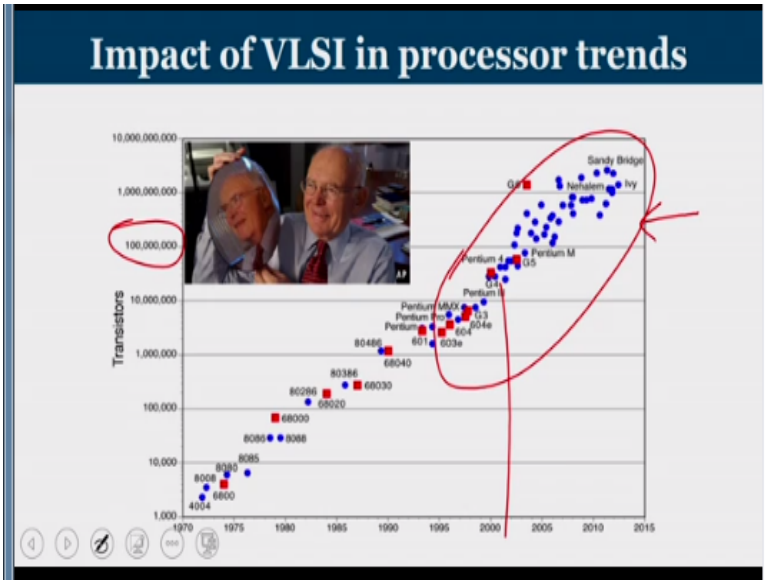
Advanced Computer Architecture
Prof. Dr. John Jose
Assistant Professor
Department of Computer Science and Engineering
Indian Institute of Technology-Guwahati

Lecture-29
Tiled Chip Multicore Processors

Welcome all of you to lecture number 20, where we are talking about tiled chip multi core processors, we have seen over the last lectures about what is processor, what is pipe lining and then different levels of memory from cache memory to main memory to secondary storage. Now we move on to slightly the advance architecture concepts, which is about the operational principles and techniques of modern micro processors which is having more than one processor inside a chip.

And they are broadly known as multi core processors and special category of multi core processors it is called TCMP tiled chip multi core processors.

(Refer Slide Time: 01:08)

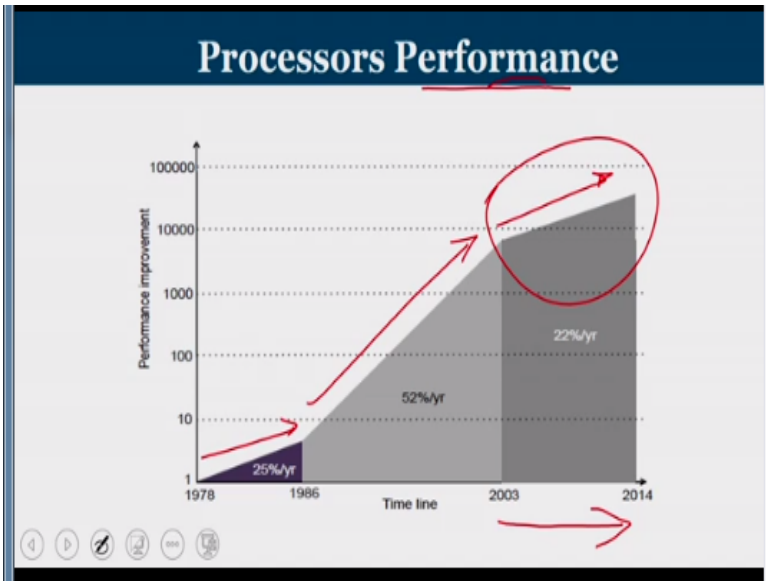


Over the years, we have seen there is a tremendous growth in the number of transistors that are inside the chip. Gordon Moore predicted in late 70s that the number of transistors that can be kept inside an IC will double in every 18 months. So we have already touched the 1 billion mark

somewhere in 2003 and you can see that these are all the modern micro processors which has over 10 billion transistors inside a single chip.

And what is so peculiar about having more transistors inside an IC the more the number of transistors, we can have more functional unit, more computation power can be given to them. So because of that we were able to integrate many features that your outside traditional micro processors like timers, interrupt controllers, peripheral ICs (()) (02:05) DMA controller these things now find space inside your modern microprocessor because of abundance availability of transistors.

(Refer Slide Time: 02:16)

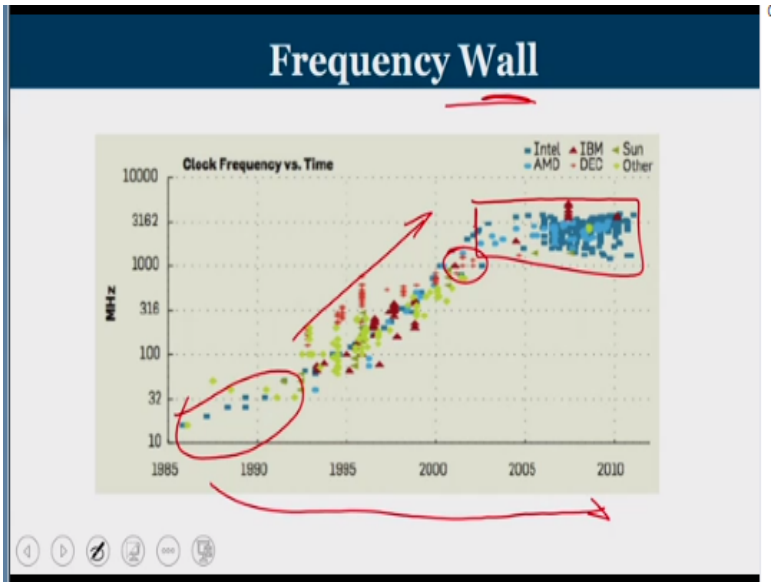


Now if you look at the performance of processors that were there in the market right from the early Silicon Valley era, you could see that in the initial years, processors were only 0.25 times more powerful than the previous generation processors. And then we saw a steep hike where roughly 0.5 times performance improvement is being observed in the processors and then we were not able to scale up to that level even though our technology is improving.

We are having lot of transistors, but somehow in these region we were not able to improve the performance of unicore processors. The whole thing is about performance of unicore processors let us say in terms of throughput in terms of the speed up some other related parameters, which all will be going to impact the performance. Now let us try to see why there is a region like this,

where we were not able to exploit the most conducive environment of having more number of transistors in the chip, but directly we cannot convert it into performance.

(Refer Slide Time: 03:24)



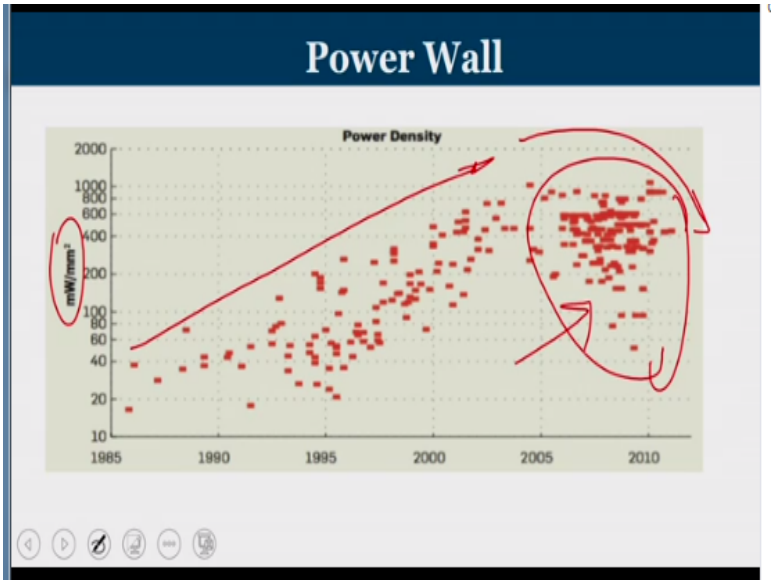
One of the biggest limitation is the frequency wall, we know that over the years this particular graph shows the clock frequency of various processors across the time in which they are introduced in the market. So initially processors were around 10 megahertz to 50 megahertz in the initial late 80s and early 90s. And then we could see a steep hike in the operating frequency, which was touching close to 1 gigahertz somewhere in 2003.

And then the trend in continue the linear increase in frequency of processors was not continued and somewhere now we are stabilizing around 3 gigahertz. Our general notion of improving performance of processor is by employing faster clock, the more fast the clock is more number of instructions that I could complete in unit time. Now how can I mitigate that but we know that frequency has another demerit.

If we increase the frequency then it is going to have an impact in the overall power dissipation. So it is always advisable to operate a lower frequency, if you wanted to take care of the power dissipation aspect. But it is we cannot go for lower frequency because it is going to reduce the performance in terms of throughput. So it is a balance that we have to make in terms of frequency as well as the throughput keeping power budgets within our constraint.

So this frequency wall that we have just seen shows that if you increase the frequency over the time it is well and good but somehow there are certain factors which limited us not to increase a frequency and now frequency is revolving around close to 2.5 to 3 gigahertz in all modern micro processors.

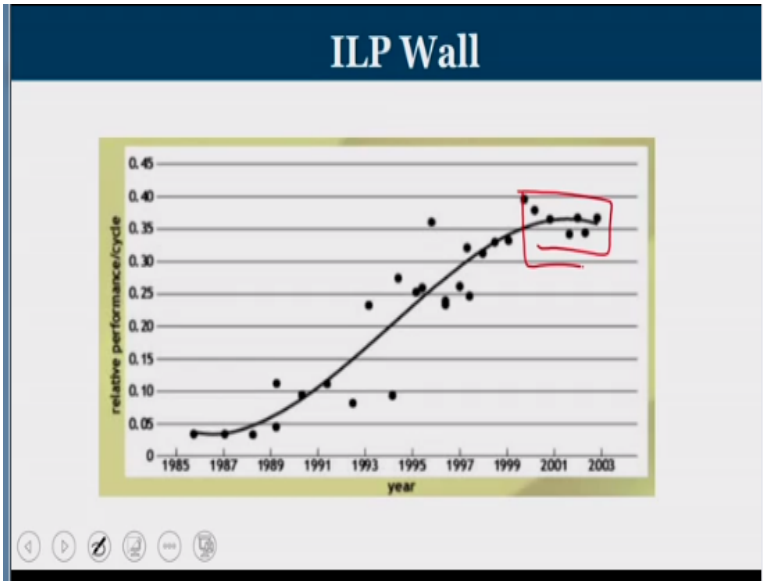
(Refer Slide Time: 05:22)



The second aspect that is drawing our attention is over the years this particular graph is showing power density milliwatts per millimeter square on processor IC, the amount of heat dissipated on various processors that were there and you could see the power of transfer going on a very high node. And now in the recent years we have found that we have come down, so adding more functional units more intelligent unit they were all going to consume more power.

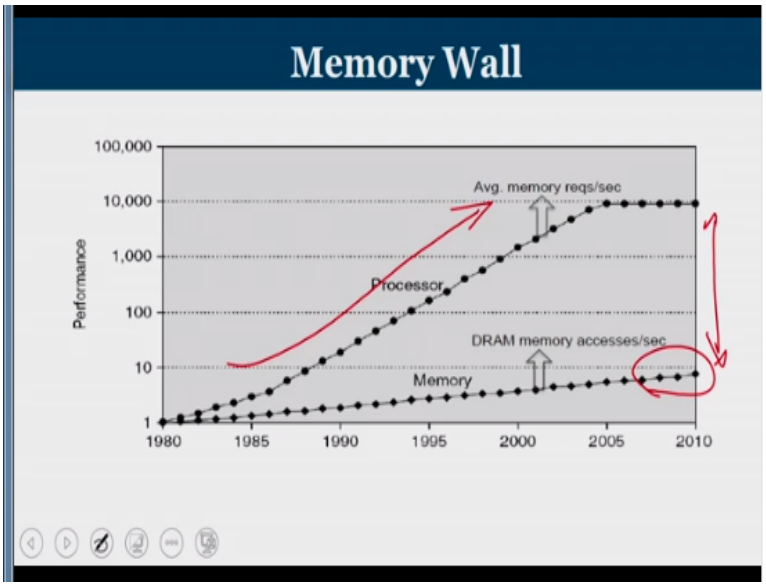
And if you go at the trend, our chips are not the power economical, so architects found out techniques by which they were able to reduce the power. This is an important observation that all modern day processors are coming up with green technologies or some kind of a mechanism wherein we could reduce the power density of a chip. So the traditional notion of having more intelligent functional units that will improve the performance has taken a hit. Because these functional units these intelligent units are going to consume power, so we have to put it down.

(Refer Slide Time: 06:29)



Now coming into ILP wall, here ILP stands for instruction level parallelism, over the years architects were trying to explore whether can we improve, Can we exploit instruction level parallelism and all these pipelined concepts pipelined processors, super scalar processors, hyper threading, multi threading, multi issue all these were tried and that also reached a saturation level somewhere in early 2000 that we cannot further exploit instruction level parallelism in a single processor.

(Refer Slide Time: 07:01)



And the last one is called memory wall even though we are coming up with better processing technologies in terms of more faster powerful high throughput processors, which was showing that processors having better performance over the years, the underlying supporting memory

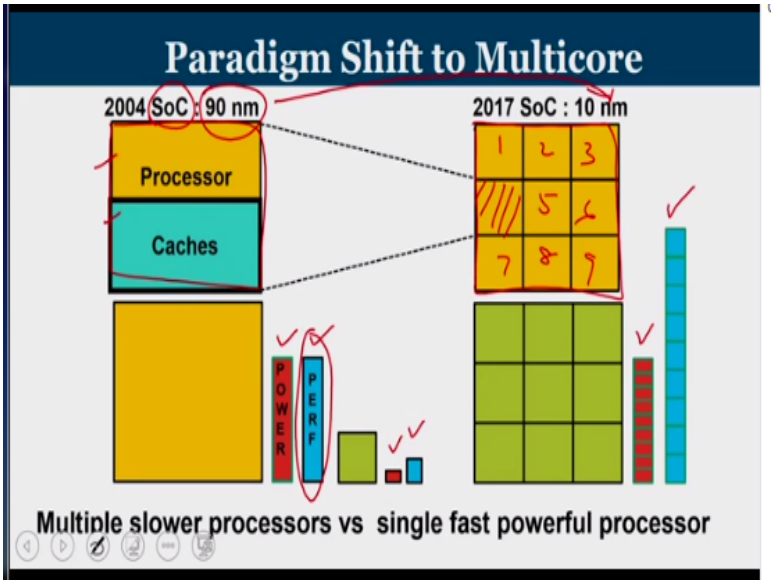
technology even though it is growing it is not growing at the same pace. So there is a huge gap between the performance of processor and the performance of main memory access.

So over the years if you try to summarize always we wanted more powerful microprocessors and that was achieved with the help of coming up with processors with higher clock frequency coming up with intelligent units, exploiting instruction level parallelism and facilitating more parallel pipelines and super scalar processors and using technology wherein memory can support the fast processor by delivering instructions and data at a very high rate.

But we were not able to increase the frequency beyond a point, we were not able to incorporate intelligent units beyond a point, we were not able to exploit instruction level parallelism beyond a point, we were not able to come up with memory technologies that can match up with speed of the processor, resulting in, a frequency wall, a power wall and ILP wall and a memory wall. So what is a solution, the newer applications demand more powerful processors.

And the traditional way by which we make processors powerful is no longer effective in terms of frequency, in terms of intelligent units, in terms of exploiting ILP and in terms of coming up with memory. So we have to find out an alternate solution let us discuss what are the tries in which architects tried to mitigate this problem of performance limitations of unicore processors.

(Refer Slide Time: 08:57)



There was a paradigm shift from unicore to multicore, every processor or every chip is been defined by something called process technology. It is roughly the size of a transistor or to be very precise it is the channel length of a transistor. Let us assume in 2004 a chip consists of processor and caches together this is called your IC. Let us say it is built with the 90 nanometer technology that means 2 transistors can be as close as roughly about a 90 nanometer.

Now by the time in 2017 whatever functionality this 2004 SoC had we could squeeze the same functionality and logic into this much space, that means in the same available space we can have 9 such processing unit we can see a shrinking in the process technology also from 90 nanometer technology, we are moving on to 10 nanometer technology, why does a take away from it since a transistor size are shrinking, the size of the functional unit also is going to shrink.

So in the available space we could squeeze more transistors, more functional logic or in a say we can have 9 processing units in this given space. Let us imagine let us say this is the physical size of a processor let us say this is the power dissipation and this is the performance that you are going to gain if this much space is been filled with transistors and this all this transistors as part of your processor and cache.

Now let me make a very simple small processor with 1 by ninth of the available transistors. Since I am using only 1 by ninth of the transistors, it may not be that much powerful a processor but still it is a processor with less capabilities, the power consumption is less the performance is also less. But the takeaway is you can have 9 such processing units that can be kept in the same space which is having the power equal to that of the previous one.

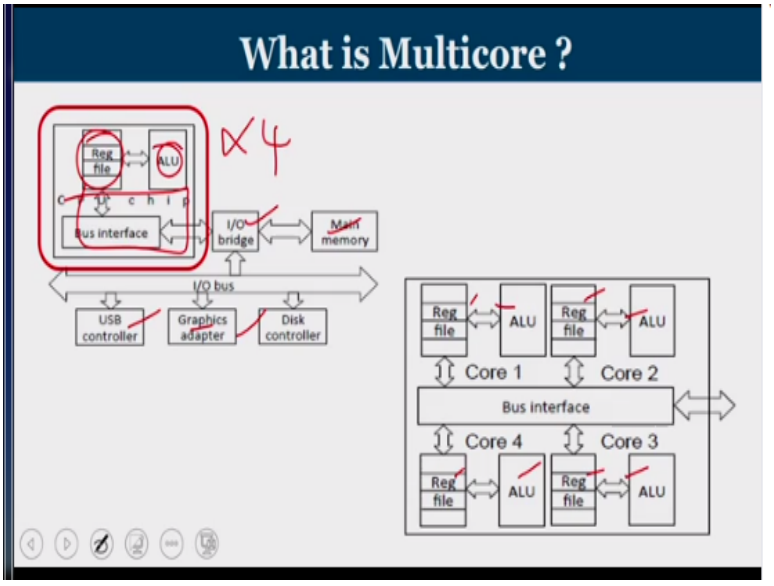
But the performance aspect when you look at it, it is much more than what was the performance of a unicore processor. So what is a takeaway, multiple slower processors are preferred over single fast powerful processor. This is the whole idea of moving from unicore processor that making one processor really capable very fast that is one side of the balance. The second one is can we have more simple processors, so that it can do more amount of work.

And if you look at the capability of each individual processor on the IC it may be much inferior to the super processor that we considered first. But the sum total of all these simple processors if they are put in into single IC, the throughput is going to be much higher than that of a unicore processor. So this actually gave a strong base to think in multi core processing domain and that is a reason by which nowadays we are all working with multi core processors.

(Refer Slide Time: 12:14)

So the paradigm shift in multi core is basically like rather than considering a single unicore processor it is always better to consider multiple simple processors which can easily beat a unicore processor, multiple slower processor is better than single fast powerful processor.

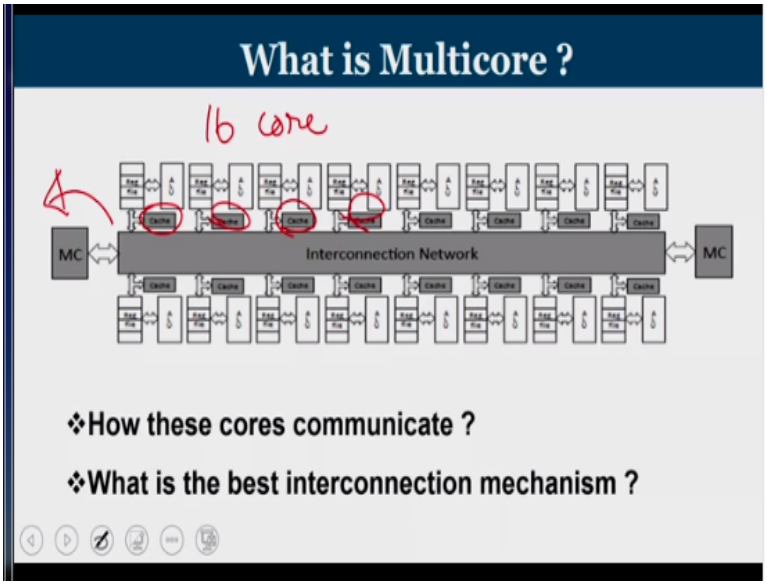
(Refer Slide Time: 12:31)



Now what do you basically mean by multi core, what do you see in this red square is basically the core of a processor, the core consists of the functional units like ALU plus the registers and the associated control logic that takes care of it. And all other things I/O bridge, the main memory, the disc, graphics controller all other things that are outside this processing core. Now if I make it into 4 times, let us say this is a 4 core machine.

So each core has it is own functional unit and register file and then it says control logic and they are all connected with the help of a BUS interface, what is multi core again.

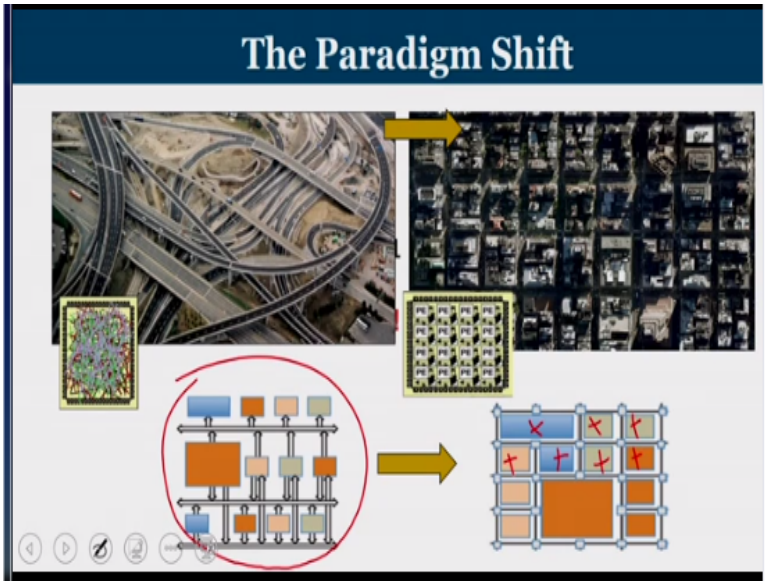
(Refer Slide Time: 13:11)



If you still scale it up this is a 16 core machine, where each core is going to have it is own cache, so the fetching and decoding happens with it is own independent instruction pipeline. So every core has it is own instruction pipeline, we have a set of cache memories each of the processing core has it is own private cache. So the fetching happens from there and if it is not available, we have to go to the next level of memory.

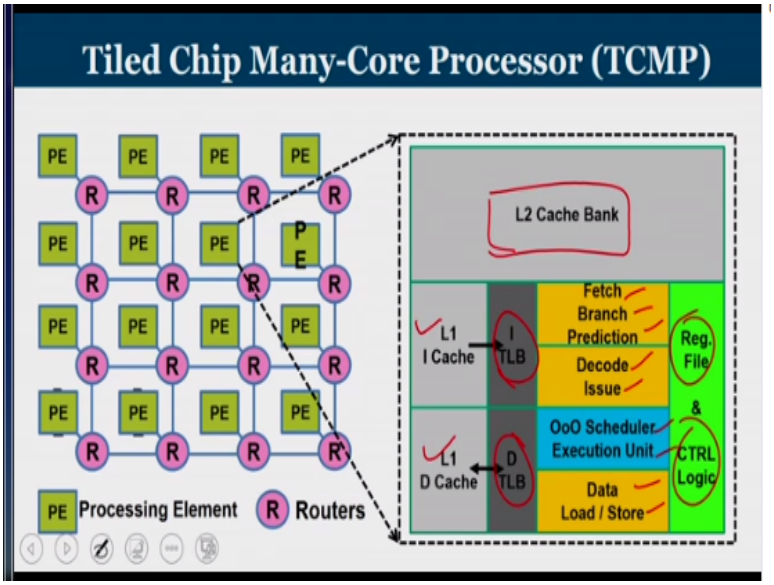
Let us say if it is main memory then you have this memory controllers that will help us to connect into the system. But how do this cores communicate our traditional BUS based mechanism won't work, so we need to find out alternate interconnection mechanism in the case of these multi core processors.

(Refer Slide Time: 13:53)



So what is basically a paradigm shift rather than a BUS based interconnection mechanism which is shown in this diagram, we are moving into tiles, where each of the functional unit let us say it is a storage functional unit or a processing functional units. They are all organized as tiles and the tiles are interconnected by short wires, which are in turn connected by control boxes called routers.

(Refer Slide Time: 14:20)

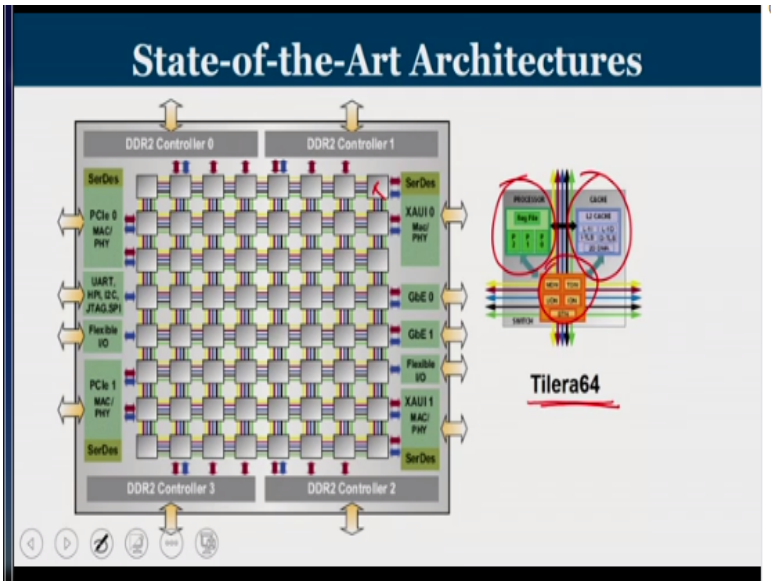


So moving on to the concept of tiled chip many core processors or TCMP, this is the logical layout of such a kind of a processor where is P stands for processing element and each of this P what are the things inside each of this P this is what we have learned over the last 19 lectures,

instruction fetch, then brands prediction, decode, then issue, then out of order scheduling, the execution unit, load store unit, register file, control logic which in mitigate all this.

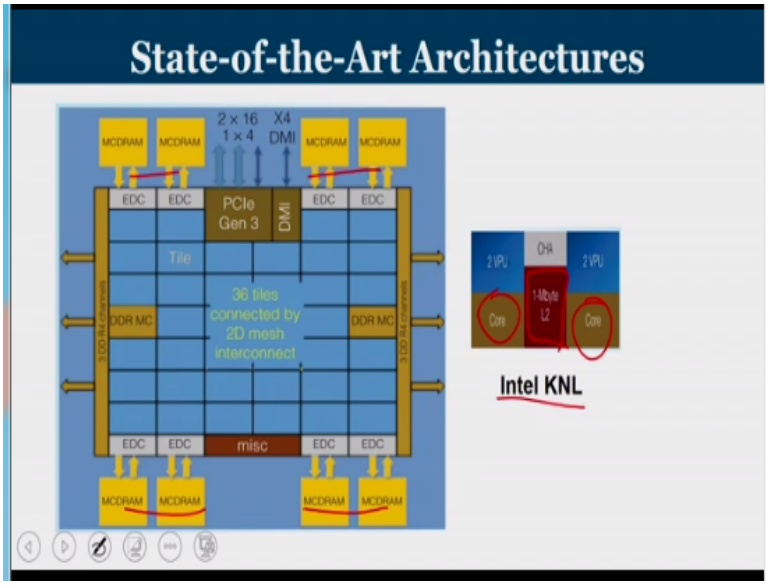
This instruction TLB and data TLB which converts from virtual address to physical address, I cache and D cache and a chunk of L2 cache this much is there inside each processing element. This is a TCMP where we have 16 such processing elements, so it is basically 16 core microprocessor inside a same tip. Now you can see this processing elements are connected through routers, routers are special intelligent units which will take care of data forwarding from one processing element to another.

(Refer Slide Time: 15:27)



Now there are certain processors which are already there for high end operations one such example is Tilera 64 where you have it is a 64 tile TCMP structure with we have 4 DRAM controllers that is been given. And each of this tile has the register file and processor + L1 L2 caches and an interconnect mechanism.

(Refer Slide Time: 15:54)

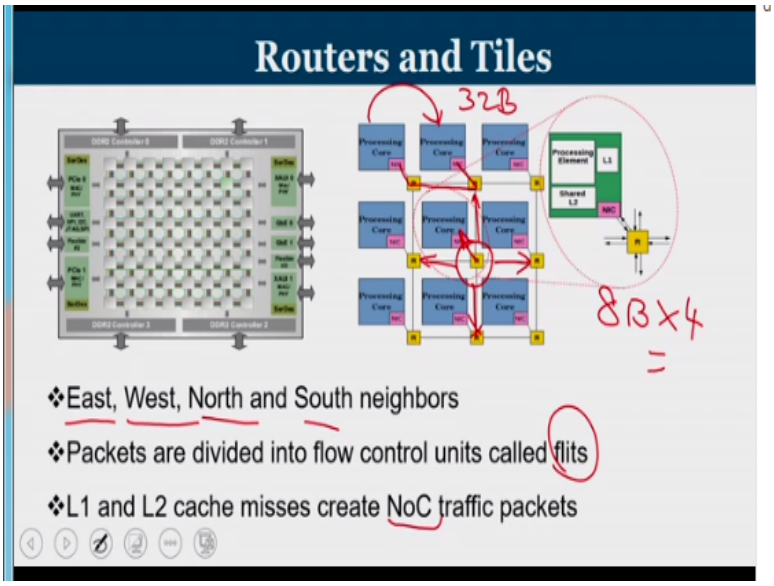


Yet another popular state of the art is the Intel KNL microprocessor Xeon phi microprocessor which is organized as 36 tiles. And each of this tile has 2 cores going to work on them with some private L1 cache and some sort of L2 cache as well. And it is having 8 DRAM controllers, so the DRAM controllers are been connected like this. These are the state of art processors like the Tilera 64, the tiled 64, and the Intel's one, Intel KNL these are all examples for modern tiled chip multi core processors.

So there was a huge paradigm shift the way conventional processors work and modern micro processors work. So what ever techniques that we have learned over this course they all are embedded into it, the only difference is there are multiple such units. Each has it is own instruction pipeline, it is own first level and second level caches, it is own schedulers and it is own advanced pipeline processing features like multi issue, super scalar, hyper threaded processors.

So it is going to be a combination of high end technology that we carry in our multi core processors, especially in our cell phones in our laptops. So in the coming years almost all our handheld devices are going to operate on this hundreds of processors which are part of the same chip.

(Refer Slide Time: 17:23)

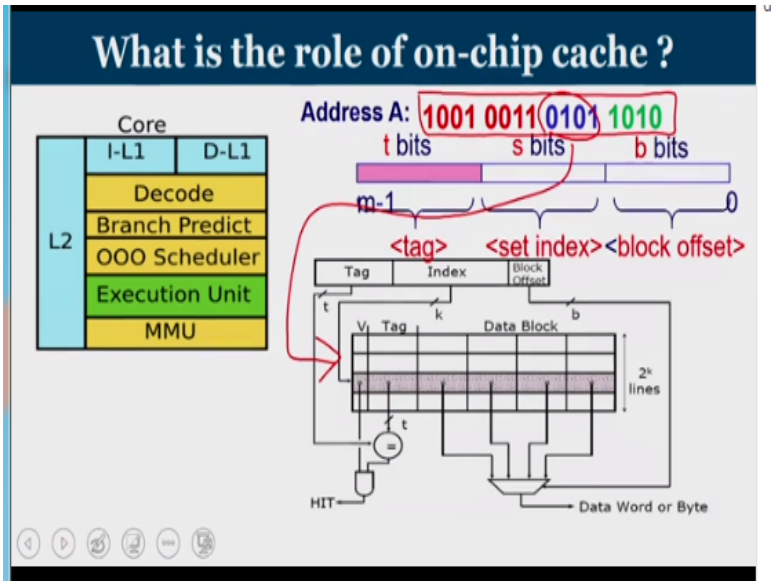


Now looking into the interconnect portion, we can see that these processing cores are connected by routers and these routers are connected with the help of short links and each router has east, west, north and south neighbors. If you take up this router this is the east neighbor, north neighbor, west neighbor and south neighbor and each router is connected to a local processing core also. And then we have packets that is the mode of data transfer from one processing core to another.

So whatever is a data that one core wanted to send it to another it is a packet but I cannot send the whole packet. For example let us say may packet is the 32 byte I cannot send whole data together it depends upon the bandwidth that connects between them. If the bandwidth is going to be 8 bytes or you have only 64 bits that connect then to achieve this 32 bit transfer I need to send it to us 4 different smaller units.

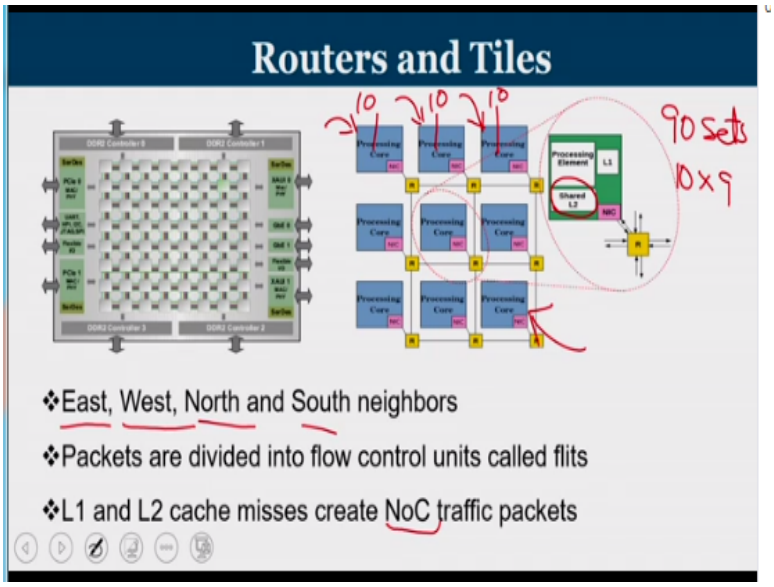
Such smaller units a basic unit of law control between a pair of routers is flit and then whenever you have L1, L2 cache meshes that is going to create traffic in this structure and this is called as network on-chip, we will learn more about NoC in the subsequent slides.

(Refer Slide Time: 18:30)



Now what is the role of on-chip cache, what we have to understand is generally we employ a shared distributed L2 cache in the case of tiled chip multi core processors. Now let us try to understand what do you mean by the shared distributed tiled chip multi core processors.

(Refer Slide Time: 18:58)

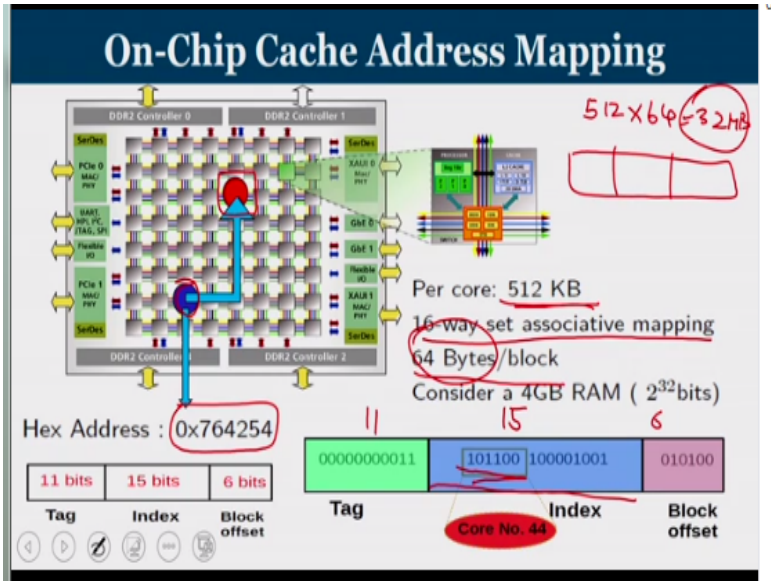


So imagine in this case, you have 9 cores and then it is a shared L2 in each of this course, imagine that the total L2 to cache. For example consists of 90 sets generally it would not come to this number I am just taking an example. Now if this 90 sets are divided into 10 set each spread across 9 core, so each core has 10 sets in it. So set 0 to set 9 will be in this core set 10 to set 19 is in this core, set 20 to 29 is in this core.

Similarly set 81, so 80 to 89 is in the last core, so whatever is my L2 cache that is distributed across all the cores and they are shared it is not like a private property of each of the core where it is residing. Even though it is located there, it is actually a shared L2 cache. Now generally processor will give an address which is divided into tag index and offset that is what we know. Here in this case the red portion is the tag, the blue portion indicates a set index and the green portion indicates the offset.

Now using the set index you go into what is a set in the cache memory and then you perform this tag comparison and if it is a hit then you are going to extract the data.

(Refer Slide Time: 20:17)



Now in our case we have our L2 cache which is shared and distributed, so imagine a case let us say in each of these core I have 512 KB of L2 cache let it be 16 ways set associative mapping and you have 64 bytes per block. And you imagine that you have a 4 GB RAM. So if you have a 4 GB RAM and you have total of 512 KB of cache into 64 core, so that is 32 MB of L2 cache is there this 32 Mb is scattered across 64 cores.

Now if you divide the tag index and offset portion considering it is a 32 MB L2 cache then you will find this split up as an 11 bit tag, 15 bit index and 6 bit offset, why 6 bit offset because you have 64 byte block. So this 2 power 15 sets what we are talking here this is scattered across all

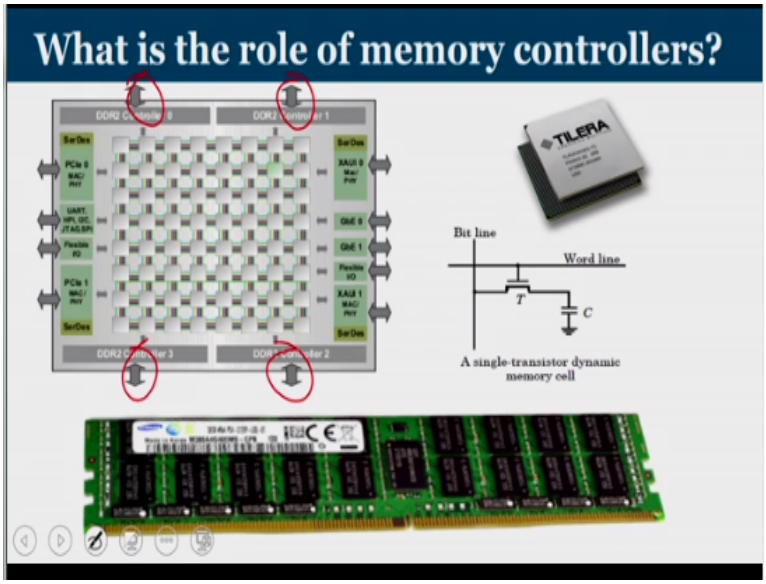
the 64 cores and this can be further divided into, so consider the case that you are going to work with an address whose value is hexadecimal 0x764254.

And if I write this 32 bit value 0x764254 which is written in hexadecimal convert to binary take the 11 bit tag 15 bit set index and 6 bit offset. Now if you look at the more significant 6 bits of the index this is your 15 bit index that is a set number. But where the set number is located look us the first 15 bit they will tell you what is a core number.

So this correspond to decimal value 44 that means, if there is a miss let us say it is happened from core number, this blue core, what is been generating the miss. Then the place in which this address is found is in core number 44. So from the blue core you have to generate a message that should go all the way up to the red core that is core number 44 bring the data and come back. So the actually this is the path in which the data has to flow.

And this movement is facilitated by the on-chip interconnection mechanism that is a network on-chip interconnection mechanism. So generally when there is a cache miss and using this process we will find out where is a location in which this missed address is and packets have to be generated through that particular destination core and this packets will travel through the **under** on-chip interconnection mechanism.

(Refer Slide Time: 23:02)



And what is the role of memory controllers, we have learned about memory controllers and it is memory controllers connects to your DRAM. So your DRAM cells, so DRAM banks are kept with it memory controllers. And depending on address mapping memory controller 0 will take care of few things. So it is basically 4 channels that you have memory controller 1 will take care of the further rest then 2 and 3 for the appropriate addresses.

So, I was trying to give an overview about how generally a multi core processor is all about what is the role of a cache on-chip or role of the memory controllers and role of the interconnection system. In the next lecture we will learn about deeper into what is the interconnection mechanism which is called network on-chip. So there is a paradigm shift that we learned all the concepts are been integrated into a single processor IC and that is what is known as TCMP tiled chip multi core processors thank you.