**Session 3**

00:00 → 01:48

**Shantanu Chakrabarty:** Okay. Let's get started on the next session. Hello, Hello! Hello! Hello! So the next session is on materials and devices for neuromorphics and architectures. Okay? And so our list of moderators, presenters and panelists are shown over here. I'm Shantini Chakraborthi. I am a faculty at electrical and systems engineering at Washu and so our next speaker would be from Ucsd. And then, following that, we will have a talk by Tony Lewis. From Brainchip. And then Deji Akinwande from UT. Austin. Okay, so with that region.

01:49 → 24:53

**Duygu Kuzum:** Hello, everyone. Thank you so much for staying with us and attending the second day of this exciting workshop. I am Duygu Kuzum. I'm from UC San Diego Electrical and computer engineering department and also Cowley Institute for Brain and Mind. Today, I would like to tell you about my research on neural interfaces based on multimodal approaches and neuromorphic co-design. So when we think about neural interfaces, we probably think electrodes that are implanted in the brain, they're recording spikes from neurons. And in this chart, we have actually number of simultaneously recorded neurons over the years by using electrophysiological recording technologies. So my background is in semiconductor electronics. And whenever I see such data, which is log. I cannot stop myself from fitting it into Moore's law. And in this case it's Moore's law for electrodes, right? So in this case. From this exercise we learned that, like neural electrodes, we have come a long way, but we are still limited in terms of the number of neurons we can record. And probably we are limited to where we are today, like about 1,000 neurons recorded by neuropixel probes that are developed by a semiconductor foundry. So, on the other hand, thanks to brain initiative and the excitement from the neuroscience community, there has been great progress in optical technologies. So the number of simultaneously recorded neurons by using optical imaging techniques have doubled approximately every 1.5 years matching the speed of Moore's law for electronics using advanced optical imaging techniques. We can actually image tens of thousands of neurons across the 3D volume of the brain, as you see in this nice example from Waziri Lab.So what are the challenges in neural interfaces? So if we start with electrophysiology, if we are interested in recording as many neurons as possible 1,000 neurons, we need to implant many electrodes into the brain. So this is what my former advisor, who was a neurologist, used to call this shish kebab configuration, which actually requires bulkhead stages. It induces a lot of tissue damage, and it's not acceptable for long-term chronic neural interface and electrophysiology, although it provides great temporal resolution. We can detect spikes with high precision. The spatial resolution is limited, due to 3D. Inaccessibility of the brain. Optical imaging is a great technique to look into large populations of neurons. We can image 10 cellular resolution, cellular level, spatial resolution, but that requires large and expansive microscopes and head fixed experiments which limit the behavior of animal models, and in addition to that temporal resolution of optical techniques is limited, due to slow kinetics of indicators and also low frame acquisition rates of microscopy setups. So all of these optical electrical methodologies. They generate huge amounts of data, but the neural entire phases are actually low throughput. They offer limited channel count. They generally do not have built-in signal processing capabilities that leaves us with the option of offline processing of gigabytes, or even terabytes of data to analyze neural signals in summary. We do not have a single single methodology that can provide both high, spatial and temporal resolution across large areas and deep layers of the brain. So to address these challenges in my group, we are following a holistic approach. So we combine temporal resolution of electrophysiology with spatial resolution of optical imaging. In the same multimodel experiments. We also developed neuromorphic neural interfaces to help with the signal processing aspect, to achieve high throughput signal processing and real-time detection of neural features to be used for closed loop operation in brain interfaces.And finally, in order to harness the complementary advantages that are offered by multiple modalities, we actually developed neuromorphic computational models to improve the resolution and also record and understand the neural activity across multiple spatial scales. So in order to do multimodel experiments without any crosstalk between modalities. We cannot use conventional probes. So my lab has been developing transparent graphene based microelectrodes over the years. Transparent graphene electrodes can be implanted in the brain because they're optically transparent. We can actually perform multimodel experiments which seamlessly integrate optical imaging, optogenetic stimulation and electrophysiological recordings so for multimodal experiments. We develop both Planar and Laminar type of microelectrode arrays, and the key advantage is actually monitoring neural activity at multiple spatial scales. For instance, in this example, we can record neural activity from the surface of the brain cortical surface by using a high density, transparent microelectrode array. So as this microelectrode array samples local field potentials from the cortex with high resolution across multiple regions, we can simultaneously monitor the activity at deeper layers of cortex using two-photon microscopy. So we look into local field potentials. Here we see some sample traces, and simultaneously we monitor spiking calcium spikes from hundreds of neurons, from deeper layers of the cortex. Alternatively, we can do the opposite. So yesterday we heard from Victor that activities and network scale, we really need to look into multiple areas so to achieve that, we can actually image the activity across the entire cortex, dorsal cortex, using wide field calcium imaging. But then we can actually use intracortical electrode arrays and implant them to deeper layers of cortex, or even subcortical structures. To sample spikes or high frequency oscillations such as sharp ray ripples. So in this example we are using a transparent laminar probe array that can be implanted in the hippocampus, so it can have up to 228 channels per shank. And it actually adopts microfabrication techniques from silicon Cmos technology to route the wires at different layers and have electrodes in multiple layers in a 3D configuration. And in these experiments our goal was to understand, when we actually record the neural activity from the hippocampus. What regions in the cortex that the hippocampus communicates in the context of sharpway ripples and memory consolidation. So such a multi-scale multi-model approach can enable us to answer very specific neuroscience questions and target them by techniques that are not possible otherwise.So like for these multimodel experiments, we always use high density, micro electrode arrays, but they, the existing neural, like data acquisition setups. They offer only like limited channel counts, and also they do not scale with the numbers. Even if we actually build very high density arrays. We do not have the interface to record data with them. So there is increasing interest in the general field of neural engineering to develop neural interfaces which can make use of different types of circuit components. Somebody's calling. I don't know. Okay. So the neural interfaces have the capability of specific neural signal processing, such as, based on fpgas application, specific integrated circuits or microcontrollers. So I prepared this table 2 years ago. It's not the state of the art anymore. So we have already heard some great approaches yesterday. So in this table, we look into different types of techniques or methodologies. For example specific neural signal processing. And if we look at all these approaches using memory microcontrollers, Soc or Fpga, they all focus on different types of tasks. So because these tasks are very different from each other, like EEG features, detection, tremor, detection, seizure, prediction. We cannot really cross. Compare them in terms of performance, like the area or power. So that is not definitely my intention to reach a conclusion by just looking at that table. But if you look at all these tasks like spike sorting is probably the one that is most widely used in neuroscience, in basic neuroscience studies. And it's probably the most challenging one in terms of computational complexity. So my group works on neuromorphic devices, memristors. And we were interested in investigating and targeting spike sorting as a computational problem to achieve like more area, efficient and high throughput approach for parallel sorting of spikes in real time.So we develop neuromorphic brain interfaces based on resistive RAM resistive RAM is also known as memory stirs resistive random excess memory. So the schematic on the left actually summarizes the spike sorting pipeline. We use it. So we record neural signals by using neural probes that I showed in the previous slides the neural signals. They go through amplification and digitization, and we feed those signals into a crossbar array of memrysters to actually perform spike sorting, and once spikes are sorted that they could be used for any closed loop, application or bci application. So what is spike sorting? Like when we record when we use a high density microelectrode array, each channel of the micro electrode array basically records a signal which looks like this, which is actually a spiking activity that is generated by multiple different neurons. So spike sorting is actually a clustering algorithm that separates this recorded signal into single neuron activity. So that at any time point we can know that this spike belongs to neuron 2. Both of them look like neuron 2. 1 of them should be neuron 3, and this spike belongs to neuron one. So this way, we can actually repeat it for all the channels. But as the number of channels increases for high density. Neural interfaces like that actually require a high level of parallelism and also scalability.So can we actually make use of non-phone Neumann architectures to offer a parallel that is needed for efficient spike sorting. So if we consider, like traditional von Neumann architecture, we know that memory and processor are separated by a data channel. So every time you need to process something like separate spikes do sorting, we have to reach out to memory. Read the data, process it and write it back. So that communication between memory and processor is the main bottleneck, and that is the main cause of the inefficiency. On the other hand, if we actually follow a man von Neumann approach, which is what I call neuromorphic, I'm sure there are people who may not agree with that. So we actually bring computation in the memory which is called in memory computing, computing memory, or processing memory in the devices field. So in processing memory or computing memory, what we do is actually, we do not store only data in the memory, but we use the memory itself to perform computation. How do we do that? So one example could be matrix vector multiplication. So if we have a memory array consisting of like memory stir devices, Rm devices that are assembled onto a crossbar structure. What we can do is we can actually convert the inputs into voltages and then convert the other matrix or the weights into the conductances of individual memory elements. And when we apply voltages, the current that flows across each element would be the multiplication of inputs with the synaptic weights. And they're summated across the vertical lines through Kirchoff's law. And by actually applying all the inputs across this crossbar array we can completely parallelize and compute matrix vector multiplication all in one step. So this is not specific to spike sorting. It's widely used in memory computing. So we are just adopting it for this purpose. It can also be used for any matrix vector multiplication for different types of algorithms, including AI and machine learning so conventional, like memories, devices that are developed for digital memory storage are not necessarily the best option for neuromorphic applications. The reason for that is those devices. Basically they are programmed by forming a conducting filament between 2 metal electrodes.But this physical process is extremely prone to variations and noise. So you can see that every time you switch this memistor, the switching voltage is different. So that level of variation is not always tolerable by algorithms. And these devices because they form a conducting channel, so they actually their resistance is very low in the on state, resulting in a very high power consumption and limitation for parallel operation across the crossbar arrays. And most of the time, because it's like an on/off type of device. So the number of states is limited. It needs to be used only for binary operations not suitable for implementation of on chip learning. So to achieve, like to develop a more like neuromorphic, compatible memeister device, we came up with the new device architecture called bulk. Rm. so in Balkaram the programming is done by changing or modulating the distribution of oxygen vacancies across the volume of the device. So this does not really form any filaments. It shows area type switching, which is very uniform. And we can program it to high resistance states which greatly helps with energy consumption, reducing the energy consumption. And also you can turn on many devices simultaneously. So you can have a high level of parallelism. And more importantly, you can program it in multi-level gradual states. So after 3 years of development, we finally actually came up with the ideal stack that can achieve bulk switching with the desired characteristics that are needed for neuromorphic computing. So our stack has, like a tunnel oxide, a titanium oxide layer with auction vacancy gradients which can actually be tuned to achieve bulk switching reliably. Once we demonstrated the bulk switching characteristics in single devices, we actually fabricated large arrays of relatively large scale arrays of Rm. Bulk, Rm. Devices at Utsd clean room. Here you see some sem pictures, half cut cells, and we are currently working on scaling these devices to a nanometer regime. So we have done extensive electrical characterization to understand the potential performance of these devices for neuromorphic computing. One characteristic which is really appealing is the uniformity of switching? So here, you see, like switching of the number over 50 cycles, which is showing almost no variation. In addition to that we actually can program it into multi-level resistance states. So in this example, it's like 100 levels. But we can go actually, 128, even 256 levels by actually tuning the device properties and programming conditions. So that multi-level gradual switching is very important for analog, like computing. We don't do analog computing just to store the parameters or weights with higher precision. And we can also use that type of capability to implement on chip learning with these crossbar arrays. So memory still, crossbars or Iran crossbars are a great template for making spike sorting more efficient. So to that end we actually developed a template matching algorithm in our template matching algorithm. We 1st like to store neuron neuron templates across the crossbar array. So if you think about it like a single neuron template. A neuron could be recorded by multiple channels like neuron, one neuron, one channel, one neuron, one channel 2. So each column of the crossbar array. They store templates for that neuron recorded by a certain channel.And we 1st like in our spike sorting scheme, we 1st do offline sorting a short recording and offline sorting to determine the templates. We store them across the crossbar array. Then what we do is actually we, once the storage is done, we actually apply the incoming neural signals into the like parallel rows of this crossbar array, and this way, by performing a convolution operation between the neuron templates and the incoming spikes we can actually assign, like, by looking by reading out the currents from the vertical lines, we can actually assign individual neurons like individual spikes to one of the templates that are stored across the array. So in this scheme, it's important to emphasize that all incoming signals can be compared to the stored templates all at once. So it's really a great opportunity to parallelize spike sorting operations.So we have demonstrated in hardware a 16 by 1632 by 32 crossbar array. And we looked into 2 different data sets, the synthetic neuronexus data set and neurofitum data set for both cases. We achieved a very high spike sorting accuracy. So if we are interested in using this for a very high density, microelectrode array. So we can simply increase the number of synaptic cores we can actually, if you use 12 of 256 by 256 crossbars, we estimated that we can process recordings from 100 channels within 4.8 microseconds consuming 30 to 50 times less energy compared to an Fpga approach. So now I would like to tell you a little bit about neuromorphic co-design. So can we actually use neuro inspired approaches to expand the spatial reach of neural recordings.So when we record neural potentials from brain surface local field potentials or echo microscopes. What we record is actually a spatial integration of many things, including extracellular potentials that are generated by spikes, activity, and every other processes going on underlying the electrode. And if we perform multimodel experiments, as we record these very complex lfts from the surface, we also image calcium activity from individual neurons, from deeper layers. So if we like these 2 signals, the surface potentials and cellular calcium activity in deeper layers, they are generated by neural processes, and they are correlated with each other through some nonlinear dynamics. Unfortunately, we do not have a biophysical model that completely explains what is going on, and how these 2 signals are related to each other. But we asked the question, can we use a data driven approach to learn the nonlinear dynamics between the surface and depth. By basically using a machine learning based approach. And can we use that model to actually predict the neural activity at deeper layers by only using surface potential. So you can see this as a way of extracting more information from local field potentials that are recorded from the surface. But the advantage is that you are trying to extract that information without implanting invasive probes to deeper layers of cortex. So we, apart from multimodel experiments combining electrical recordings that are implanted across the visual cortex. We imaged calcium activity in layer one and layer 2, 3, and in layer 2, 3. We have plenty of neurons. We can look into their responses, their selectivity to orientations, and their selective responses to visual stimuli at the cellular level. So we have this very rich data set. So what we did is we actually built a recurrent neural network model and trained it by using this multimodel data set. And the 1st thing that we attempted was so if my model is trained right, and if I use a new data set from another recording session, can I actually predict the neural calcium activity, average calcium activity at deeper layers in layer one and layer 2, 3. So in both cases we can see that our ground roots and the decoded predicted neural activity are very consistent with each other. Surprisingly. We found that, like, if you look at the decoding accuracy, most of the contribution to that was from very high frequency bands such as multi-unit gamma, and high gamma bands, which contributed the most to decoding accuracy. So we wanted to go one step further and ask questions that can actually decode individual cellular activity by only using surface potentials.So we added some dimensional reduction techniques to our recurrent neural network model. And we looked into the calcium fluorescence response that is decoded by only using surface potentials. So in this graph, you see 5 representative cells. So the Blake traces show the ground root that is imaged using 2 photon microscopy and the red traces. They show the decoded responses by only using surface potential. Can we predict every neuron? Of course, not. Every model has limitations, and in this scattered plot we see the correlation between the ground root and decoded or predicted calcium activity for all the sizes within the field of film. So this is a really compelling approach to actually learn more from local field potentials and expand the spatial reach of neural recordings beyond what is possible using single modality. So how can neuromorphic help for the future of neuroscience or neuromedicine? I think there are 2 directions. The 1st one is brain inspired models or neuromorphic computational models. So we are collecting a lot of data. We collect things like imaging electrophysiology. I mean, these are for human studies that could be fmri that could be chemical sensing. There are lots of sensory modalities and behavior as well. Perhaps we can actually look into developing models to integrate that information for a more accurate low, dimensional representation of neural dynamics which can substantially improve the performance of neural prosthetics or bcis and neuromorphic hardware can actually impact the way that we record neural signals by enabling on-chip signal processing in a more efficient area and power efficient way. And we can also, if we can also introduce learning capabilities, we can shift the processing based on the change in the neural signals that can substantially increase the longevity and stability of neural interfaces. Neural implants, as they will also show plasticity similar to the neural dynamics over time.So that is my idea about the neuromorphic future of neuromorphic computing. For biomedicine. I acknowledge my funding agencies. Thank you so much for listening.

24:54 → 25:15

**Shantanu Chakrabarty:** So our next speaker is Tony Lewis. So one thing. Could you please silence your phones? That was, you know, a message that was sent to us

25:43 → 54:24

**Tony Lewis:** So while we're getting set up. So I'm Tony Lewis. I'm CTO at Brainship Corporation. Publicly traded corporation. Before that I was head of AI and emerging computer at HP. I've also been at Qualcomm, where I led a pretty big neuromorphic engineering effort. You know, about a hundred people or so we're trying to build devices to transform artificial intelligence just before the burning revolution. Excellent timing. Let's see. So I have a soft voice. I apologize for that. I'm going to talk to you a little bit about the stuff that we do at Brainchip. So some solid data, a little bit about the architecture, a little bit about algorithms. And then also, I plan to introduce a framing of how to think about, perhaps neuromorphic systems in a slightly different way. It's something that I've been working with with my collaborator, John Tapson. If you've been involved in the neuromorphic community for a length of time. You might know John Tapson. He was a frequent telluride. And he and I, basically, we control the research and all the technology at the company. So we've taken over the company from the founders. I hope this isn't recorded. But they're actually very cool people. Okay, so, but okay, so so brain chips at a glance. You know, we claim to be the 1st to commercialize neuromorphic IP and produced a reference ship, 15 years of research..We have a bunch of data scientists, hardware software people, engineers were traded on the Australian Stock Exchange and we have a number of customers, including people who have purchased our IP as well as, you know, early access to chips or engineering samples, and and quite a lot of other things. And interestingly, we get a lot of traction in the space and aerospace industries they seem to like. If you say neuromorphic, you know they just love it. The overall architecture of Akita 2 point O is probably pretty something that you're probably familiar with. It's a distributed array of processors. The processors each contain 4 nodes. They share memories in various ways. So we call this near memory computer. They're connected together in a mesh network. And then we can. We can connect chips together, build, build huge arrays of chips. We can also connect them to A to a CPU. We're a digital event base at memory computation. And we use weighted spikes. So I get a lot of calls from people saying, Is this spiking? You know the system? I would say that it's kind of like a spiking system. But we also use wave spikes. We don't have a lot of dynamics. It's highly scalable. Each node is connected to the mesh network. Network, as I said, and inside each of these nodes is a tens processing unit. And so I'm going to talk a little bit about tens. And so you should know that that tens will be on a key to 2 point X. It's not in our current product. A key to 1,000 or a key to 1,500 and execution is all on chip, so it can execute independently of a chip unlike a dla or or a deep learning accelerator, which usually requires a CPU to drive all the action and takes a lot of energy. Alright, so, you know, for those of you who you're wondering. You probably heard about sparsity events, and like, you know, what is this stuff, and why is it good? Well. Let's see, have a point over here. So this is a classic frame based convolution. So you have an incoming image. Here you have a kernel, and then you can do a computer. And you can do this by just doing a basic multiplication. And so in this case you end up with 225 Max. Or multiply and accumulate operations. Here we have an event based thing. And what event-based does is if there's any zeros, any place, it just ignores that. Okay. And so it doesn't count that as an event, you know, through the mesh network we're only sending, like the ones the twos and threes and those kinds of things. And so in this toy example, we go down from 225, Max. To about 27, Max. And we're able to compute the same result. So that's like in a nutshell. That's what we do in the convolutional domain. Okay. Now, another simple example of event-based processing, and something that should get you to scratch your head and wonder why everything is an event-based processing is an example of Lebesque versus Riemann sampling. So this is everything you've learned in college about control systems about, you know, process digital filters, etc. You take samples of a signal at well- defined intervals. And so you notice that right here, this part, the signal isn't really changing very much, but you're still sampling, taking it, accessing that data. This is all wasted computation in the best sampling when you cross a certain threshold. That's when an event is generated. And so in this particular example, I think you have about 21 or so samples when you're doing Riemann sampling and about 12 here in this toy example. And so half as much data to process. And if you're if nothing much is going on, almost no data to process. So using this simple idea, you can see that you can be much, much more efficient than a conventional Dsp kind of processor. We also can. Now, there's limits to what we can do here. This is the event density coming through our network. And when the event density is rather high, we actually do a little bit worse than just a typical Dla. And the reason for that is that the event based processing takes a little bit of overhead in order to execute. But when we fall below about 40% or so of event density or about 60% sparsity, we start to get a very high relative efficiency. So this is sort of a magical property. So this is the best that a Dla can do. But as we get lower event density, we start to see that we're effectively computing far more than you'd compute with a DLA. And so the whole secret to making this work is to create networks which will throw off low event density. And so it's not just. And so the key message here is. It's not just about the cleverness of the hardware. But you have to have software that you run the hardware that can take advantage of the hardware. Otherwise you don't get any advantage. Yeah. So it's hardware plus software. Okay. So let me. So from a product point of view, let me tell you what I think matters. I think. You know. First, its algorithms. And that's what I'm gonna talk about today, because you can get a tremendous amount of this, then that toy example, I showed you that you can probably get, you know, a little bit of lift by having the right algorithms. And so I'll talk about this extensively. There's also software. This is the 1st thing that customers touch. And so a lot of times people, I think they tend to develop hardware and then add software later. And to be honest, that's that. You know, the brain chip approach initially. But really, at the end of the day software. You can't believe how important software is. We do have a pipeline which is called meta tf, which can take tensorflow like models and run them on our device. And that's a big Savior, because we're more or less compatible with conventional deep learning tool chains, Macs matter. So I view the whole world as Pico drills for Mac. So that's the compute efficiency, basically. And some people in the norm market community can get really incredible numbers for picojoules. Per Max. And I'll talk about that in a second size also matters, meaning that when you look at chips like true North or North Pole, these are massive pieces of silicon, massive frigging, pieces of silicon. And so you're not going to really be able to sell this to people? You know, if you, if they're in a cost sensitive area. So what we focus on is trying to get a balance between Max size software algorithms. And you have to balance all that to create a commercial product. And then, of course, data. We train most of our networks using open source data. But data is the thing that would allow us to build new models, particularly in this community. If we had access to data, we could build models very rapidly. Data is more important than you can imagine. This is a concept that I've been working on with John Tapson. He's the Chief development officer at Brainchip, and as I mentioned for a long time. Neuromorphic engineer. I'd like to divide the world. And this is the world of flow machines on the right hand side. This is where the future lives. And this is where the past lives. Okay when customers come to us, one of the 1st questions out of their mouth is, Yeah, but can you run mobile net? Can you run? You know, resent 50, and right, that's not what neuromorphic engineering is about. But you know you want that. We'll give it to you. So what's happening here is you? You imagine that you divide the world into a series of perfect photographs, you know, and there's they're coming at you at 60 frames per second, and each one you see anew. You don't have any memory of what has come before. And so you have to process that entire image each time you see it versus a streaming approach. You have a memory. You have a memory of what you've seen before, and you're just paying attention to things that are different. Okay, so that's the basic world of difference between this sort of stuff which includes, you know, all sorts of conventional things based on Imagenet. Any sort of Lstm's over acting over short frames transformers. Transformers, which are all new things. They're in the past. Okay. Here on this side, we have things which leverage state memory. And so those might include animals. So they have an internal state. And you're really using sensor information to update that internal state. So that's the key thing. You have neuromorphic devices which are naturally like that borrowing from animals. And then you have these long context state space models, and which I'm going to talk about in a moment. These include, Mamba, S. 5 models which have become all the hot new rage over the last year, and also our tens models which we develop in parallel, and we also already have hardware that can run these things? Okay? So any questions about flow machines, like, okay. Well, so you want to know, like, yeah, something about where neuromorphic lie. So that's my answer. This is for Sunny. Okay so tens is temporal event-based neural neural processing. It's a state space model. And I'll show you the equations in a moment. It can replace many transformer tasks, including language models, time series data, spatial temporal data. It dramatically reduces power, and it can leverage event-based computation. Okay, so this is an idea. We kind of stumbled on. One of our researchers, Lige Kung, was trying to model spiking neural neurons. And so these may have complicated dynamics. He has a background in theoretical physics. So when he looked at this, he said, Oh, well, this is. This is simple. We're going to model these, the kernels using polynomials. And there's 2 classes of polynomials which we're looking at. One are Legendre polynomials, the other one are Chebyshev polynomials. What you need to know about them is that they're defined over a certain narrow region, usually between 0 and one, or you can rescale them between minus one and one. What's really cool about them is that they have a recursive relationship. So you can see that the derivative is equal to previous computations, and so you can compute them in a recursive fashion. Same thing with Chebyshev polynomials. You can also compute them in a recursive fashion. These genre polynomials were 1st investigated by a fellow at applied brain research, and that's probably what kicked off a lot of the state space movement. But we found that these don't tolerate discontinuities very well, and Chebyshev polynomials worked out better for us. So the basic idea is that you can compute kernels. So the kernel is like you give a spike, you put a spike into a system, and then you'll get a response. And the shape. That response will be a kernel.Those things can be computed recursively. Okay, so the key thing here is that there's 2 ways of looking at kernels, either one. You see what the kernel actually looks like, or you compute it recursively. And so that's a big deal. So kernel mode and recursive mode. So now, when you're in convolutional mode, you can train these neural networks just like you do. CNN's, so if you're familiar with Lstms and artificial intelligence, it's really hard to train very, very deep models when we're in kernel mode. We can train incredibly deep networks that are recursive. So you know, 100 layers deep easily Lstms. It starts to get a little bit wonky. We also can run these things in a recursive mode, and they look like Rnns or Lstms, etc, they end up being very compact. Let's look at the equation. So if you have any control engineers out in the audience, you'll probably immediately recognize this structure. And so this is a typical dynamical linear time invariant system, where you have a state, which is X, you have an input U, you have this little matrix B, which transforms U, and then you have an A matrix, and typically that a matrix is something that is, it kind of dampens the signal, so to speak. Then you have a readout where you can take the state, and you can read out into the Y, you can also have a feed forward connection through D, so people discovered that they could create these very linear models and they could stack them together and make them into very large networks. And so the nonlinearity doesn't come in the model itself. It comes in y, which might be a relu or some other complex function, but which will take this and produce the input to the next layer. U, so, our guys discovered this, and apparently Albert Gu at Stanford discovered this. He got like 100 most influential people, or something like that. It didn't bother our researchers at all, I can tell you. And and and you know, we're all we've been franking away at this. And we've actually put this into hardware. Okay, so let's look a little bit more about some, some interesting properties. So this is our state space model domain. So this is the recursive domain. And then, when you unfold this. And so this could be the impulse response. And so you end up with a bunch of matrices which are stacked together. And you know. 1st they're squared, cubed, etc, out into infinity. And but what you see is that you know the original parameters in the A matrix get repeated again and again and again throughout the whole kernel. And so when you're training things in the kernel mode, there's, you know, if you were to do that, if you're to train all the matrices separately, you wouldn't be taking advantage of this internal structure of the A. Matrix. And so you'd end up with systems that end up being very not information dense in terms of their waste. If I can use the term loosely. And so there's a lot of room to take a convolutional model and then sparsify it, you know, create it so that it runs on neuromorphic hardware nicely. But I'd like to assert that this transformation from the convolutional domain into the recursive domain effectively does the same thing. It squeezes out a lot of that redundancy. And so now you end up with a very tight computation. And so, while we were trying to achieve the same thing that neuromorphic engineering might try to obtain just processing changes, not being and being very dense in your computations, we do end up doing it in a very different way which is based upon mathematics that everybody on the planet knows if you you know. Wow! Well, if you're a double e. It's kind of an interesting insight. So we have compactness and we have redundancy. Okay, I'm gonna skip over this. But you know. Basically, the other. Another point I was going to make is that the kernels end up being described in a parameterized fashion, and I'm only on Slide 14 in a parameterized fashion. But this basic idea is not new. If you're back in the 19 nineties, people were, you know, using Gabor filters which are also a parameterized form of kernels. But then, when deep learning came around. Those parameterized kernels were replaced by wernable kernels, and the problem with learnable kernels is that you end up with. If you want to describe the shape you have to have all these parameters instantiated. These are the weights at a certain region on a grid. And so you have all these parameters, whereas the shape of the kernel might be parameterized by, you know, in a much more dense way. And so that's another way of looking at, you know. Maybe this is an alternative to kernels. Okay, some of the use cases, you know. Audio. I'll talk about that generative AI industrial IoT. What we would really like to explore is this over here? So this is the unexplored region for us. We have a little bit of results there, not a lot. We're very interested in biomedical applications because we feel that we're going to be very low power and easy to train. Okay, so how well does this work? If you look at? You know the gesture recognition task for Dvs. Cameras. You know. We're state of the art eye tracking state of the art, denoising results versus conventional Dns, models or state of the art keyword spotting, you know, state of the art. Large language models. You know, for the we've only trained up to about 330 Megabyte tens. But we beat Mamba at over twice the size, and also mama models that are about 3 times the size in terms of perplexity. So this all looks very, very cool. I noticed that a lot of people showed pictures of signal processing pipelines, and they usually start with some sort of conventional pipeline over here, some translation of the raw signal into basically a picture. And this is an example of what you might, how you might process a conventional audio signal. You have to take a large slice. You convert it to an Mfcc format, which is basically a picture telling you certain aspects of the sound. And then you begin processing with tens. We just process the raw audio signal coming in, and you know we can tell you what the keyword is coming out. In terms of knowing our audio pipeline.Let me skip to the next one in terms of the keyword spotting solution. You know, we're keyword spotting a 10 word data set at about 0 point 2 5 milliwatts and 97% accuracy, which is pretty much state of the art in audio denoising. We're also doing this, you know, denoising of audio signals in 3.2 milliwatts. This is processing a 16 kilohertz sample. So if you're only processing data at a fraction of that. Then your power is going to drop by a corresponding amount.And then I'm just gonna this is like hopefully, this is gonna work. And people online can hear it. I can't promise anything. I'm CTO, but not a technologist. Okay. So on the left hand side. We're going to play some noisy samples, and then on the right hand side those would be samples that are cleaned up. Falling hot and fast between face and veil, for she had talked till she was very sorry indeed for herself. So could you hear the background noise? No. So falling hot and fast between face and veil, for she had talked till she was very sorry indeed for herself. Okay, so it's yeah. Pretty much cleaned up. Here's another one. The rainbow is a division of white light into many beautiful colors. Could you hear the background noise? In that case the rainbow is a division of white light into many beautiful colors. So it's pretty good. So I showed this at an investor meeting, and a woman came up to me. It was hard to hear, and she said I could hear the Denoise audio very cleanly. It was very nice and that is actually taking us into a different domain which may be another time I'll talk about, but it's called super intelligibility. Where you take the original signal, and you actually don't just denoise it, but you make it sound better than the original signal, the super intelligibility. So after you get past a certain age, I've been told. People start turning on closed captions when they watch TV. Well, imagine if you could just watch TV. And suddenly, it's crystal clear. Okay, would that be worth something to you? I don't know. The Rainbow is a division boat into many beautiful colors. Okay, come on. Not a technologist. Okay, so this is a devious camera. And it shows brain chips results versus you know, everyone from Ibm true north intel you know, etc. And it just occurred to me when I showed this video that my guys put in the clapping because they're clapping about their results, you know. But I didn't get that but we get a hundred percent accuracy on this gesture recognition task. So that's kind of cool. We also accumulate. This is something that I think people see in animals where we accumulate like evidence through time. And so this shows accumulation of evidence using a fast model and a slow model. And so with time we get more and more confident about our signal. If you think about gestures, you know, if you just make one part of a gesture. It's hard to figure out what you're going to do. You have to wait a certain period of time before you can understand the full gesture. And then this is us. You know versus say, Ibm true. North is they make an estimate here intel and I guess they didn't do so well probably. I imagine they must be doing better now and then some other competitors. But we end up doing, you know, pretty well for that particular task. No, hold on that. That's part of the memory. Yeah, that's part of the accumulation through this recursive you know. Mechanism.Yeah. Now, we're gonna turn to LM models. I know this. This sounds like it's pretty far. Okay, so we have LLM models. Now, you might be thinking, what does LLM models have to do with biomedical engineering or biomedical use cases? It's interesting because you can actually use it for a whole host of tasks including adaptation, etc. So you end up just sending a brief text, and that can be converted into our hidden matrix. And that hidden matrix could be used to drive adaptation, for instance. So there's a lot of stuff to be explored there. So versus mamba, which is another safe space model. You know our perplexity score, which is basically a score as to how well you hallucinate the lower the better. So we hallucinate better than Mamba. We hallucinate better than mama. Very proud of that. There's another portion to this which is the Mmlu task. And we haven't looked at those downstream, and those include all sorts of other things like, how many hours are there in strawberries, which for some reason it's fascinating to people in the machine learning community. But though we don't incorporate downstream tasks, I just wanna give that as a caveat. Okay? So this is an example of text on the right hand side. It's kind of a good example. That's why I put it up here. And the prompt is a deep warning, and then just starts going out. And so this is a point 0 point 4 billion parameter model. Okay? So I won't show this. This was finally able , anyways excluding, hallucinating how to use neuromorphic technology for biomedical applications. But I don't want to know. Okay, we'll move on. Okay. So I'm just going to skip to the very end. The key takeaway is that we have this concept called flow machines, which include animals, traditional normorphics and tens kind of overlaps with traditional normorphics and the concept of flow machines. And that's it. Thank you.

54:35 → 56:01

**Duygu Kuzum:** Yes, okay, thank you. So we're finally able to have that panel. So our panelists are the speakers. And also I will introduce our other panelists, Gina Adam, from George Washington University. Gert Kavenbergs from Ucs. And I'm hoping he's online. Oh, okay. Jennifer Hasler from Georgia, Tech and Ilka Angela from Worcester, Polytechnic Institute. So we are gonna start with a few questions and please feel free to walk to one of the microphones if you have any questions.So our guiding questions for the panel, I'll start with the 1st one. So we heard a lot about new materials like, how like, what are the opportunities in using new materials? How will these interface with the tissue? Are there any concerns in terms of biocompatibility? And how will they integrate with Cmos electronics? So this is an open question to any panelists. Oh, position statements. Yeah. I think I'm I. I need to change the order. So we'll start with the position statements. I'm sorry for that. So I had this one as the 1st slide. So guiding questions, we are going to revisit again. The 1st position statement by Shantanu.

56:02 → 57:35

**Shantanu Chakrabarty:** So this is just. I would lean into a topic that was discussed yesterday. That control is a very important topic, and I'm just doing a play on words from late. Dick Feynman, who said that? What I can't create I don't understand. Actually, I would say you can't control it. You also don't understand. And what we are. Actually, I feel that one of the things that we are missing in the hybrid neural interfaces is this cross mapping which also ties to embodiment? Do biological neurons treat silicon neurons as one of their own? Okay, and I will highlight this with one of the projects that is going on at Wash U, where we are trying to enhance the sense of smell of these bugs. These are locusts. And what we are asking is that, okay? If we add some silicon neurons into the mix does the system as a whole smell better? And if, indeed, if it's able to sense better, that means that the silicon neuron is now an embodiment of the biological neuron. So and the reason why this is important is that as Ralph has mentioned, this field of neuromorphics has been existing for the last 40 years. Every time you go to Telluride, they say, Hey, my silicon neuron is better than yours and everyone competes. But who else to ask this question to accept neurobiology. If neurobiology can treat this, treat this as one of their own. That means that essentially, that's the best silicon neuron that we have.

57:45 → 58:45

**Jennifer Hasler:** Okay. So basically, I just wanted to make the simple point that you know, I think when you know, we've had a lot of discussions about things like, how should we look at multiple applications? And I do say, at some point we start talking about the bio. You know, biological, neuromorphic applications or wearable, implantable systems. When we finally figure out what we want to do there. And I think we're starting to see different cases and applications there. Then we are gonna have to ask this quickly. Then I do think we need to make sure that we have the technology that's there to rapidly solve this part of that is a little bit of, you know, continuing to keep the all the technology running now as well as have tech techniques much like we're already used to in digital systems that I can quickly program change, or even use code to kind of approach those things, and those are certainly things that I care a lot about. So that's part of my both introduction and position statement, and also argue that this could very well be done entirely within standard. Cmos, if we wish.

58:59 → 01:03:00

**Gert Cauwenberghs:** I think it's important for us to go back to foundations. And we enjoyed the discussions yesterday and today, including discussions. What is neuromorphic engineering? And it's important for us to realize the way that covers are first described. In fact, they even launched the term neuromorphic engineering. It goes back to physical foundations. And it so happens that, despite the vast differences in the substrates that biology and silicon and other materials operate. There are actually very similar principles of play. The biophysics of, say, how ions move to ion channels versus how electrons and holes move through the channels of transistors, for instance, and that's all governed, say, by Boltzmann physics, I mean thermodynamics on national, I guess physics, and even at higher levels. So, for instance, at a network level. We all hear about this, the Nobel Prize in physics now given to AI people. Well, actually, really, this is true. Physics of thermodynamics that plays out in understanding how representations in neural systems, and how the collective behavior of so where you have interaction between the physics of the substrate. And then the environment plays out in giving rise to, I guess, energy-based dynamics and basically optimization with error functionals and etcetera. And so at each level of scale. Here you have those isomorphisms that can be exploited. And so the synergy that you have between neurosciences, analysis, and then engineering or neuroengineering synthesis also plays out nicely here, right? You have this top down and bottom up approaches that are very complementary. And that allows synthesizing. I guess, from an engineering perspective. As soon as you have a better understanding of how things operate at the scientific level, you can now start synthesizing larger systems. And so neuromorphic systems. Engineering has been this scheme that allows at each level of scale this morphism, right, this translating it into silicon and other materials for emulating the function. And then the computational system. Neuroscience can also close the loop, because in all it gives on a circle here. So you can start approaching full brain level interfaces, including the brain initiative, has been really prominent in doing this right? So the point here is that we have these really important synergies. I would like to make 1 point that may have been missed here in our discussions. Even though they are just talking about flapping wings, etc, right of how far? Of a detail they need to adhere to when modeling biology? Well, it's all about physics, physical foundations. And so, in taking advantage of understanding of, say, how the biophysics of neurons and the channels, and then the foundation, say, of how electrons move to ion channels. We have come to realize there are different ways of operating transistors. A threshold MOS operation that gives you much greater efficiency and actually closing the loop, we're able now, the greatest circus. Now for interfacing to the brain that scales actually, circuits that are inspired by neuromorphic computing. So those are really great. Advances in the efficiency of neural interfaces with noise, energy, efficiency is driven by physical foundations that are now really basic a key in this interface between silicon and materials and and and and policy. So I'll stop there.

01:03:08 → 01:07:10

**Ulkuhan Guler:** Hello, everyone! I want to convey my perspective through a biomorphic example. Thanks to the discussions about descriptions yesterday. So our recent research is focused on transcutaneous sensing and actually transcutaneous sensing research started in the late fifties, and, thanks to electrochemical sensing success. Now, these devices are available in intensive care units and nominated intensive care units in the shape of, you know, bedside monitors. What is transcutaneous, sensing, sensing oxygen and carbon dioxide molecules, diffusing through skin and correlating them with arterial oxygen and carbon dioxide. And how this is done, actually, in traditional techniques is applying heating. So if you look at the pink curve here by increasing the blood flow and diffusion. Oxygen and carbon dioxide. Partial pressures are approximated to blood arterial partial pressures. So what we do today, we actually developed several prototypes based on reminiscent sensing and in the form of patch or watch. We don't apply heating. And if we don't apply heating how we are gonna approximate oxygen and carbon dioxide on the skin to oxygen and carbon dioxide in the arteries. So well, we have to do this. Otherwise conditions will not understand what these values are. Right? We do it through this biology and physics based modeling. And actually, we create estimation algorithms based on computational models. We model tissue layers, layer diffusion rates, boundary conditions, and everything. Thanks to Nsf and Nih for acknowledging the value of this research, and I also want to highlight that since we have a tool now, actually, we can, you know, parameterize these models because we know that trans continuous sensing depends on factors such as gender age. Pmi. So with that, actually we will. This will give an opportunity to personalize a precision house and then I have a question. If we don't do this this way. What would be the approach we would follow? Okay, so maybe we can take it in the question section, okay, this is very quick. Oh, Gina, I'm sorry. Yeah, this is. This is just very quick. Doable. Because last year, oh, sorry. Yeah, you can take the question. I searched for Gina. Okay, cool. So basically, let me try very quickly. We would have to, you know, get a lot of blood samples from arterial lines, right, which is very difficult, and we have to correlate them with our sensor data. And I don't think it would be really quick, because, you know, getting an arterial sample is very, you know, prone to a lot of, you know, hurdles and we would apply brute force machine learning right? So if we don't have enough data. And if we know physics and biology, then another approach would be actually, you know, creating computational models.

01:07:13 → 01:08:45

**Gina Adam:** Yeah. I just wanna say 3 things. The 1st one is, we need the data sets, and it has been brought up before. To identify. What are the champion applications where we can identify the neuromorphic advantage and contribute, contribute to that. So hopefully, as a community. We could come together to build those data sets and help the co-design from materials. Devices all the way to native algorithms that could potentially do computing locally, such as cellular neural networks. I wanted to emphasize that closed loop between sensing, computing, and actuation. And the second point is, we'd need metrics, and it has been discussed with energy efficiency, and compactness. But I also want to bring up the issue of robustness and robustness of the hardware and the software to noise, input noise, hardware, noise, etc. And the 3rd point is, that kind of tie. Everything together is interdisciplinary collaboration and education. And I know this is sponsored by NSF. As well, and having the students be able to learn from different fields and down the line, help bridge the gap.

01:08:46 → 01:09:28

**Duygu Kuzum:** Thank you. It was a great summary of your beautiful slide, unfortunately, but you all imagined how it looked so we have, like a few guiding questions due to time limitation. Ralph is answering all the questions we can run to about quarter past. Maybe we can have one question, one person answer, and if you, if anybody from the audience likes to ask a question or answer, you're welcome. So what are the opportunities in using new materials? Device technologies? What are the concerns for like in terms of biocompatibility or integration with existing cmos electronics?

01:09:32 → 01:10:22

**Audience:** Yeah, I think, I could address this, since we work on materials and devices. And it seems like for a long time we think of organs or the human system as a target to learn from a target, to probe a target, maybe to treat some disease. And our thinking is, can we make the human or the organs part of the device rather than just a target? So this is what is driving some of our research where, you know, I showed, you know, using the skin as part of the composition of the transistor itself. What can that enable in terms of advancements in complexity, in instance, or processing and those sorts of things.

01:10:30 → 01:10:35

**Moderator:** This is a very controversial question. Is neuromorphic. The only solution to power bandwidth throughput problems is to implant the same plans.

01:10:36 → 01:11:25

**Shantanu Chakrabarty:**: And I think this is where the notion of neuromorphic advantage has to come in. Because, we have to show that for a given benchmark, neuromorphic is one of these solutions. And so some of these paradigms, like computing memory. All these things. You have to make a case other than just low power latency is latency a key or not? If latency is the key, then yes, neuromorphic is a solution because you have to do things using the physics of the device fast enough in a closed loop manner. So there are these certain applications where? Yes, neuromorphic is maybe the best solution. But there are other applications, and there are projects in my group where, hey, you can just use a standard off the shelf, Ml. Algorithms, and they work just fine.

01:11:26 → 01:12:30

**Gert Cauwenberghs:** And I totally agree, Chantano. So too often, neuromorphic engineering is made synonymous with you having a spike in your network with smp learning right? But there's much more to neuromorphics. So going back to the foundations, what Carver meat told us, listen to the technology, listen to the media right? And so when you interface materials with biology, of course, and with silicon and material in between, it is important to optimize that dialogue, this interaction between the different levels of the stack and optimize the circuits that interface this efficiently. And again, synomorphic engineering has been a guideline for extremely efficient and resilient circuits. And that's what you need, because you're putting a really tight power budget. So. But that token, because very aggressive environments with very, very tight power budgets, better token definitely inspiration from biology in building very efficient and resilient circuits can definitely help.

01:12:31 → 01:12:52

**Moderator:** Okay, thank you. So our 3rd question is, what are the challenges in manufacturing neuromorphic chips for biomedicine. So these neuromorphic chips are more mainly like this question targeting large scale manufacturing. So maybe I can ask Tony's help to give us an industry perspective.

01:12:55 → 01:13:47

**Tony Lewis:** So you know, obviously, we produce all our chips and we target sea moss because it's widely available. There's people that know how to do it. It's not a big deal. If you want to do something more exotic. It might be a little bit more difficult. Tying back to the last question. You know, it's typically much easier to design things, you know, digitally synchronously and that may take us away from other principles in neuromorphic engineering. Perhaps for us to really fully realize all the potential of neuromorphic, better tool chains will have to be developed. So we can synthesize more exotic things.

01:13:48 → 01:14:22

**Panelist:** Those are great points. And I wanna add here, the challenges are like split foundry when we have to get chips from the foundry, and then we have to add our own processing on top, or we have to add our own devices or our own integration. That can add significant challenges and how to best facilitate that particularly for biomedical applications that have very tight requirements. Can be a big challenge in a big burden.

01:14:23 → 01:15:09

**Audience:** I just want to add 1, 1 or 2 more things on. Just the fact that you're dealing with at some point going to start up from a sensor approach. You're going to have some analog computer computing of some level doesn't matter. There will be some. And then you're going to probably have some digital back end. You've got to be able to have expertise in all of those, and you've got to find a way to be able to parallelize that. So, for example, if you had like a common platform that was, maybe you know, analog or reconfigurable, or had some digital structures that would allow you to be able to build your sensors and your analog hardware in parallel could be huge.This is a huge, because usually any one of those things becomes a major bottleneck. All of them together really decreases the probability of something happening on a time scale you're hoping for.

01:15:10 → 01:15:25

**Moderator:** So what are the benefits of on-chip learning? Adaptation for neuromorphic hardware, especially biomedicine. Maybe doing who you want to take swipe it.

01:15:26 → 01:16:34

**Panelist:** I think the benefits would depend on the target application for neural sensing. We know that neural signals shift over time; they drift over time like, it's due to a biological response. It's due to neural probes moving. So there are many biological and non-biological reasons for that. If we can actually develop like algorithms that provide on chip learning and do it in an efficient way, using like on chip learning architectures, it may be possible or correct for those types of like shifting in neural signals like, for instance, you can't think you're you're recording your spikes in 3 channels over time, those channels shifts to another location due to physical moment. So if we can have adaptation on chip adaptation to drift off the neural signals that could really benefit like long term experiments, so that we can track the same type of neural dynamics over time. Study, behavior, study, cognition, both in Vigo, like, both in animal models or also in humans as well.

01:16:35 → 01:16:56

**Audience:** So I can just add a very quick thing about variable devices and human physiology monitoring. So, as I mentioned, there will be interpersonal variations, but also interpersonal variations are there. So if you are considering a long term, then, some of these parameters will definitely shift, and you may wanna adjust them.

01:17:00 → 01:17:21

**Gert Cauwenberghs:** Yes, and from a circuit perspective. The additional advantage of adaptation is also being able to circumvent its own imperfections. Right of the analog hardware, especially as you want to get extreme efficiency and resilience of the media. So adaptation is key for making that happen.

01:17:24 → 01:17:38

**Moderator:** Okay, last question standing between lunch. And this is okay, what are the challenges in translation of neuromorphic to clinic? What role should industry play in that. Maybe Tony give you that.

01:17:40 → 01:17:42

**Tony Lewis:** Yeah, we haven't transferred anything to clinical applications, so I'm probably the wrong person to ask.

01:17:43 → 01:18:28

**Moderator:** But maybe I can piggyback on that question right? Because that was what I was going to ask right? Which is wonderful, that we're getting an industrial perspective. And then we'll have some in the afternoon as well, right? But so far, if you think about it, you know neuromorphic chips are being commercialized, you know. Be it you know, whatever the application is. It hasn't, you know, like you said. You know, the true North was way too big, too expensive for it to be commercializable right? In your case, you know, in the branch of scenario, it seems like, you know, you guys, you know, are producing products of some sort. Right? So what is different? And what would be the future in order to make it, you know, more commercially viable.

01:18:30 → 01:19:15

**Tony Lewis:** I'll just answer from what comes to mind is a business perspective. I hope that doesn't offend anybody. But when you look at things that have a long payoff it becomes very difficult to invest in them on your own. So if you look at the discounted expected value of a reward in the future and you apply heavy discount factors. It's pretty much like that. You limit yourself to a horizon of maybe 3 to 5 years, and so if you're not going to get a payback in 3 to 5 years. It's difficult for a company to invest. And so maybe that's where that's 1 of the roles of government investment to help, you know. Help us with that risk.

01:19:16 → 01:20:04

**Audience:** I'd like to amplify something you said so, because, thank you. No, I think you're right. I think there's some different metrics on the commercial side, one of which you know. So I think there's different questions you ask, one of which I think you actually brought up as a slight comment was the, you know, the size of the chips, because sometimes what you need in a particular application, a particular size or a particular application, is not a gigantic thing, but something smaller, something that fits into that space in that particular niche application in that particular cost. So I I think that there's some different metrics you have depending on how you're trying to commercialize some things at least in my experience.

01:20:05 → 01:21:03

**Audience:** Yeah, just adding that to adding to that like, when we think about neuromorphic, it's not a device. It's not an algorithm, it's not a circuit or architecture. It's a co-design of everything. So when we actually want to optimize efficiency, if we develop the most energy efficient device. But if the circuits are actually not optimized for that type of operation. Then we're going to lose the advantage. If you have a neuromorphic chip. Hopefully, Tony will provide us for neural signal process for free as part of this conference. So that's why we were invited. So it's basically that we need to also optimize the algorithms to map onto that hardware. Right? So what we are doing right now is just using computers. Fpgas and it's an algorithm hardware co-design. So it's very important to like, take the most advantage from both software, the hardware and optimize energy efficiency particularly.

01:21:04 → 01:21:22

**Moderator:** And I think also yesterday there was a question that was asked about the regulatory pathway. So what are the regulatory, regulatory pathways for AI and neuromorphic related architecture? So can we guarantee reliability and issues like that? Anybody from the FDA?

01:21:23 → 01:21:47

**Audience:** Yeah, one quick question. After listening to all this, I was wondering about privacy as an issue. Will that come up at some point? Because, reading all the signals later on would be easy, especially if people are using Bluetooth or something to communicate. So any thoughts on privacy before it can be commercialized?

01:21:48 → 01:22:23

**Audience:** Yeah, this is not my area, but it brings up an important point. I mean privacy and security. Right? What happens when you might have an implantable.A device that gets hijacked. And is there a way to have, maybe for it to be bioresorbable? That was actually something that I thought about the materials that our keynote speaker brought this morning that can actually be resolved by the body. If there is an issue of security.

01:22:24 → 01:22:39

**Audience:** So these are bio devices, and it's a very fine line, because we don't want them to, you know, stop operating. So functionality, proper operation. All those things are really important.

01:22:42 → 01:24:05

**Audience:** So as the token investor and an engineer in the audience. I just want to sort of get the 3rd point across. So it was very subtle. Right? So what is the cost? I don't think anyone really understood what the wafer start cost to make a chip actually is. Okay. So there was a couple of back and forth about it. This is very expensive. It's if it's 1 off. It's very difficult. We need government support. It costs between 2 and 30 million dollars to start a chip. Not even make one just to negotiate with the Fab to start a chip.So when Jennifer, for example, says she runs on standard Cmos process. I'm not sure that you all realize just how significant that is. It means you're not asking them to do something special. You're not more towards the 30 millions dollar side of the equation. You're more towards the 2 million dollars side. This is a big deal. It's very expensive to make a chip. Okay? So compatibility I and you know, working with the standard processes. This is a really significant thing. And then the ability to actually directly interface with analog to what I've heard over and over again versus something that's a purely digital chip. That's an enormous hurdle to get over. So there's a couple of things there, I think, in the cost that you really need to touch on in more depth.

01:24:06 → 01:24:12

**Moderator:** So that is one area, Gert Commonwerks and I also struggle a lot. Maybe Gert, is there anything you like to add?

01:24:13 → 01:24:57

**Gert Cauwenberghs:** Well, 30 million is a large number. But I think cost is all a relative issue, because in academics I know we can get access to semiductor technology with multi-project wafers on the order of a few KA few right? $1,000. Get it done, of course, for a large volume production. It gets tough, especially if you're going to go deep submicron. But there are vanilla type processes that are available actually, that work perfectly well that are much more affordable. And with the Chips Act there's a lot of push these days for making that available to the masses, to the whole industry, and I think we'll see some outcomes of those developments, too. I don't know.

01:24:58 → 01:25:59

**Audience:** So I want to add one more thing on this, just from a commercial side of this. Let's say you know one of the issues. If your mask costs are going to be a certain number, let's pick. Make it an inexpensive process. Call it 100 K, we're doing real well. It requires you're typically on a from a commercial side, gonna spend about 10 x in in personnel cost in industry to make this to make that fabrication, and from a from an internal company, whether it's small company or large company, it's gonna have to be about another 10 x up from that, whether you're actually gonna bother even building it in terms of the product if you're gonna get unless you're gonna get that much revenue. So some of these economic questions are really really critical. In terms of where you can go with it. By the way, that's why you see digital chips are processors, GPUs that can be programmed many, many times. They have that same constraint. They can't just randomly go build something.

01:26:00 → 01:26:17

**Audience:** Yeah. And that's why I think, having champion applications with specific metrics and targets really could help narrow down what chips should be made.

01:26:24 → 01:26:20

**Moderator:** I think we probably should close. Okay, so thank you. Thank you all for asking nice questions. Thanks to the panel.