if you had a magic wand? The three things really not to university education. Now, I think university education works perfectly well for all the reasons that public education does not work well. You have well-qualified teachers or faculty and universities. You set high expectations and you have feedback groups. A feedback group is if you research and you can't research funds in the university. That's a feedback group, which you would understand. If you look at K212 public education in the United States, it suffers from a lack of quality that's teachers.

In fact, if you look at math and science teachers across the United States, the probability of getting a math or science qualified teacher is 60% or 70% at best. And I'll let the audience do the mathematics. If you need to have 12 good math teachers in the world and the probability each year of getting a good math teacher is 70%, in your perfect filter you can do the math. It's 0.7 to the 12th power. It's less than 1% chance of getting 12 good ones in the world on average. So, you need better teachers especially for math and science. You need higher expectations. We have dumbed down the education systems in the United States. If you look at nearly any state that has an exit exam for high school graduation, they're a joke. They're designed on 8th to 9th grade.

And you guys have taken them. You know how bad they are. So, we have dumbed down the expectation level. And then, as far as a feedback group or how the system responds, President Bush tried to institute one of the simplest feedback systems called "No Child Left Behind", which was just a study of our kids succeeding, our teachers succeeding, our schools succeeding, our administration. If you're an engineer and you run a company, you don't do things without measuring things. You don't make decisions without data. NCLB was an attempt to make data available so you can make decisions. So, if you look at K212, basically poor teachers, low expectations and no feedback group. Other than that, it has been great trip. I'm glad I asked.

The next question actually was focused on your career path. Most the way through it, you were running manufacturing or in the manufacturing side. But when you became president, CEO and chairman of the board, you had to make allocations between investments in the process side and investments in the product side and architecture. How did Intel think about that balance of constantly having to renew capital investments and process versus new investments? Well, it's a challenge if you choose to rigorously follow Moore's Law which is two-year cycle on process technology, which means you have to redo your plans every two years, big-ticket item. The thing that we've done is, first of all, we do the financial analysis. Does it make sense to follow Moore's Law? And unequivocally, it always makes sense to follow Moore's Law. That doubling you get in capability more than pays back for the costs of the investment. The other thing we did is to make sure that everybody in the company understands that the success of the company is Moore's Law. You follow Moore's Law, you have leading technology, leading products, you're successful. If you don't, you're not.

So, when we get in tough times, there are two things that we don't touch or cut, the R&Dbudget and the manufacturing capital budget. Even though we're laying off finance guys and marketing guys and gals and human resources people, we're never cutting those two items because everybody in the company knows the only way the company is successful, the only way my stock option is worth something is, in fact, if we're the technology leader. You can't be the technology leader unless you invest. So, there is never any animosity, never any second guessing. You got to keep the R&Dbudget whole and then the investment in the new technology. Got it. And last question from the class and then we'll open it up to the audience. Given that every generation of computing platforms seemed to switch processor technology, mainframe to mini computer to desktop, Intel own the desktop and the mobile platform now emerging, how much did Intel worry and how much investment? Was it that Intel was going to break the mold of some new entrants? We, in fact, concentrated heavily on the computing side. We flipped that trend that you mentioned. It used to be there was a different technology for mainframe, separate technology, minis and servers and different technology from desktops to laptops.

We turned that around as biologic common architecture now with some exceptions, everything from the smallest laptop computer up to the biggest petaflop machine, supercomputer. Unfortunately, it's mostly built on our architecture which is nice. And other people who have big screens negate that. We quite frankly do not pay much attention to the cell phone industry really for two reasons. One is we were concentrating on processing power because that was what was driving desktop and big compute. And processing power carries with it the burden of electrical consumption. And that's contrary to what you have in cell phones. So, we didn't focus heavily on the low-power devices. And much of that market got seated to ARM and other architectures were now more serious about going to low-power Intel architecture devices and going after that market. But we didn't pay much attention to that for an extended period of time.

Anyway, the other reason we didn't pay much attention to it is that during that timeframe in the 1990s and the year 2000, the average selling price for an Intel microprocessor was $150 to $200. The average selling price for an ARM processor to go on a cell phone was $5. It was not a difficult decision to make. Let me open it up to the class. In the back? How did you convince the Japanese electronics manufacturers to let you come in and observe their process? It was a combination of things, really. I mean, first of all, the industry is pretty open to that and those sorts of business. And I think the Japanese were incredibly proud of their capability and didn't think anybody else could copy it and so were proud to share it. The other thing you have to remember is there was maybe, you don't have to remember because you're not that old, but there is pretty much of a trade war going on between US and Japan at that time. And I think the Japanese folks were trying not to exacerbate the nature of that trade war by being open and inviting people into their country. But they were very open and sharing, as much as we were when they came back to visit us later on.

Next question. Biggest death threat and what comes after Moore's Law? The biggest death threat is, as Andy Grove described it, only the paranoid survives. Now, the death threat is that you lose your paranoia. You stop the relentless advancement of Moore's Law in technology. Every time we have had problems in the market, it's usually not so much competitive threat as it is we did not need our own plans. We had a development program that didn't need its criteria for some reason. So, usually we'd say our biggest challenge is our own ability to do what we plan to do. What comes after Moore's Law? What comes after Moore's Law? That's for you guys to figure it out. I mean, I think Moore's Law is good for another 15 years. I retired in May.

You're just starting your professional challenge. It's your challenge. What comes after Moore's Law is pretty critical. It's simple. A switch is going to replace the transistor. And the question is what switch, and nobody has decided that yet. Actually, it's following up on that point. That's true, what you thought the role of the Internet, distributed computing, embedded computing seem to be playing towards moving beyond physical limitations. Well, the Internet which was a huge driving force for computing for every reason possible, also gives you the possibility of more distributed computing capability and cloud computing and all that. But I think fundamentally what you have is the human desire for personal computing element.

That was why a PC is called a "Personal Computer". It's not a dumb terminal that the PC replaced. I don't see us going back to dumb terminals which always have to be connected to do something. But you will see consolidation of server farms and things like that for higher utilization of that resource, for sure. I have my own Intel processor and everyone uses them. But then, you also like competition. And I know that you're least concerned about competition in this space. I'm from Europe, Denmark. And when I think of Americans, I think of them as very competitive. But in the desktop space, there's virtually no competitor because you've been so good.

Do you think there should be competition and how would you want to stimulate competition because some say where there's competition, there's innovation? I'm not quite sure how to raise this question. Well, we'll forgive you for your opinion. For the American antitrust issues are designed to protect the consumer. The European antitrust laws are designed to protect the competitor, not the consumer, which is why we have problems in Europe because Neelie Kroes and the Competition Commission are designing their decisions and their actions to protect competition. That's point one. Point two is if you look at this industry from many objective measure, it is the most anti-inflationary, most competitive industry known to mankind. If you don't believe that, go to the Department of Labor which tracks 1400 commodity devices. Including one of those 1400 is the microprocessor. What about in the PC space because it's just you and AMD? Well, we compete fiercely and have done for 25 or 30 years. Every time I hear that question in one form or another, I think that, "OK, we've been following Moore's Law for 50 years.

How much more competitive can you get?" Moore's Law is actually speeding it up rather than slowing it down. And that is because of the inherent competition in the industry. Ergo, how can you say there's no competition? I tried to say that to Ms. Kroes to no avail. Actually, one more question. You mentioned that engineering discipline was critical to Intel's success in manufacturing. Could you give an example of what you mean by engineering discipline? Very simply, if you look at a typical processed technology where you take a bare cell wafer and turn it into a high-performance microprocessor, probably 200 manufacturing steps or 200 variable steps involved in that. And the control of each one of those steps may be that you're laying down a thin-gate oxide and it has to be four mono layers plus or minus zero. This is not kidding. That's what we are doing across a 12-inch diameter wafer.

Now, how do you achieve that and how do you measure that you're achieving that? How do you measure that you're in control? If you want to change one of those 200 variables, how do you do it intelligently when natural drift and some of the other variables may be bigger than the variable that you're changing? So, how do you design your experiments with that many variables to make sure that if you've changed something, you really have the right answer associated with what you changed. You can then just take it to the more mundane stuff of scheduling. It's a batch processing industry. How do you schedule it effectively to get maximum utilization of your multibillion-dollar resource? Every step of the way has an engineering discipline associated with it. So, Craig, thank you very much.