Multi-core Computer Architecture - Storage and Interconnects

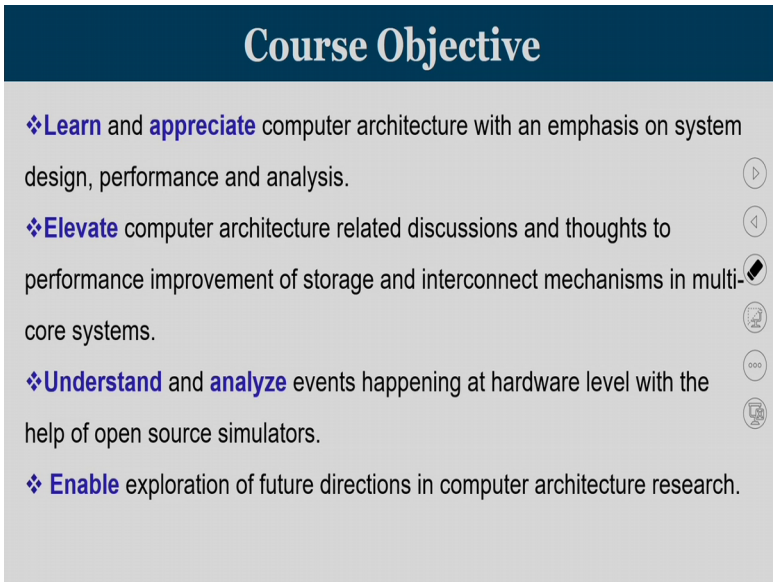
Dr. John Jose
Department of Computer Science and Engineering
Indian Institute of Technology, Guwahati

Lecture - 01
Introduction and overview of the course

Welcome to the Introduction and the course overview of this course, Multi core Computer Architecture Storage and Interconnects. This particular session we will be basically focusing on what will be the coverage of the course. And we will discuss about the syllabus a bit in depth. And what is the week wise course plan it is going to be covered. So, that this will get candidates and approximate idea about what will be the material that is going to be covered.

So moving into these course details.

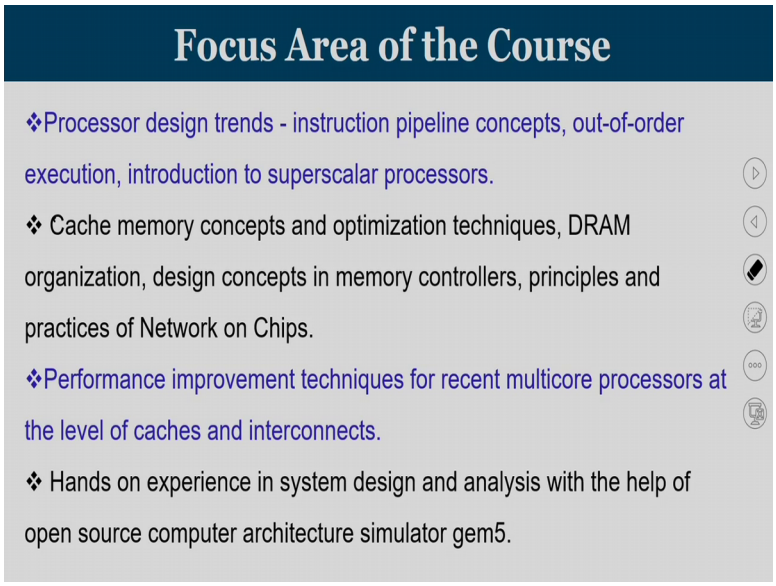
(Refer Slide Time: 01:06)



The objective of this course is: to learn and appreciate computer architecture with an emphasis on system design, performance and analysis. Moving to the second objective, to elevate computer architecture related discussions and thoughts to performance improvement of storage and interconnect mechanisms in multi core systems.

Understand and analyze events happening at hardware level with the help of open source simulators.

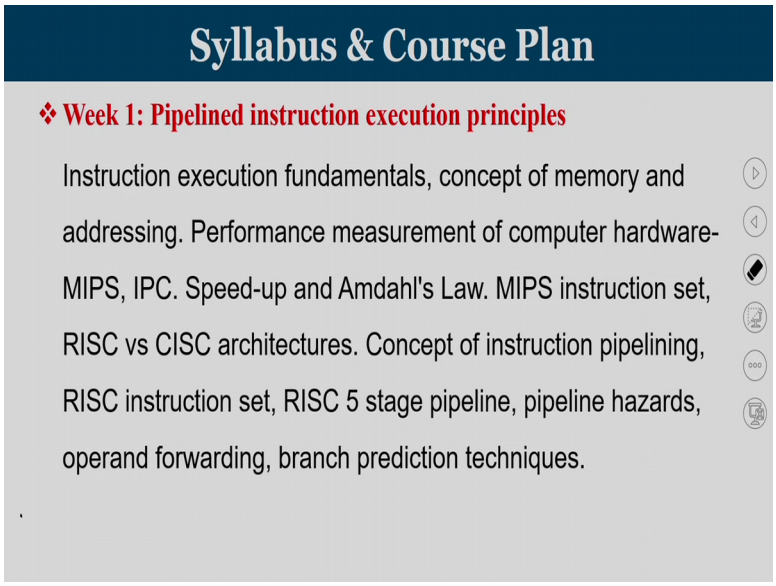
(Refer Slide Time: 01:50)



And at the end to enable a couple of you in exploration of future directions in computer architecture research processer design trends, we will be focusing on instruction pipeline concepts, out of order execution, and introduction to superscalar processors. And then we focus on cache memory, concepts and optimization techniques, and then DRAM organization, design concepts in memory controllers. And then we focus on principles and practices of the interconnection structures which is known as network on chips.

Performance improvement techniques for recent multicore processors at the levels of caches and interconnects. And last a couple of hands on experience sessions in system design and analysis with the help of open source computer architecture simulator gem 5.

(Refer Slide Time: 02:38)



Now, coming to a week wise planned I think this will help you in understanding what will be the material that is covered on a weekly basis.

Like already mentioned, our course will span across 8 weeks. So, I will just give a brief overview about what we will cover in these 8 weeks. Week 1; our focus will be on pipelined instruction execution principles. So, all the background that is needed from the processer side of the architecture will be covered in the first week itself. And then from second week onwards, we will be looking more into the storage aspect and interconnect aspect.

In week one: we will be having a brief coverage about instruction execution fundamentals, then concepts of memory and addressing. And when you work with any computer system performance measurement of such computer systems is also a crucial design parameter. So, performance measurement of computer hardware with the help of MIPS with the help of IPC will also be covered.

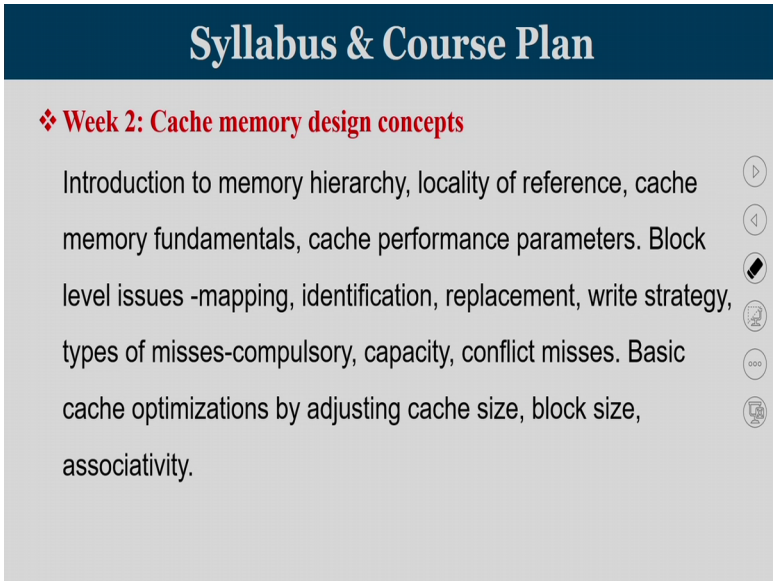
When you design new architectures, we wanted to understand how much performance that the new system will have over the traditional system. How are you going to measure it? There are a couple of laws which help us in computing how much performance improvement we get. Once such is the Amdahl's law so speed up with Amdahl's law will cover. And MIPS instruction set, a lighter treatment of that will help you in

understanding about how instruction pipeline works. A comparative study of RISC versus CISC architectures will be there. And then we will focus on instruction pipeline.

We all know that every task is represented as a collection of instruction that is stored in memory. And we are going to fetch these instructions from memory into the processor, decode them and then execute them. Rather than fetching this instruction all one after another can we do some kind of a parallelism on it. And such a kind of a principle is called instruction pipeline we will cover to it in the subsequent lectures.

So, concept of instruction pipeline RISC instruction set RISC 5 stage pipeline. And there will be certain cases where you cannot run instruction in a pipelined manner which we call it as hazards we try to understand what are the basic hazards that we have basically we call it as pipeline hazards. And there are some techniques that will help us in overcoming thees hazards. And they will be discussed and some of the techniques like operand forwarding branch prediction techniques are there.

(Refer Slide Time: 05:36)



Moving on to week 2 is basically on cache memory design concepts. So, like already mentioned, we need to have a good memory system also along with a high end processor in order to facilitate good performance improvement. So these memory, these high end memory systems are known as cache memories. So, we will try to understand how cache memory works or; what are the basic design principles of these cache memories.

So, introduction to memory hierarchy, we have different levels of memory we will talk about that that is from registered to cache memory to main memory to disc, then the working principle of cache memory, basically locality of reference. The cache memory fundamentals, cache performance parameters, and cache memory consist of multiple blocks. And we will see how block level issues are there are few mapping techniques, how will you identify whether a given or a requested word is available inside cache memory that is basically called cache identification.

And once the cache memory is full when you have further request that is coming, and what are the replacement policies that is being used. And when you write something into cache memory, we need to know how writing is performed the that can be a hit in cache memory the address location can be a hit in cache memory or it can be a miss in cache memory.

So, there is a right strategy that governs us in finding out what are the actions to be done when it been encountered with a right miss. And something when it is not present in cache memory, and that is what is called a miss. There are different categories of miss we have to understand that and based upon the various categories of miss. We try to focus or we take a call on what are the actions to be done.

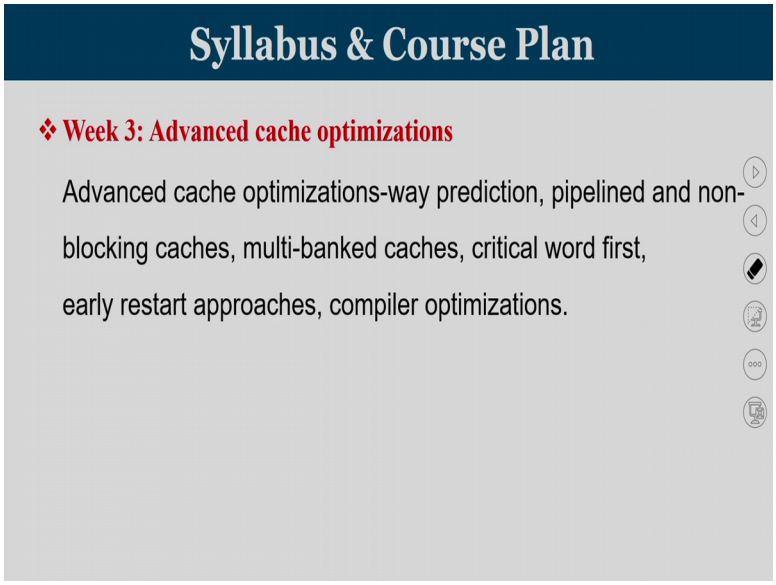
So, there are 3 types of misses what is called the compulsory miss, and then we have the capacity miss, and then we have the conflict miss. So, a treatment on what are the characteristics of all these miss how can we overcome this misses will be covered. And then we have to focus on optimizations in cache memory. So, when you talk about optimizations. The idea is can you improve the performance of cache memory under the given constraints.

So, we are looking for a parameter called average memory access time. The amount of time that we spend on the memory hierarchy, sometimes it can be a heat from the cache sometimes it may be a missing cache which will go to main memory, sometimes it may not be even present in main memory we have to fetch the corresponding instruction and data from the disc.

So, if you wanted to reduce average memory access time. There are certain things there certain adjustments what we have to make on the cache memory hierarchy. So, that is

what is called cache optimizations. So, basic cache optimization focused on adjusting your cache size, then block size and a parameter a design parameter called associativity.

(Refer Slide Time: 08:40)

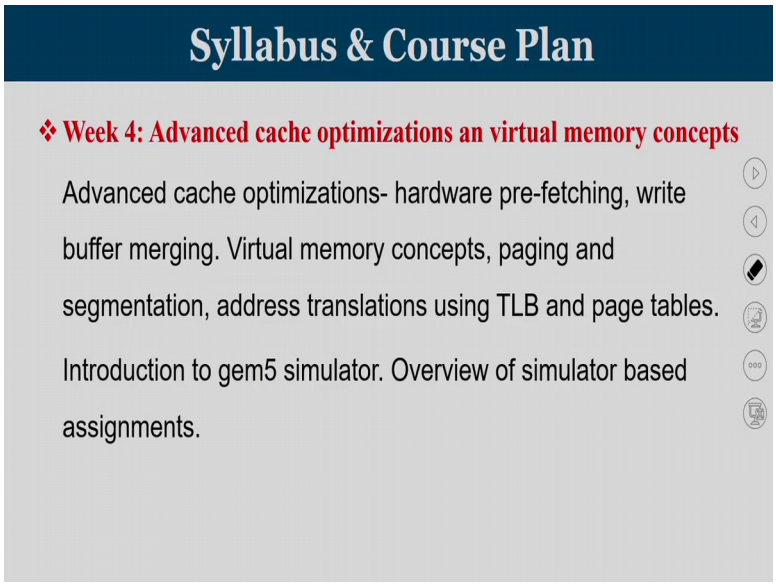


Moving on to week 3 we will focus on slightly advanced cache optimization these are all the research outcomes in the last 3 decade, which help our cache memories to perform much better than the primitive caches. We will be exploring on certain techniques, such as way prediction techniques pipelined and no blocking caches. And there are certain category of caches called multi banked cache which will improve your bandwidth so, we discuss on multi banked cache.

And then certain specific approach like can you bring the critical word first can you restart caches early than normal. And in since computer architecture goes hand in hand with the compiler. Some sort of help from compiler also help us to improve one hardware.

So, compiler and hardware has to go hand in hand compiler has to share some kind of an information which will help your hardware to do some kind of an optimization, they are basically called as compiler optimizations. So, that will conclude our week 3.

(Refer Slide Time: 09:45)



Coming on to week 4 we will be focusing on little bit concepts of virtual memory and the corresponding cache optimizations.

So, we will continue with advanced cache optimization with something called prefetching. Prefetching is an idea where we bring something before it is needed. So, if I know that at some point of time later in the execution, I want a memory address m can I bring m to cache memory slightly early that is called prefetching techniques we will explore on that. And then write buffer merging when processor write something on it is to first level of cache memory. If you wanted to make your cache memory bit of coherent, then it has to be updated in your L 2 cache or in your main memory.

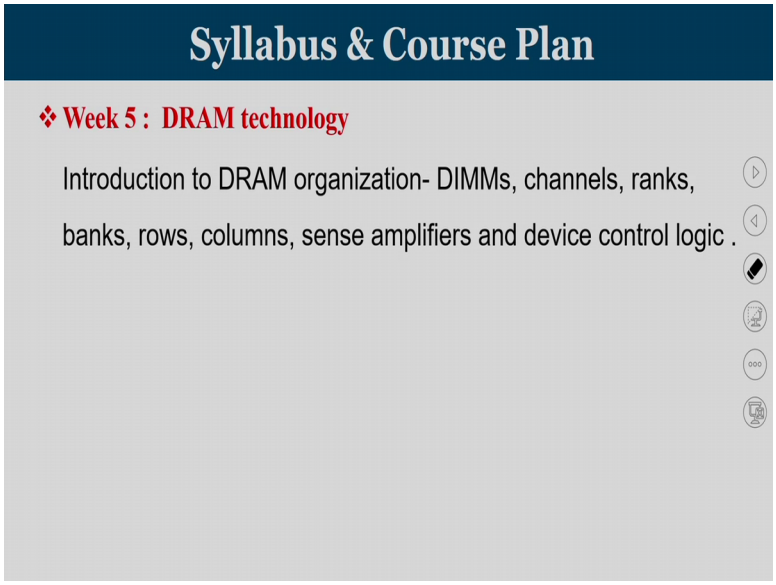
Write buffers are an architectural units, that facilitates updation of your next levels of memory incoherent vista your first level cache. And write buffer merging is optimization which will improve your miss penalty. And then we will focus on virtual memory concepts with paging and segmentation. And if you wanted to work with virtual memory, the virtual address is given by CPU has to be translated to physical address. And there are certain architectural units which will help in transferring of a virtual of a given virtual address to a physical address and the page tables, and TLBs will basically help in this task.

And this will cover up the first half of our codes. And in the second half, we will be focusing on main memory systems, and then the interconnection system. So, our week 5

will be focusing on main memory systems. Now in order to help a better learning experience, we will be working with architectural so simulators there will be two assignments that will be given to you which will give you a grip on understanding what happens what are the events that are happening in the hardware when the program is been executed. For that we have chosen the most popular computer architecture simulator that is called gem 5

So, there will be one session which will talk about how can you configure your systems with this simulator, some initial toy experiments will help you in gaining some confidence in this. So, introduction to gem 5 simulator overview of the simulator based assignments also will be covered.

(Refer Slide Time: 12:17)



Coming on to week 5, we go to the next level of memory that is DRAM technology. Introduction to DRAM organizations, the DIMM architecture, and this entire DRAM structure is divided into various channels ranks banks rows and columns.

And then we have sense amplifiers which will help you in understanding whether a logical one is stored there or a logical 0 is restored there. So, sense amplifiers will work on and what is the basic device control logic.

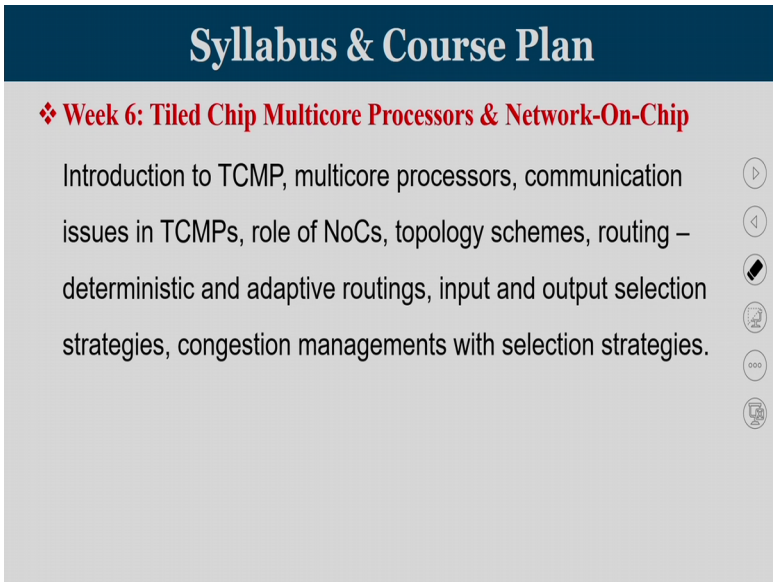
Now, apart from the basic storage hierarchy in DRAM, we have an intelligent unit, that is called the memory controller that facilitates reading and writing from the DRAM. So,

we focus on DRAM timing and signaling access protocols, and what are the basic commands that is been used, and how will you address the bank and rank conflict. We all know that DRAM is built with capacitors. So, capacitor has a leakage property that will lead to DK of the data that is stored.

So, we make use of a refreshing circuitry, and this refreshing circuitry will make sure that whatever is the decade charge that is being stored back. So, we will spend little bit of time in understanding; what are the properties of the DRAM refresh circuitry, and what are the power management schemes that are available in these DRAMs. So, better power management scheme, then our device is the handheld devices or mobile phones can stay longer before recharging.

And then since in a multicore setup, you have multiple processers and multiple applications will be running on these processors. And these applications can encounter with cache misses which ultimately will end up in your DRAM system. When you get multiple request on a DRAM how will a DRAM take a call whether should I satisfy request r 1 first or should I give preference to request r 2 over r 1. And that is been taken care of by the DRAM scheduling policies.

(Refer Slide Time: 14:32)



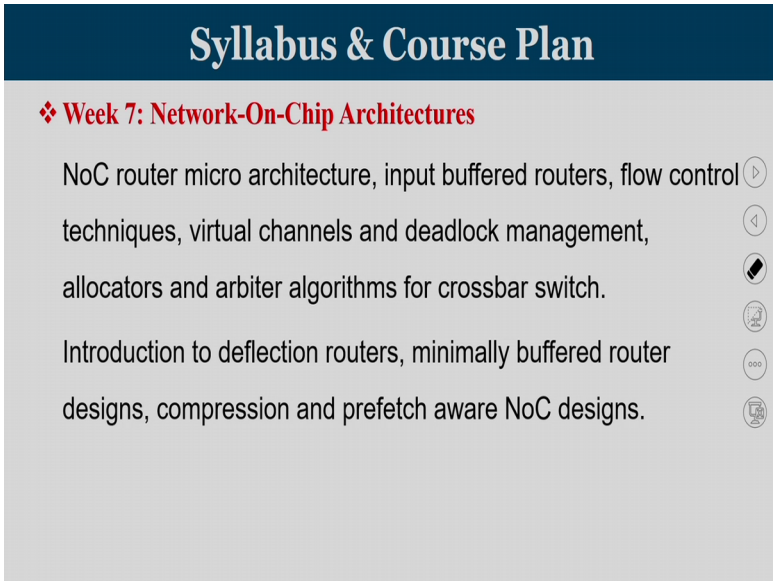
Now, on week 6 we will introduce you to the concept of tiled chip multicore processors. That is the recent multicore designs where all processors are organized the tiles inside the chip. So, a brief introduction will be given regarding what do you mean by tile the chip

multicore processors, it is also abbreviated as TCMPs and the communication fabric in such tiled chip multicore processors is called network on chip. So, introduction to TCMP multicore processors and what are the communication issues that we have in TCMPs which forced us to switch from the traditional bus based communication mechanisms into the network on chip based mechanism. And then we will tell; what is the role of network on chip which is abbreviated as NoCs.

What are the various topologies that is being used, and since we are going to lay a network inside a chip that is responsible for transferring of data from one unit to another. We need to understand some kind of networking principles as well. And one of the important concepts in any networking is rooting of your data or a packet from one place to another. And that is being covered in called the routing techniques. And then we focus on input and output selection strategies.

So, when you have multiple routing options available. It is these strategies that help you in picking up the right packet to be forwarded. And these techniques the input and output channel selection strategies will help you in reducing the congestion in the network.

(Refer Slide Time: 16:15)



So, in week 7 our focus is on the network on chip architecture. So, micro architecture of this NoC designs. So, NoC router microarchitecture will be discussed, followed by what is the structure of input buffered routers. And since you have multiple routers that is available inside your chip, how the data is communicated from one router to another.

What guarantee you have that your data that is being sent from one unit is reaching the other unit. This is ensured with the help of flow control mechanisms.

So, we will touch upon flow control techniques. It is more or less similar to the traditional flow control that is adopted in macro networks, but this is only hardware there is no software layer on it so, that aspect we will work on, and every routing this prone to deadlocks. So, the common mechanism that is been used in network on chip is to use virtual channels, and this virtual channels helps in deadlock management. And then we have allocators and arbiter algorithms, that work on this crossbar switch which will facilitate priority picking of packets and subsequent forwarding into the adjacent links.

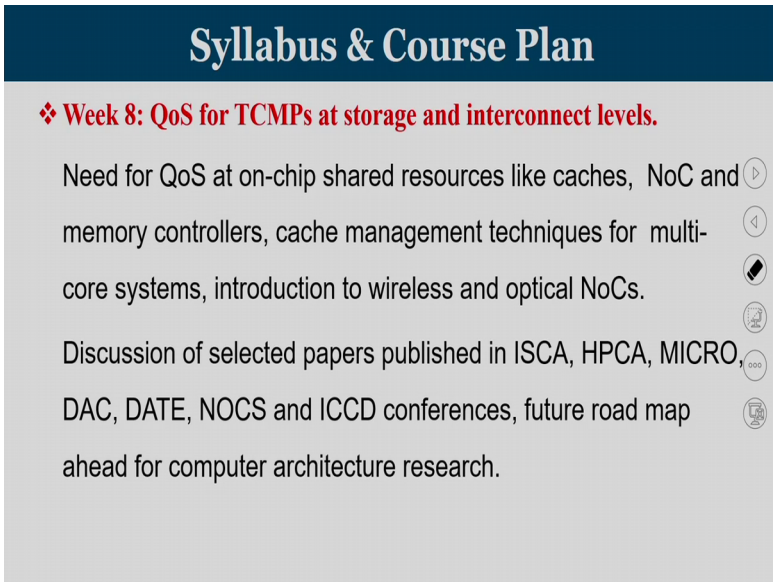
Now, these network on chip routers are going to be having some kind of buffers. And it is the handshaking mechanism from one router to another. That facilitate the smooth flow of packets, but this buffers are power consume power consuming unit. So, do we have any mechanism that is available such that we can get rid of this buffers, yes we do have. There are techniques called buffer less router designs which will focus on deflection routing.

So, we spend almost one lecture on deflection routers, and minimally buffered deflection router designs. Having said this since there is tremendous amount of data that flows from the cache memories into these processers, or from one core into another. Can we see or can we explore the possibility of can I compress this data, or can I bring this data before it is required. So, these 2 techniques one is called compression and other one is called prefetching that is going to improve the performance of future multi core processors drastically. The data especially majority of the data that flows in common handheld mobile phone is either a video data, or it will be something like images. And we have lot of data pattern that is available in this videos and images which can be compressed.

So, can you have an NoC design or a network design, which will provide you some possibilities for compressing it first and then send the data at the receiving side the combustion send. So, we will spend little bit of time on compression and prefetching aware NoC designs. And on the last week we are going to work on what is the quality of service, what is the quality of experience that a user will get on a tiled chip multicore processors at this storage and interconnect level.

So, what is the need for quality of service?

(Refer Slide Time: 19:42)



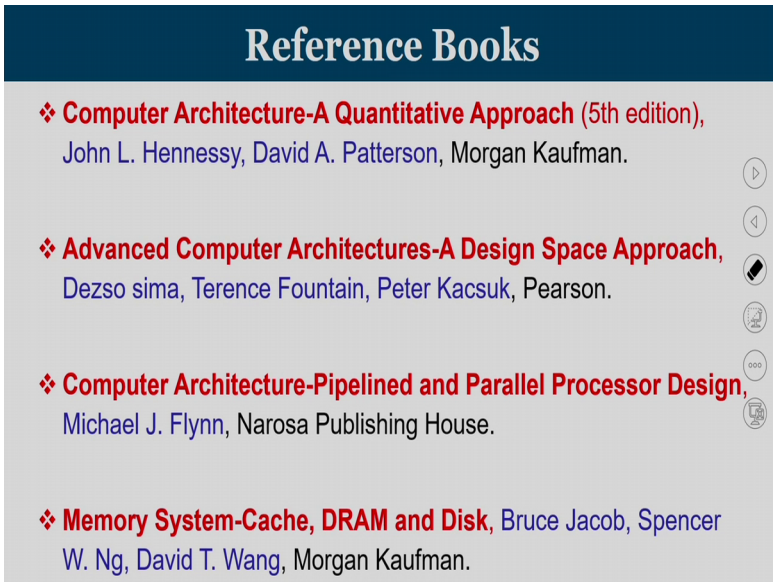
Especially when you have on chip resources on chip shared resources like cache memory is setting on the tile, you have your NoC setting on the tile, and you have your memory controller setting on there. So, all these are shared resource and all of the wanted to get the service fast and how can you manage. This how can you ensure quality of service?

Cache management techniques for multi core systems also will be explored. And before concluding a very recently emerging topic which is called optical network on chip, and then we have the wireless network on chip will be covered there. And maybe for those of you who wanted to explore in this field further those who are looking to do a some kind of projects or looking to work on post graduate thesis or doctoral thesis in this subject. I will just introduce you to a couple of good research papers that are published in top premiere conferences in the field of computer architecture. And that will help you in exploring further in this domain.

I hope with this kind of an overview, it will give a good amount of treatment and exposure to candidates who wanted to explore in this area. For beginners it will be a very interesting course. And for those who are already having some background in computer organization and architecture this will give them an in depth treatment and in depth knowledge in this areas which can help you in working on projects related in this same and those who are passionate about research very well you can work on the research topics related to this.

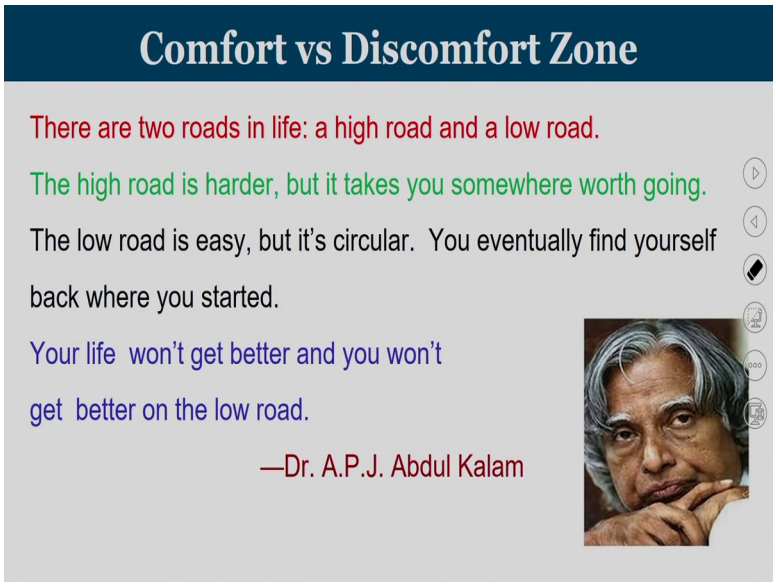
And these are the reference books.

(Refer Slide Time: 21:26)



One is the computer architecture quantitative approach by John L Hennessy and David Patterson. And this book is also helpful by Dezso Sima advanced computer architecture design space approach. And this book will help in parallel processing design computer architecture, pipeline and parallel processor design. And the next one is about memory systems by Bruce Jacob and his team.

(Refer Slide Time: 21:57)

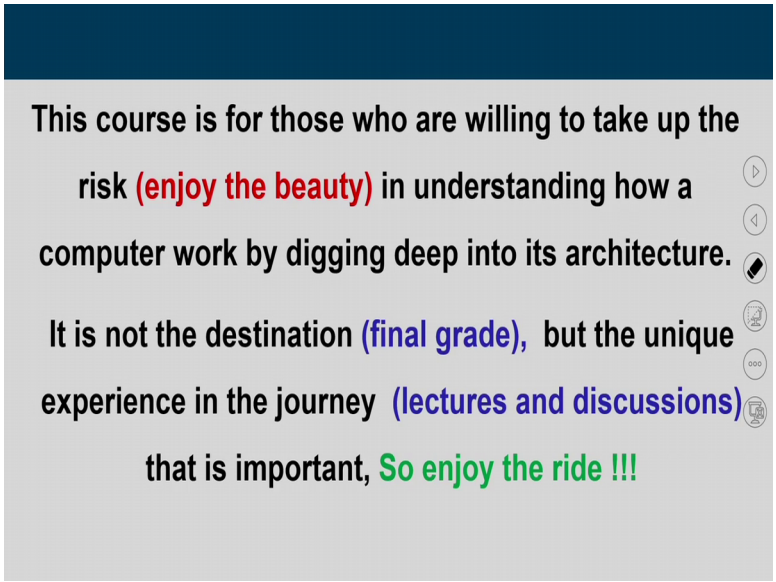


So, having said this something sometimes you may get a feel of this entire course is going to be very tough or it is going so much deeper, but to achieve something if you wanted to have to become a designer. Then knowing certain things at hardcore having an experienced with hands on experience on simulators working on design problems will give you much conceptual clarity that is required at an engineering level.

So, like I would like to put in the codes of our former president APJ Abdul Kalam. There are 2 roads in life a high road and they low road. High road is harder, but it takes you somewhere worth going. Low road is easy, but it is circular you eventually find yourself back where you started. Your life will not get better and you will not get better on the low road. So, I will provide you all the necessary help and support that is required in understanding the entire course to it is final granularity. So, enjoy the ride.

So, this course is for those who are willing to take up the risk or we can call it is enjoy the beauty in understanding how a computer work by digging deep into it is architecture.

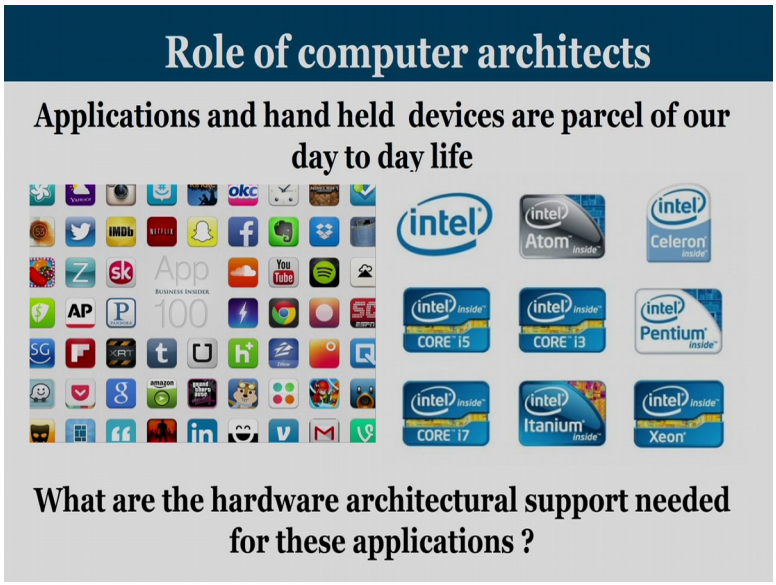
(Refer Slide Time: 22:57)



It is not the destination, that is a final grade that you are going to get in this course that is important, but the unique experience in the journey that is the lecture sessions and the discussions the more active you are there in the discussion forum, you will be able to clarify much of your doubts. And that will give you much much clarity that is required at this level. So, I request all of you to enjoy the ride.

Now, we will see; what is the role of computer architects in high end compute intensive applications that are very much important in common man's life. Applications and handheld devices are parcel of our day to day life part and parcel of our day to day life.

(Refer Slide Time: 23:49)



Like, these are some of the applications in which our we are daily interacting with. Our day starts probably on a smartphone by looking at the alarm, and then we will see; what are the updates in our social media. And then further on there are many important programs or applications that run on our mobile phones which are very useful to us. Some are scientific applications, some are applications that are related to social media, connected to your mails, connected to your various accounts.

So, in all these the underlying principle is basically a processer that is running. And on top of that we are having an operating system and various applications that that are run on top of this. So, how it is solve been done? We have microprocessors that is there in these devices, and this microprocessors are going to fetch decode and execute task. What are the hardware architectural support that is needed for this application? This course is basically going to explore to find out architectural requirements for running these kinds of high end compute intensive applications.

Now, we will see a couple of applications that are going to be very much important in our day to day life in the future next one decade.

(Refer Slide Time: 25:17)

One is smart homes; we are actually heading on to a smart home era. What do you mean by smart home? There are various devices that are there inside our home, many consumer appliance electronic like your AC, your washing machine, your heater, your TV, many things are there. We are going to connect all of them together and trying to control these things from remotely or with the help of a computing device.

This involves lot of sensors, and these sensors will be fitted to various sections in our homes, and they are going to exchange data, some of these data need to be processed on real time, and there are certain things in which you have to turn on or turn off certain devices, or control or manage certain devices. So, this is a very heavy; that means, data intensive application, there are lot of devices that are connected there can be a high end computer or a work station or a server inside your home that is going to process these data. And the number of gadgets or devices that are going to be connected on these server systems is going to drastically increase in the next few years.

So, to understand hardware of such kind of applications, we need to know what are the kind of data that is coming how fast we have to process these data, and what are the characteristics of this data. Yet another application that is going to come very common in the next future is going to be driverless vehicles.

(Refer Slide Time: 26:49)

Now, in driverless vehicles, we know the concept of driverless vehicles, we have lot of sensors that is been fitted in the vehicles that are going to capture lot of information like, how far or close is the next object the next vehicle, how far or close you are from the borders of the road. It may assess some of the parameters of the road like the friction of the road whether it is a raining or not.

Like that many parameters are being sensed and these data has to be processed in real time this is an example for a hard real time system. Any delay in processing and acting upon certain data can cause problem to the life and security of the people who are travelling in this. So, we are heading on to this driverless era.

(Refer Slide Time: 27:42)

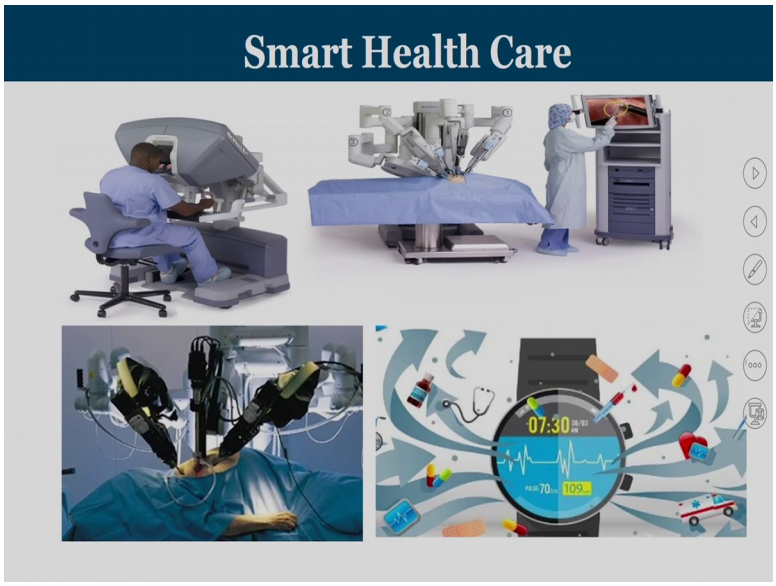
The next important area which is going to have a significant impact is all about video surveillance.

Now, a day security is a big threat to infrastructure. So, we have fitted upon this cameras, and this cameras are going to live record the videos of the event that is happening in and around an infrastructure. And nowadays we have huge server rooms where in there is a human personal who is going to monitor what are the things that is happening across various sensing points.

Once the number of cameras that are going to increase, this human intervention will become extremely difficult. Under that context we require content searching inside videos and find out whether there is any anomaly or not and if there is any anomaly then we have to initiate certain actions. Here also the concept is same we have a set of video collection points, this videos are transmitted across to the remote centers or connected centers where this videos are going to be processed. And based upon some anomaly in these videos appropriate actions has to be taken.

We look into one more application.

(Refer Slide Time: 28:56)



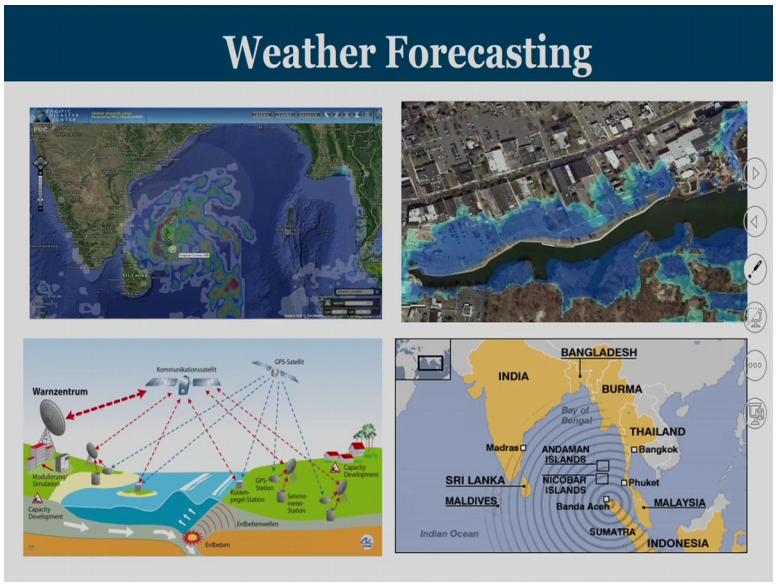
This is about smart healthcare systems. In this case, we can actually see this is a doctor who is going to do some surgery on a patient and this is the patient. They need not be there in the same place. For example, the doctor can sit in United States where as the patient will be there in UK. So, there are lots of sensors that are going to be fitted.

And this particular patients data is been recorded, monitored and it is been passed on to the doctor on real time. And doctor can view as if he is there in the same place as that of the patient even though the doctor is miles apart. And the doctor can work on these devices, and these devices will map into appropriate control signals which will enable this robots to work. In these cases also this is what is called a robotic surgery a doctor assisted robotic surgery.

Similarly, we can have lot of gadgets that will be in plant in our body. And these small sensors are going to check or monitor the level of various enzymes, like our blood sugar level and many other like minerals inside our blood. And then they can give appropriate actions to outside world, or can act as warning cases in the case some of the ensign level is going beyond a level.

So, all together these smart healthcare systems are going to play very important role. And what is the heart of this kind of systems? We are still going to have some kind of a sensor which is going to take up these values, and that are going to be passed it across where further intervention from humans or by a software that is needed.

(Refer Slide Time: 30:44)



And nowadays our planet is been extremely they affected by lot of natural calamities, like cyclones earthquakes tsunamis and all.

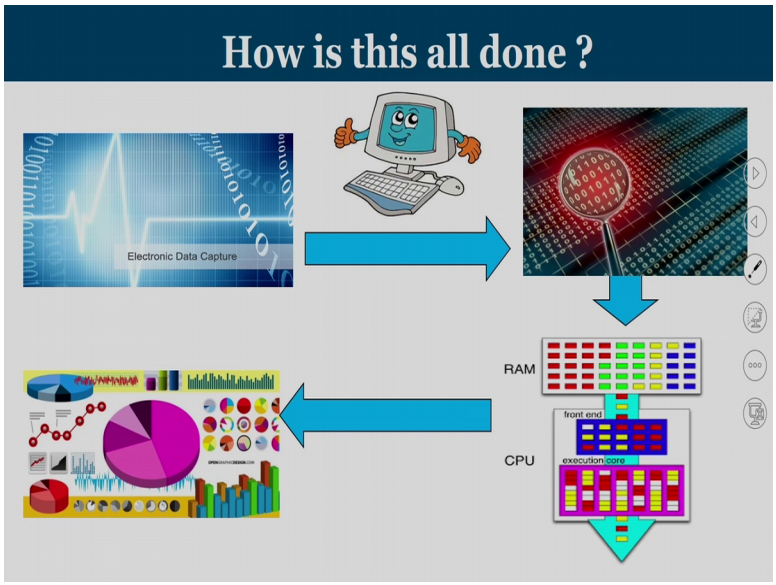
Now, we have a very important role to play here with help of advanced computing system. What are we trying to do here? It is about weather forecasting. We are going to absorb data; that is occurring during these natural calamities, and then we are going to put it in heavy end computing systems, and these computing systems are going to give us updates; by what time the cyclone will hit the land? What is going to be the trajectory of the cyclone? What is going to be the speed of cyclone? All these are heavy compute intensive application, and underlying hardware that facilitate these kind of services should have good architectural features such that we may be able to get results in real time.

So, we have learnt about different techniques or different kind of application domains which are going to have a major impact in future. So, first one is the smart homes then we have driverless vehicles.

(Refer Slide Time: 31:54)

Then we have smart healthcare systems, and then we have security and video surveillance systems then we have. So, all these essentially we are moving into a smart planet.

(Refer Slide Time: 32:05)



And how it is all done?

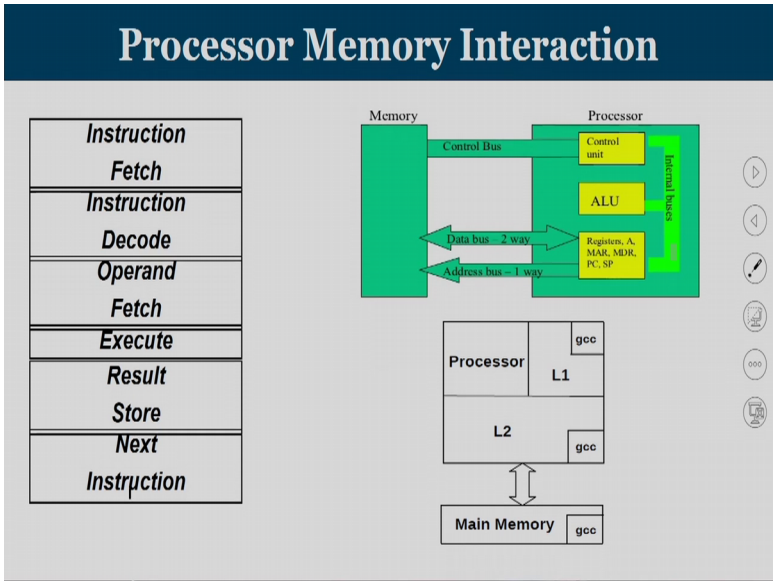
It is all about some kind of an electronic data capture. This captured data is going to be analyzed with the help of small and computers to see whether there is any anomaly or not whether it is going in normal fashion or not. And then we are going to put it in high

end computers with good storage and processing capacities, and we are going to get some conclusions.

In some cases this conclusions are given to human beings to act upon, whereas in some other cases these are going to be given to other devices or gadgets which will automate the whole process in controlling the system. So, this is the focus area of this course. We are going to see especially in modern multi core computer architect computer architectures; wherein we will see a specially the storage side of it and the interconnect side of it. So, the rest of the course will be focusing on. Digging deeper into the architecture of these kind of systems which are going to facilitate the kind of applications we have just discussed.

Now, to start from we know that a computer or a microprocessor is going to fetch and execute instructions.

(Refer Slide Time: 33:30)



Now, these executed instructions which are already stored in the memory are been transferred to the processor via buses. So, in short a task that a processor is going to do is represented as a sequence of instructions. And these instructions are stored inside your memory, these instructions are stored inside your memory.

Now, from the memory based upon an address sequencing mechanism, we are going to fetch these instructions, now we will see from the fundamentals of the processor memory

interaction. We know that microprocessors are designed to execute a task. Now the task is been represented as a sequence of instructions and are stored in memory; we are going to store it in this memory.

Now, these instructions which are already there in memory has to be fetched in sequence. Under a program sequencer to the processor we try to decode these instructions and then execute this task. This whole cycle can be represented as an instruction fetch, then we have an instruction decode, then we fetch the corresponding operands execute the task, and then we store the result. Then depending upon whether it is straight line sequencing or a brand sequencing we are trying to find out what is the next instruction and go and a repeat this process.

Modern microprocessors are fitted with multiple levels of cache memories. So, consider a processor which is having a cache memory an L 1 cache memory that is also there. So, we have a L 1 cache memory and then we have an L second level of cache memory which we call it as L 2 cache and we have a main memory. So, consider that a c compiler is running the GCC compiler. It is there in main memory then we bring a portion of their program into L 2 and then we bring it to L 1 and it is from L 1 we are going to fetch it to the processor. So, all that we have discussed here. And these all things are going to happen between your L 1 and this processor.

So, in your subsequent lectures we are going to see how modern multicore systems. This is a unicore system with one processer and 1 or 2 levels of cache. Modern processors are having multiple such process modern tiled chip multicore processors are having multiple of such units. And they are going to run multiple task at the same time. So, we are going to see how these things are going to evolve soon.

So, with these we complete the introductory lecture. And we will see deeper into this storage and interconnect aspect of multicore processors in the subsequent lectures.

Thank you.