Tony [00:00:04] Welcome to Code Together, a podcast for developers by developers, where we discuss technology and trends in industry.

Tony [00:00:11] I'm your host Tony Mongkolsmai.

Tony [00:00:18] Today we're going to talk about one of the interesting but not necessarily well understood areas of computing, quantum computing. Something you may have heard is that simple quantum computers can beat supercomputers and will replace them at some point. In fact, Professor Satoshi Matsuoka-san the head of the Riken Center for Computational Science, the largest supercomputing center in Japan, recently coauthored a fun paper, "Myths and Legends in High Performance Computing" and this was number one on their list. Of course, there are places where quantum computing excels, but how it works and where it can be useful can still be confusing. To paraphrase Richard Feynman, a Nobel Prize winning pioneer in the quantum computing space, if it was easy to explain, it wouldn't be worth the Nobel Prize.

Tony [00:00:58] All that being said, there are a number of companies working towards commercializing quantum computers. Small companies like D-Wave Systems and I am Q and larger companies like IBM, Microsoft, Google, NVIDIA and of course Intel are working on building quantum computing solutions. Today we are joined by Anne Matsuura, one of Intel's experts in quantum computing, to help us understand quantum computing and how Intel is tackling this challenging space. Anne is the director of Quantum Applications and Architecture at Intel Labs. She previously held positions as the chief scientist of the Optical Society (OSA) and chief executive of the European Theoretical Spectroscopy Facility (ETSF). She has a Ph.D. in physics from Stanford. Welcome to the podcast.

Anne [00:01:41] Thanks, Tony. Glad to be here.

Tony [00:01:44] So can you give us an overview of how quantum computing works and how it's different from traditional computing?

Anne [00:01:51] Sure, Tony. You know, quantum computing is an entirely new kind of compute. So it has a entirely new execution model from classical computing.Classical computers, we're familiar with using binary arithmetic zeros and ones in in the execution. Quantum computers execute programs in entirely different way. It's all about probability. In a quantum computer, you're manipulating the probability amplitudes of these quantum wave functions, and then you're sampling resulting probability distributions. So programing a quantum computer is entirely different than programing a classical computer, and it requires an entirely different kind of machine with new kinds of hardware, new kinds of software, new system architecture, new applications.

Tony [00:02:46] One of the interesting challenges there is that we have a large number of developers in the world that are comfortable with kind of their programing languages C and C++. And binary computing we think of, for me at least, I think of like binary operations. But this requires a totally different mindset. I wonder how possible it's going to be for a traditional developer to then kind of move into this quantum computing space. Is it something that people should be able to do or is it something that you think will be challenging for a traditional programmer such as myself?

Anne [00:03:20] I think it will be challenging, but we are actually, and others, are really trying to make it easier. We are really trying to reach out and get classical programmers interested in learning how to program quantum computing. In fact, you know, we've recently launched a full stack in simulation called the Intel Quantum Software Development Kit, and it's in C++, right? So we're really trying to encourage classical programmers to try out quantum computing, and we have an ulterior motive. We're trying to get the brainpower from classical computers who are used to developing workloads, algorithms, applications to contribute to the quantum computing field.

Tony [00:04:10] And you mentioned that with quantum computing, we're dealing with probability distributions and kind of overlapping wave functions. And obviously when I think of computing, I think I think of binary because I'm a traditional, I guess, computer scientist. How do we actually get our answers out? So I can imagine from my basic physics knowledge, overlapping probability waves, but that's giving me probabilities. It's not giving me necessarily this is answer A or this is answer B, It's somewhere in between that suggestive of an answer. How should I be thinking about that? Is there a way I can think about that? That kind of maps back to traditional computer science?

Anne [00:04:51] That's a really perceptive question. You're absolutely right. What you get out of the computer is the quantum computer is a probability distribution. You have a certain probability that this is the right answer. Right? And that is some that is an entirely new concept. I think it will change the way we have to think about computing.

Tony [00:05:16] Will we have to have, you're a physicist, I guess by a degree. Maybe you're a computer, maybe now you're a computer scientist by trade, maybe not. But in the future, do you think people are going to have to really understand quantum physics in order to program these quantum computers?

Anne [00:05:40] I certainly hope not. Right? I mean, today most quantum computer users are people who have PhDs in physics. And if we really want to gain traction with this new technology and if we really want to build an efficient, scalable computational device for quantum. It is very important that we reach the classical computing, classical computer engineers, programmers in order to help us to attain, you know, to get to the goal of building a practical, really large commercial size quantum computer.

Tony [00:06:27] And one of the interesting things as I've gone through and, you know, kind of dabbled in quantum computing, is that I noticed that quantum computing is still being programmed of kind of at in the traditional computing analogy at the gate level. Right? So we build a circuit, you know, so like when I was taking my computer science classes at the University of Illinois, we had two classes in computer architecture where I was learning how to build circuits, our use adders, XOR gates, things like that. And that's kind of the model that quantum computing still seems to use.

Anne [00:07:06] That's right. And that is going to have to change. Right? There are there are groups that are working on programing languages, trying to abstract away the hardware. More and more, the hardware in this area is still rather nascent. Right? And that also is...so at Intel, what we're doing is you're trying to kind of work from the bottom up. We have our quantum dot qubit hardware and we're learning what gates are native to it. And we are, you know, we've built a compiler. We're also starting to think about how do we abstract the native, you know, the gates away from users so they don't have to be so hardware aware, right, in order to use a quantum machine. But yes, you're right. At the moment people assume that you will know what the quantum gates are that you want to actually put into your algorithm circuit.

Tony [00:08:07] How long do you think that type of transition will take? How far along are we in that process? Are we at the very, very beginning? One of the things that I said in my intro, or at least that I thought about saying in my intro, which I may not have said in my intro, was that quantum computing has been around kind of since the early eighties, but now because we're getting them in practice, you know, we've got I guess this is 70 years, you know, since the 1950s of experience with binary type computers. Maybe more than that, I should probably know the answer to that, but in my mind, it's 70 years. So here we have maybe 40 years, but really seventy years of practical experience versus a couple of years of practical experience around quantum computing. How long do you think it will take for us to kind of understand these paradigms, Obviously not holding you to any timelines that you may have in your mind?

Anne [00:09:06] Yeah, I mean, it's still early stages for us so far for the field, so. All layers of the computational stack. Now it's just it's a research field. It's still a research field, even though we have these small quantum computer, you know, these small machines where one can dabble with small numbers of qubits and run things on small numbers of qubits. But, we always say we're at least ten years away from larger scale commercial sized quantum computer. It's more practical that we can actually do things that are, that you aren't able to do on classical computers. Right? It may actually take longer than that for us to truly make inroads into the to make it so that a practical programmer will be really comfortable with quantum computing. I hope it you know, we're really trying to make that jump right now. I mean, we and there are other other entities in quantum computing. We really are trying to reach out to the classical programing community. It's difficult because most of us were trained as non computer engineers or like you pointed out, I'm originally physicist, right? And so it's kind of a cross-disciplinary thing at the moment. My team is like half computer engineers and half physicists, and we have we have a very interesting time communicating with each other. It's turning out great. I mean, it's it's it turns out to be a really it's a really exciting area to work in because you're learning constantly from really smart people, from other disciplines while you're doing your day job. Right. And I think that's the key We have to in order for great innovations to happen, we have to have, you know, the point of view from really smart people, from the different fields, computer engineers, physicists, you know, and other kinds of scientists and engineers as well.

Tony [00:11:11] And because we are kind of talking about it. But I probably didn't, we didn't cover kind of the basics of it. You mentioned the qubit, which is kind of the basic, I don't know, maybe what's the right terminology, the basic unit of quantum computing? The basic strcture?

Anne [00:11:24] Yeah, it's the a bit. It's a bit. It's a quantum bit. So yeah, you can make a quantum bit out of any two level quantum system, basically.

Tony [00:11:34] So for people who who understand traditional bits of 0 and 1, can you quickly just give us a high level? What is a quantum bit? What does a quantum bit represent? Because it's not a zero or one.

Anne [00:11:48] No, it's in many senses, it's kind of everything at once. Right? So if you have a if you think about a quantum bit as, we like to say use this kind of cartoon example where if your quantum bit is a spinning coin, it's sort of in both heads and tails, sort of in both zero and one at the same time. And then you bring two qubits together and you entangle them and then you have 2^N states, you know, four states at the same time. Or if you have N qubits, you have 2^N states at the same time. Right? And that's sort of the power of quantum computing, is you kind of have this exponential property to it. And you can calculate on all those states at the same time. That's the point. Right?

Tony [00:12:44] Right. So that's the interesting thing. It's funny, as you started talking about that, you said it's kind of everything. I was trying to figure out, how do I leverage the movie Everything Everywhere, All at Once, as part of our promotional stuff. But yeah, I think the interesting thing for quantum that, at least for people who are thinking about it kind of in a traditional sense, is that quantum computing allows us to deal with areas where there's high combinatorics, if that's a good way to put it right, when there's a lot of combinations that can be achieved, which can be very expensive in traditional computing to capture all of those states. Quantum computing allows us to kind of deal with that complexity in a way that is simpler in terms of how we program the solutions, right? Because the things exist in the multiple states. So it kind of combines all the probabilities of the combinatorics together. Is that a fair way to put it?

Anne [00:13:40] Yeah, you can compute on all the different combinations at one time, right? I think that's that's the power of it.

Tony [00:13:47] And so we should probably... It'd be interesting to talk about kind of the areas that people are looking to apply quantum computing. And of course, as any good host will do, I just Googled to try to figure out where there would be some type of opportunities to use this. And the most popular one that I think everybody's heard of is kind of cryptography, right? The ability to crack what is currently secure encryption algorithms with quantum computing. And there's a lot of effort around building quantum resistant cryptography. Are there other places that we are thinking that we can apply quantum computing that makes sense?

Anne [00:14:26] Sure. And I just wanted to note that for cryptography, we're going to need a very, very large quantum computer, right? So that's not generally where, you know, those of us who are thinking about nearer term applications, we don't generally start there. I generally gravitate towards simulation of nature, right? So having your quantum machine simulate other quantum systems, so simulation of chemical reactions, understanding how to create better catalysts or understanding how to create better materials with electronic and magnetic properties that you want. And now clearly to do anything really exciting in these areas, we're going to need a very large scale quantum quantum computer. I'm thinking, think thinking like a million qubits, right? This is, that's the size that I'm thinking, or that Intel thinks, for that kind of commercial scale practical computer.

Tony [00:15:31] One of the things the applications that I saw when I was researching applications was people using it to try to identify better battery technology, right? The optimal properties of a battery in order to make them more efficient, which obviously becomes more and more important as we move towards the electrification of cars and trying to move away from fossil fuels and things like that.

Anne [00:15:52] And these are all problems, this kind of simulation of material simulations of chemicals, chemical reactions. These are all things that we generally use HPC centers to try to, it takes a lot of computational resources to try to solve these problems and simulate these materials. And I really do think that's a way that quantum computing, large scale quantum computing is going to help us in the future.

Tony [00:16:21] But you did mention that you think that we might need a million qubits to solve a problem. I think the last thing I saw was IBM had published something with 50 qubits.

Anne [00:16:31] That's probably right. That's probably right.

Tony [00:16:34] In terms of a million qubits, how far away do you think we are from that type of technogy, is that kind of the ten years you were talking about?

Anne [00:16:42] Yeah, that's kind of the ten years that I'm talking about. And that will also, you know, we'll also need to have error correction, right? That's something that is not implemented in these small systems at this point. We're just working with physical qubits instead of logical. We're using eight one physical qubit as a logical qubit, but it is not error corrected, right? so that's something that's going to have to change to make this truly a computational device.

Tony [00:17:08] And why do we have to error correct it? I did read something where they were talking about the fact that quantum computing is very susceptible to environmental factors because of the small scale and things like that. But what do you mean by error correction?

Anne [00:17:19] Yeah, so most qubits only last for fractions of a second. So when we run workloads on today's qubits, we generally have to run things really, really fast. And as you had mentioned, you brought up the statistical nature of quantum computing is absolutely true. So you run a workload thousands of times to get statistical relevance in your answer, right? So we need we need qubits...we need to be able to sustain our qubits for longer. So we will need more robust qubits and we'll also need to be able to error correct the qubits as we go along.

Tony [00:18:01] So when we have a million qubit computer, you're not saying that we're going to run an experiment so that it runs over a million qubits? You mean we actually will need a million cubit computer, which we will have to run multiple times in order to get a correct answer, or at least a reasonable answer?

Anne [00:18:17] Yeah, that could be true. Yes. It is possible... I'm hoping that by the time we get to that scale, a computer that will also be able to run things on various sections of the qubit chip, if you will, right, in parallel. But yeah, I was thinking, yes, the computation on a quantum computer is very fast.

Tony [00:18:42] With a 1 million qubit computer. That's an incredible amount of, I guess, combinatorics that can be built into that 1 million qubit computer. Is that more than what would be possible with, I guess, the most advanced binary computer that we could conceive of? So there are certain things in physics where we say like, okay, the number of electrons in the universe is X, and if we used a typical binary computer, you might be able to compute, you know, Y, to come up with another variable name, is the million qubit quantum computer of such power for those types of problems that it would never be possible for us to build a binary computer that could solve that problem?

Anne [00:19:29] I do think it's true that quantum computers will be able to eclipse classical computers, not that they will replace classical computers, but will be able to eclipse them in the number of possible states that you can compute on at the same time. But one of the reasons we advocate for a million qubits is that each probably about ten physical qubits will be in each logical qubit. So when we're talking about a million qubits, I'm talking about a million physical qubits, right? It will take we believe the field believes it will take a lot of physical qubits just to create one logical one.

Tony [00:20:13] Just out of curiosity, when we build a quantum computer, we say quantum, so it makes it sound like it's really small...

Anne [00:20:20] No it's huge...

Tony [00:20:23] Yeah. How big is a quantum computer? Because I actually need a lot of things to harness these forces, right?

Anne [00:20:28] It looks like the old the really, really old time classical computer, Right? It takes up a room. And our quantum, our qubits live in a large cryogenic refrigerator, you know, So and people are talking about daisy chaining cryogenic refrigerators together in order to have a bigger machine. So it takes up a room, at least.

Tony [00:20:56] And how many how many qubits do I get if I have a room?

Anne [00:20:59] It just depends on what what kinds of qubits, what you're making your qubit out of. So one of the things that Intel, one of the reasons that we've chosen quantum dot, silicon quantum dot technology to make our qubits is that they're basically single electron spins. They're very small. They're much smaller than other kinds of qubits. And they look like fabricating one electron transistors, frankly. So they're perfect for our Intel fabrication process, and we're experts in silicon and all of that. So we're hopeful that we can we can jam more qubits onto a chip. Now, that may be we may find out that there's physics we don't understand when you put these quantum dots too close together. So there are many things to learn. There's still research field, right? But a lot of qubits are large and therefore it will take a lot of refrigerators or or huge refrigerator to in order to get to the million qubit mark.

Tony [00:22:10] Yeah, and I'm thinking about it, just random thoughts about like sustainability. Obviously, one of the things that people are thinking about about nowadays is sustainability. And we talk about kind of bigger systems. How much power does it take to power a quantum computer, at least right now? And then kind of where are we going?

Anne [00:22:37] Oh, I should have numbers on that. It's not as big you think. I am actually the software side of the program, so I'm not thinking about those numbers on a daily basis. But it was surprising. I have asked that question before, and it's not as large as you think. We will remember that these these systems are our quantum machines will be situated at a data center or at HPC center. So for them, I really don't think that our power requirements are going to be too extravagant. You can buy cryogenic refrigerators off the shelf. Now they're they're typically their standard lab equipment. So although they sound exotic, it's it's not quite as bad as all that, but they are big.

Tony [00:23:22] That's interesting. And then we've talked about you mentioned a little bit about the the new Intel Quantum SDK. I know that you guys have a new release out. Can you talk a little bit about the software side of things? As you mentioned, you're on the software side of things. What's new with the Intel Quantum SDK? I know that it was mentioned last year at Innovation for Intel at our big conference. The Intel Quantum SDK was announced. What what are going to people are going to see when they go try this Intel Quantum SDK out.

Anne [00:23:52] So the Intel Quantum SDK version 1.0 has just come out and what you can do with it is a full stack in a full quantum computer in simulation. Basically, it's a C++ based compiler based on LLVM Industry standard. It's optimized for running algorithms that require classical and quantum parts to the program. So these kind of very what they call variational algorithms is what we are optimized for. From our beta users, we found that computer engineers particularly find it interesting because you can you can do things that you can in classical computing, like you can compare compiler files and see how your algorithm is being optimized during the execution. The back end is now a simulation of the Intel quantum hardware. So this is really part of the roll out towards the the Intel quantum computing machine. What we wanted to do is to put out the SDK early so that people will get used to using Intel quantum technology and will know. So using the SDK will have the same feel as using the Intel quantum computer. It's available on the Intel Developer cloud and I think we can we can give you the, the links to sign up for the Intel DevCloud as well as the Intel Quantum SDK.

Tony [00:25:20] Okay. So this is publicly available, right? People can actually just go try it out themselves.

Anne [00:25:25] Yes. And we encourage that and we would love feedback as well to remember this is something that is just coming out of beta version. We're just releasing version 1.0, you know, now. So I would love feedback from users.

Tony [00:25:43] And in the long run, the idea here is that if you program for the Intel Quantum SDK and go run it kind of on the back end simulator in the long run, actually when Intel quantum hardware is released, you'll use the same code, kind of the same toolchain at that point, whatever that looks like, obviously. Yes, you know, it could be many years away, but it'll be the same toolchain for for your customers and our customer.

Anne [00:26:06] That's correct. And when you write, when you when you program the Intel Quantum SDK, it lowers your operations into native operations for the quantum dot simulator. So it is it is a true compiler. So you can run it full stack, right? Like it's a computational device, right? We also have an alternate qubit back end that was on our beta version and that's still going to be available. And that's that's a qubit simulator. That's a generic qubit simulator, but it's high performance and you can run up to 30 qubits on a single node and more than 40 qubits on multiple nodes. So it's really a high performance qubit simulator and that's it that you can also use as an alternate qubit back end for the SDK.

Tony [00:26:50] The Intel backend... You said it's a simulator for the Intel hardware. How many qubits would I get if I'm running that versus kind of the generic simulator?

Anne [00:26:57] Because it's actually a physics simulator of quantum dots, it can simulate up to nine qubits at this point. And it's also on the on the Intel Developer Cloud. We have been, you know, so we hope that we'll be able to have access to more nodes. And that will also increase the number of qubits that are available. But that's a new qubit backend for us. So that simulator, of course, will continue to be upgraded as time goes by.

Tony [00:27:31] Interesting. So obviously as part of the SDK you guys have, and I've tried out the SDK, actually the beta version, not the 1.0 release yet myself. You guys have some some samples there and some examples that will help people kind of understand how the SDK works and how quantum computing works.

Anne [00:27:52] That's correct. And hopefully there are kind of representative samples to help you help get you started in programing quantum computers.

Tony [00:28:03] So one of the things that you mentioned that I found interesting was you said with the Intel Quantum SDK, you can kind of interleave quantum computing as well as traditional computing, which I think is pretty important. The way that I think of quantum computing perhaps incorrectly is similar to how people are using GPUs. So obviously I talk a lot about oneAPI with my colleagues. And so we think of things as I'm going to run it on the CPU, but potentially I will offload something to a GPU. Ideally in the long run for us, we're going to interleave the two. And it sounds like you're kind of thinking the same way, right? You're going you want to be able to kind of have traditional logic, but when when it makes sense, you want to offload things to the quantum computing part of the system.

Anne [00:28:50] Absolutely. I think quantum computer will actually be an accelerator at an HPC center. There are some things that you would never use a quantum computer for that classical computing will always be better for. So it's it's it's great if you can leverage both sides for leverage classical for what it does best, you leverage quantum for what it does best. And I think that's the best solution. So it's a hybrid approach.

Tony [00:29:17] I think the most interesting thing that you mentioned was about what a quantum computer actually looks like. Like nobody talks about the physical requirements of what a quantum computer is when we talk about quantum computers. In my head, I think of computer chips. I think of silicon.

Anne [00:29:35] Oh my goodness. You know, it looks like a large physics experiment. So if you think of a physics lab with all the big, you know, cryogenic refrigerator, the big equipment in it, it looks like a physics lab, if you're familiar with that.

Tony [00:29:53] Honestly, I don't think that I am. So it would be it'll be really interesting. And I think we'll try to put a picture of that, if I can find one online.

Anne [00:29:59] Yeah, I got one here. Look like I have a picture of our lab.

Tony [00:30:06] Oh, that'd be cool. Yeah, I'll see. Actually, I'll see if I can find a way to link it publicly, because I think it'd be really interesting for people to to see what a quantum computer looks like, because I think everybody, at least everybody, when I think about a quantum computer, I think of a chip, right? And we talk about it as bits and you even mentioned it kind of it's like silicon. We talk about electrons, but that all implies small. And then when you mentioned refrigeration, I was like, I didn't even think about that. For stability, it makes sense. We want to slow everything down so we can control the physics of what's going on. Which requires kind of lower temperatures. Right. We're taking out variables of the equation, right, when we start taking energy out of the system. But I don't think that that's something that people think about when they think of quantum computing, because a lot of people think of quantum computing as something that will replace traditional computing and, oh, I'm just going to have something on my desk, right? That's going to be a quantum computer. And it sounds like that that is something that is very, very, very far away if we ever get there.

Anne [00:31:02] So I think one thing that Intel is really trying to do is to transition quantum computing, which started in the realm of physicists and the physics lab into a computing technology that can be commercialized. And I do think that is one of one of Intel's goals.

Right? And that's one of the reasons why we're thinking scalability. We're thinking about how to create a true system architecture around this type of machine. But it's really and it's why we're also really reaching out to computer engineers because, frankly, if we want to move this technology from the physics lab into a computing technology, we need the help of computer engineers and we're doing our best to leverage the skill set, the computer architecture skill set that we already have at Intel.

Tony [00:31:59] One of the things that we talk about a lot on this podcast is A.I. Solutions, neural networks. And obviously these things are big. We did a whole podcast on ChatGPT and GPT-4 and how big it's going to be in terms of compute resources required. Is there a place where quantum computing will overlap with kind of the AI deep learning space as it exists now to be able to solve problems in a more efficient way?

Anne [00:32:27] So quantum computing can compute, do the computation, for machine learning quite quickly. There is one caveat, and that is if you have data sets that classical datasets of any size, it can take a long time to actually load your classical data into a quantum machine, to actually encode your classical data into a quantum machine. And that is a challenge that people are working on right now. So at the moment, I wouldn't say that quantum computing is the best solution for a data set, a data training set driven machine learning problem. But once you get the data into the quantum computer, the computation is is fast. So that's a definite maybe.

Tony [00:33:26] And that's the same thing in the AI space, right? Which is, AI can solve a lot of problems given that you have everything formulated the same way. But one of the things that we talk about a lot of is that the the pipeline of coming up with an AI solution is the challenge, right? The part of I have a neural network, I have a lot of data. The data is all loaded. How do I run my neural network? That part is computationally expensive, but pretty easy assuming you've done all the hard part, whereas the hard part is getting even in the traditional AI workflow, getting all of your data organized in a way that I can load it into my traditional AI model, right, with kind of my vectors of data and things like that. I don't know what the representation would look like to a, a quantum computer. Like, how would I get it into the right qubit space for lack of a better term? But I think it's probably a similar challenge there, although there's a lot more it's more, much more well understood when people think about kind of vectors of data. Yeah.

Anne [00:34:25] Yeah, yeah. So yeah, it's, I think it's, I think it's going to be tricky, but if someone solves this problem, which they probably will and it becomes easier to you to use classical data with a quantum machine, then who knows? Things may very much open up for quantum and as you said, we have time, right? Because we're going to need to build a bigger, much bigger machine. So there is time to be doing research. And all of this.

Tony [00:34:54] Where you hope quantum computing is going in the future?

Anne [00:34:57] I hope that we are successful at bringing quantum computing out of the physics lab and into a commercial computer engineering technology reality. I would like to see a very large scale million qubit practical quantum computer that can do things that classical computers cannot do, and this means to do useful things that users actually would really solve problems that users really want to solve. And I think that it's a very exciting area to be working in, and I think that we'll get there.

Tony [00:35:45] I hope so. And with that final thought, I think that ends our time for today. I'd like to thank Anne and you, our listener, for joining us. Please join us next time when we talk more about technology and trends in industry.