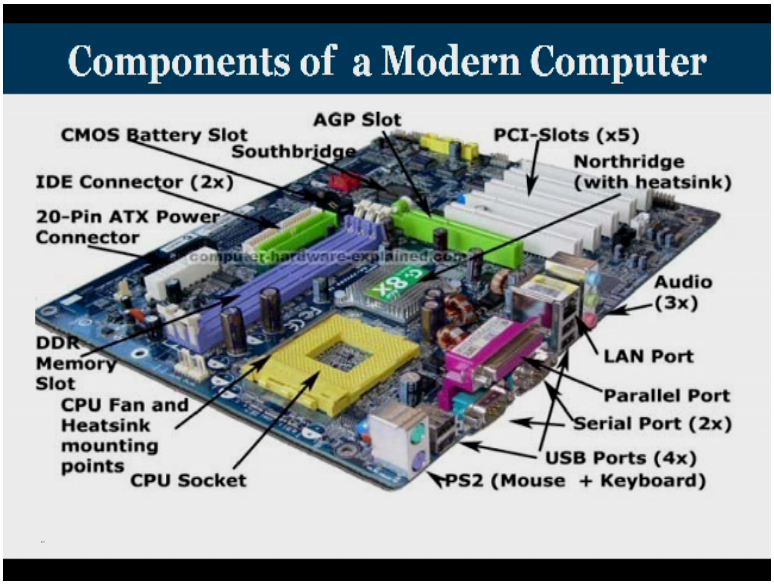
Advanced Computer Architecture
Prof. Dr. John Jose
Assistant Professor
Department of Computer Science and Engineering
Indian Institute of Technology-Guwahati

Lecture-25
Introduction to DRAM System

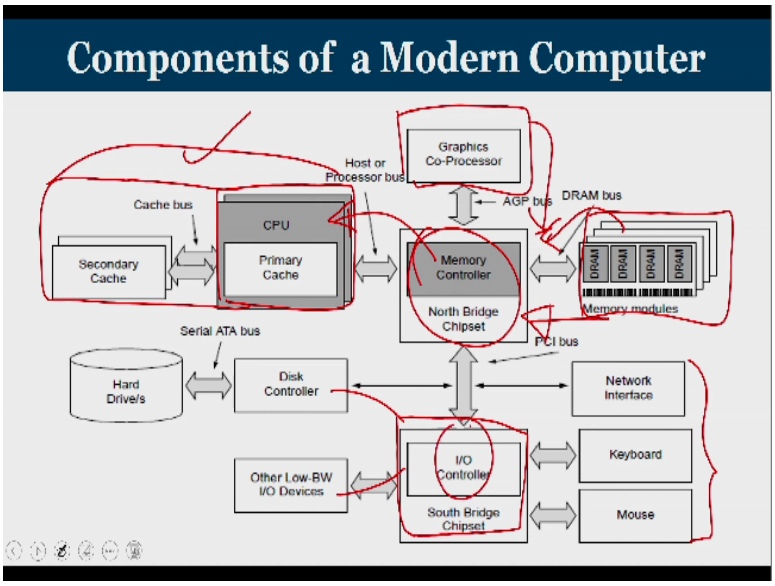
Welcome to lecture number 17 where we are studying about introduction to DRAM system. In the last week, we have seen about the memory hierarchy out of it is the very first component that is cache memory. Now we know that it is not possible to have very large cache memories on chip due to availability of space, cost and power constraints. Now, we move to the next level in memory hierarchy, which is the primary memory generally known as the DRAM system.

(Refer Slide Time: 00:59)



This is the general layout of the motherboard, where this is the place where processor are kept.

(Refer Slide Time: 01:06)

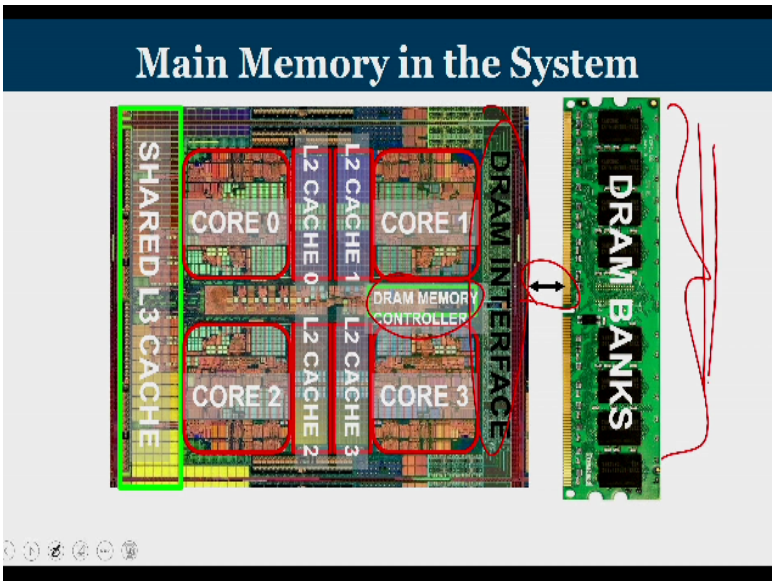


So this is place where processor is kept. And this is called the slot for keeping your main memory. And we have something called north bridge, we have something called south bridge and these are the ports for sound adapter, for network adapter, for display adapter, your USB port, your mouse and keyboards are being connected, it may look like a messy diagram, let us take it into a block diagram mode.

So, where you have CPU and your caches are going to be here and this is your secondary cache as well. So altogether CPU plus caches are kept here. And then you have your DRAM which is off chip, you have a graphics core processor if there is anything or GPUs that is also off chip. And then we have your north bridge chipset, which houses your memory controller in a conventional designs, modern computers of your memory controllers also are moving into the chip.

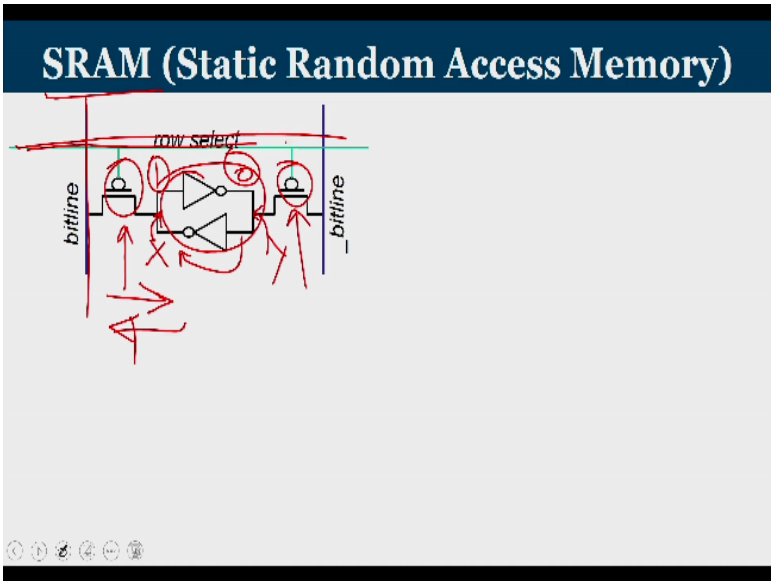
So, all relatively faster devices like graphics processors, your DRAM, which working electronic speed, they are connected to north bridge chipset and electromechanical devices like keyboard, mouse, hard disk can other I/O devices are connected to the south bridge chipset, which houses your I/O controller.

(Refer Slide Time: 02:26)



And in modern microprocessors, which multiple cores are there, we have your DRAM controllers also, which are part of the unit. So you have this is your chip, and this is your DRAM and throws your system bus you are going to connect your controller which is residing inside your chip to the DRAM bands where actual storage happens.

(Refer Slide Time: 02:57)

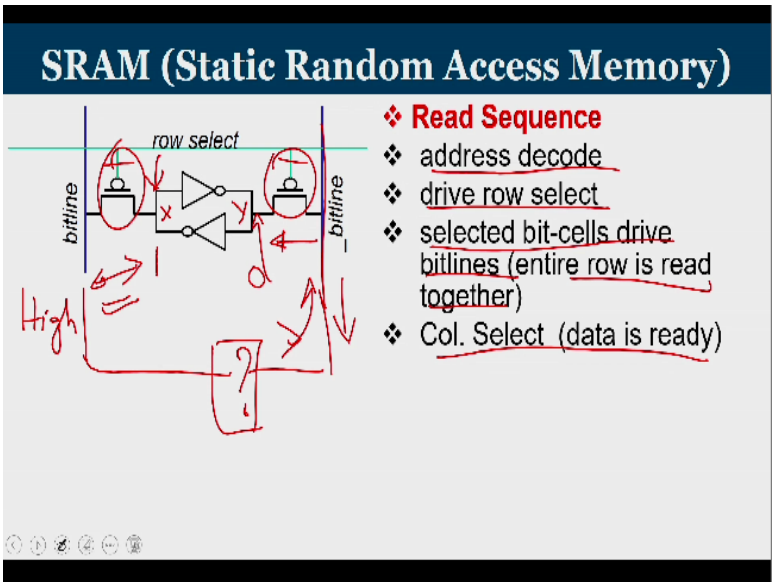


Let us try to understand what is the storage level, there are 2 different types of primary memories, one is can be SRAM based techniques. This is the structure of a single B cell which stores 1 bit of information. So, this is the point where you are going to, let us say call it as X and Y. So, the point X will store the corresponding value and Y will store the compliment value. So, you have row select line which will activate these 2 transistors.

And then whatever the value that is there in the bitline depending on a read or write operation, you are charged flows either from X to bit line or from bit line to X and this is called supposed to the value here is 1 passing through north gate will become 0. And again passing to the second north gate the value will be become 1. So, this 1 and 0 is actually stuck in this range. For that to happen these 2 transistors that have been drawn should not permit the charges to go to the bitline and bitline bar respectively.

So, the transistor act as a conductor for a very short time only when the row line or the row select is enabled.

(Refer Slide Time: 04:07)



So, in short to summarize, whenever we wanted to perform a read or write operation, activate these 2 transistors by the row line for a very short time and depending on the operation either charge flows from here to the bitline or value in the bitline flows to the charge point. So, basically you have an address decoder. So, the address decoder will activate one of these lines and it will drive the row select and the selected bit cells drive the bit lines then and the column select is being used.

So, now, let us see how reading and writing happens in the case of an SRAM. So, the concept of reading is very simple. Imagine that, you have a value 1 that is stored here. Imagine a value 1 is stored there. Initially what we do is then here it will be 0 initially you keep both the bitline and

bitline bar at high potential. Now you have a sense amplifier that is being connected here. So, the sense amplifier is connected with both bitline and bitline bar.

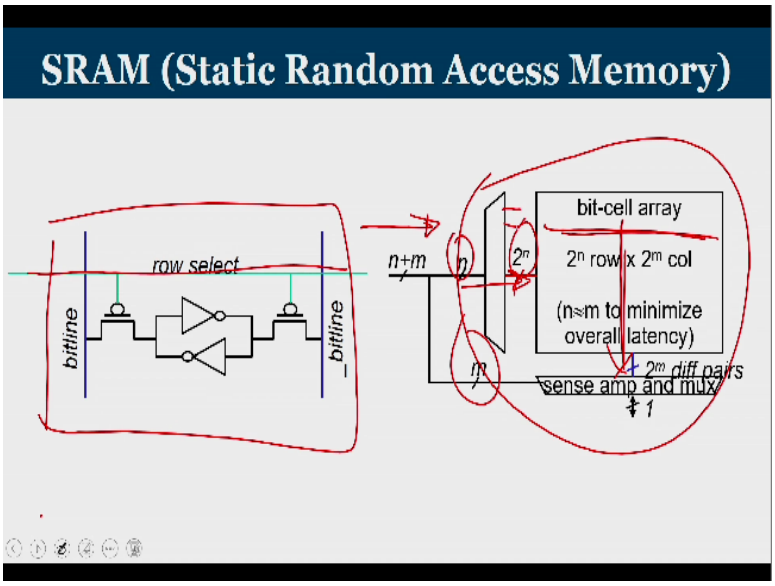
When you want it to perform a read operation this transistor is been put in on position such that this will conduct and when it is 0 when this value is 0 and if this is in high potential, the charge will flow into this. The moment you see that the charge is going to flow into that and whereas in this case charge is not going to flow and that is going to be released. So, you release this one, this transistors immediately.

So, if you look into sense amplifiers you see that this value on the bitline bar is coming down the voltage is coming down because the voltage from here flows into the bit cell it flows into the bit cell, whereas here there is no difference. So, when the value of bitline and bitline bar which was initially the same, after some time you find that bitline bar is coming lower than the value stored is 0 where in this point that means the actual bit value is 1. So, this is value X and value Y.

So, to rephrase initially your bitline and bitline bar have the same voltage when you enable the read operation. There is a flow of charge from bitline bar into the point Y because Y is zero potential and there is no flow of charge from between x and the bitline. So, when you sense that bitline value is larger than between bar value after some time, it means that x is storing your value 1.

Imagine you are actually storing a value 0 and in the same operation then bitline voltage drops whereas bitline bar voltage remains high. In that case you understand that your value of X is equal to 0.

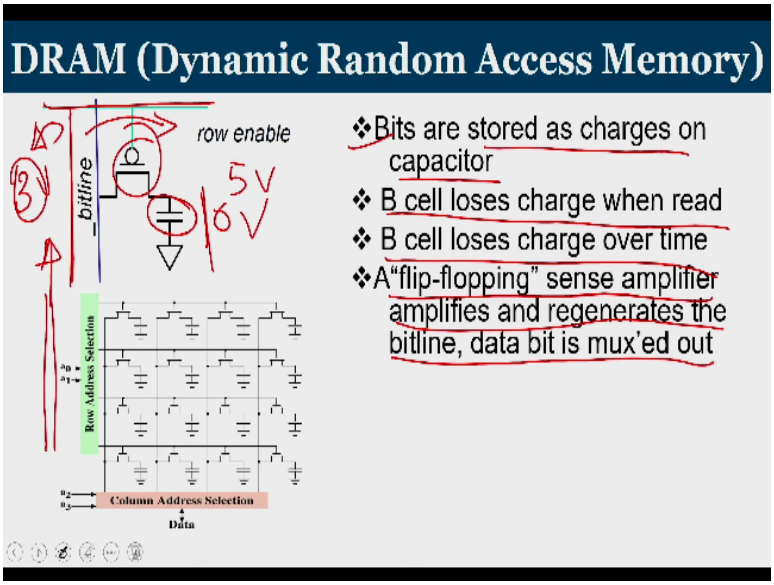
(Refer Slide Time: 07:04)



So, when you get an address, you divide the address into rows and columns and you have decoders, which will take up this n bit input and provide 2 power and outputs. So one of these 2 power n output is going to be high and that select the row and using your column bits, you are going to transform the corresponding data. This is the way how you build large B cells. So, it is nothing but these kind of smaller B cells are being replicated row wise and column wise.

And all these are all selects are connected by a common output of a decoder, entire row is selected and then appropriately you can sense.

(Refer Slide Time: 07:42)



And now we move to another category of memory which is called DRAM. Here we use a capacitor. If the capacitor have the access to a potential difference between the parallel plates of a capacitor then that is called logic 1 and if there is not much potential difference or the potential difference is less than a threshold, then we consider logic 0, here also we have a row enabled and then you have a transistor that is going to help you.

So, once the transistor is on, then charge will flow from the bit line to the capacitor plates or from capacitor back to bit line depending on which is having higher potential. Here also let us try to understand what happens, you keep a voltage for bit line, let us say it is an average voltage, you imagine that your logic 1 is 5 volt. So, you keep it as 3 volt if already the capacitor is charged, then what happens is this value is going to increase from 3 to 3.1, 3.2.

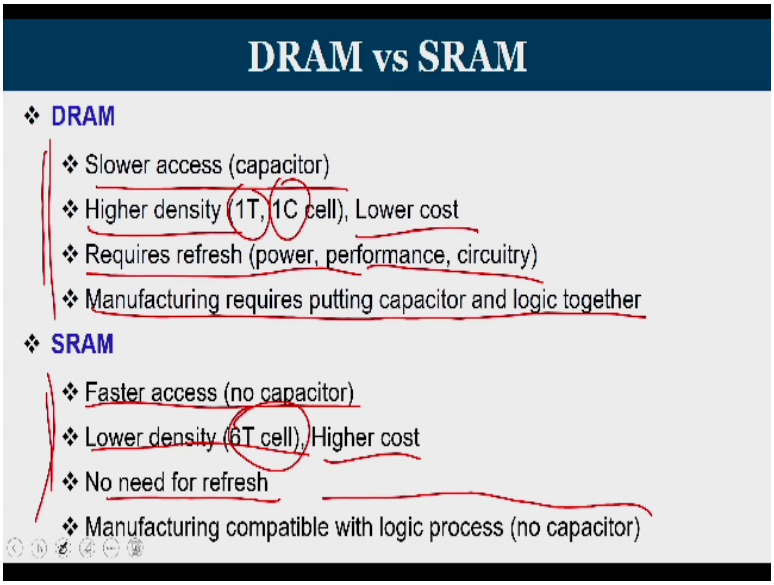
Because from hyper potential it flows. So, when there is an increase in the voltage in the bit line that is understood as logic 1. Similarly, if the value is 3 and if the capacitor value is already 0 volt or 1 volt say examine, in that case value flow from bitline back into the capacitor. So, there is a dip in the value in the bitline voltage. So, initially we supply a value in the bitline, if the value is increasing because of the charge that is there in the capacitor plates then we consider it as logic 1.

If the value in the bitline is decreasing then we consider it as logic 0. So, here with 1 transistor and 1 capacitor we are able to store a binary value and this is the way how we build large memory cells, where you arrange your binary cells in row wise and column wise and few bits of the address is used for selecting row and few other bits are used for selecting column. So, in the case of DRAM cells bits are stored as charges on capacitor.

And B cell loses charge when read and B cell looses charge over time also. Now, the property of this capacitor is if you read any way to discharge, if you are not reading also there is a leakage between the parallel plates of a capacitor and because of this leakage after some time whatever is the potential differences that you have maintained between the parallel plates that is also lost. So we need to have some mechanism such that we can refresh the charges that is being stored.

So we use a flip flopping sense amplifier that amplifies and regenerates a bitline and data is multiplexed out.

(Refer Slide Time: 10:17)

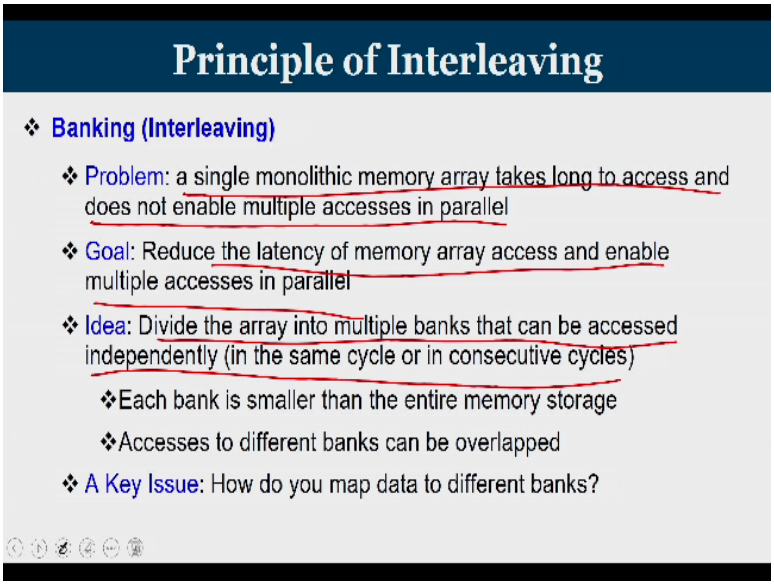


So we will try to compare what is the difference between SRAM and DRAM, So, DRAM is slower access because you need to discharge the capacitor and see what was the value that was stored in the capacitor, but we can have a higher density of storage because we require only 1 transistor and 1 capacitor to realize one bit storage. So it is lower in cost, but it requires a refresh circuitry which will consume a little bit of power.

Now manufacturing requires putting a capacitor and the logic that is transistor together, challenging one. So because of this issue is it is not a kept inside your processor IC, generally memories of chip are built using DRAMs and memories on chip are basically we need to use SRAM there faster with this no capacitor involved all our electronic components, but you require 6 set transistors.

We have 2 not gates and 2 transistors and each of the not gate requires 2 transistors or alltogether we require 6 transistors to realize 1 bit storage. So that makes it higher cost. But there is not refresh circuit needed and there is no capacitors, but manufacturing is compatible because we use only logic circuits not capacitor there. SRAMs are basically used for building cache memory, and DRAMs are used for building main memory.

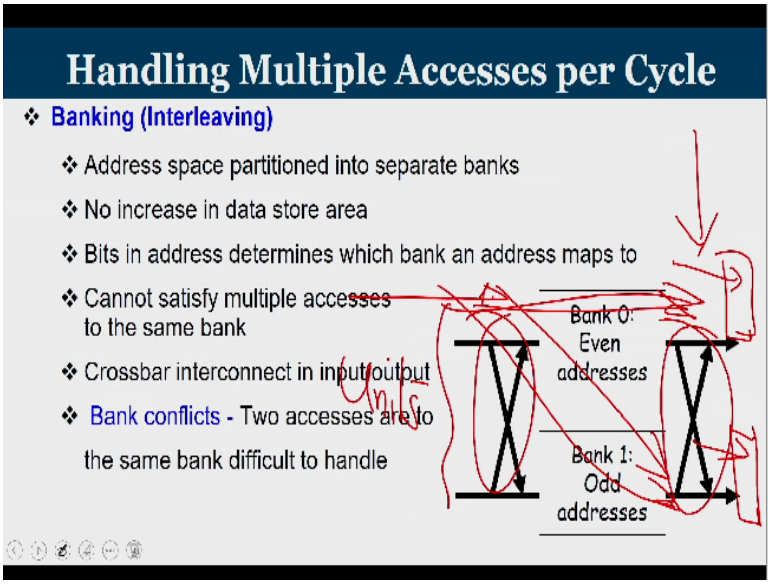
(Refer Slide Time: 11:32)



Now we will try to understand the principle of inner interleaving or it is also known as banking. So we have learned in cache memory that a single monolithic memory will take longer time to access. So if you can split your memory into multiple smaller components that is better. So the goal is to reduce the latency of memory access and enable multiple access in parallel. And how can do that, divide the array into multiple banks that can be accessed independently in the same cycle or in the consecutive cycle.

So each bank is smaller than the entire memory storage and access to different bank can be overlapped. But how do you map data different bank, it is a choice that you have to make.

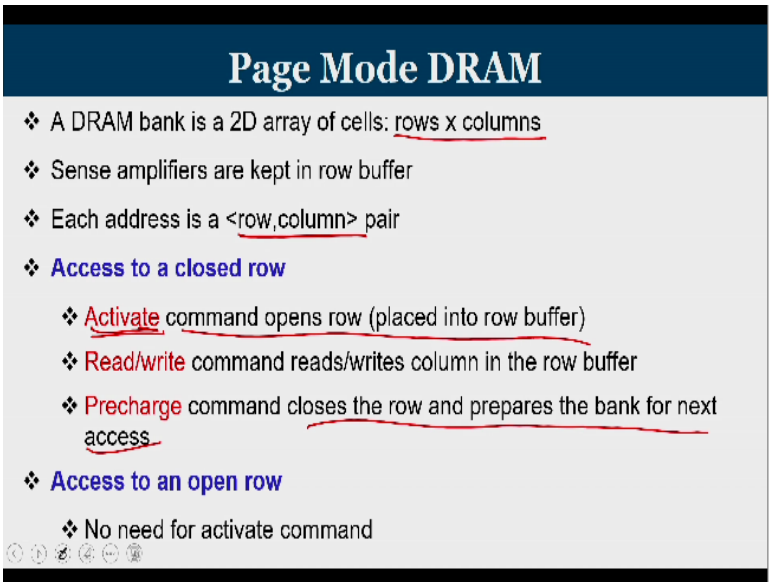
(Refer Slide Time: 12:12)



So think of a case that let us say you are going to have different unit that is going to connect to your memory. If you imagine that all even addresses location, that address row, address 2, address 4, they are part of this bank, and all odd addresses like address 1 3 5 7 and all that is part of this. So from the incoming one, if you come to know that the address is 7 then you pull it over here.

If the address is 6, you pull it over here. Now, let us say I am going to get the bulk of addresses. If the even addresses I will push it into this queue. If it is old address and put it into this queue, eventually, one is been taken from here and one has been taken from here, and that will take care of your parallelism. But we cannot satisfy multiple accesses that are mapped into the same bank. So, these kinds of crossbar connections will help you to connect to any location. And 2 access are to the same bank then we call it as a bank conflict and that we have to solve separately.

(Refer Slide Time: 13:14)



And DRAM is typically we call it as it is operating in a page mode and DRAM bank is a 2D array, which consists of cells, that is rows and columns. And then we have sense amplifiers that are kept in each of the row, which will help you to find out whether the value store is 1 or 0. Now, generally each of the address has the row number and the column number. So, from the address to split few bits are taken to find out the row.

And few other bits have taken to find out the column and access to a closed row. So, just imagine that the row is not kept in the row buffer. So, you require certain special commands to transfer

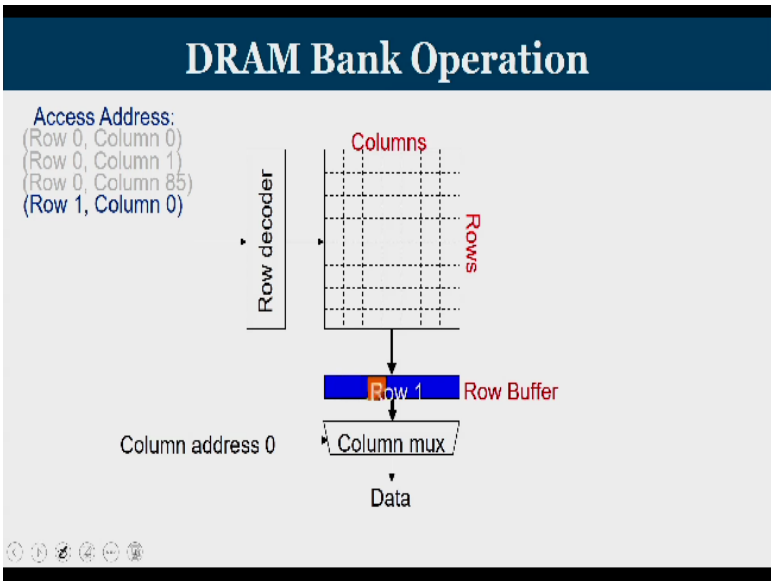
the contents of a row to the row buffer. So activate is what we call, activate is a command that is used to open a row. So once you activate row number 10, the contents of entire row number 10 come to a row buffer.

And then you give the column number that is basically called read or write commands. It performs the column that is operation. And pre-charge basically means the contents of row buffer is stored back into the row. So in DRAM, when you perform an activate command, then the entire contents of one particular row is transferred to row buffer. And we know that when you read a value from a capacitor, capacitor loses the potential difference there.

So it is called a destructive read. So the contents is no longer there in the row, it is there in the row buffer. So we need to perform a precharge operation by which we are storing it back. That is why it is known as a precharge operation. So during preacharge whatever is the content of row buffer it gets stored back into the corresponding row. So when you wanted to activate a new row, prior to that precharge to the previous row is must.

Because the value of the previous row is only available in the row buffer not in the corresponding row. So, once you precharge the row is being returned back, the row is getting its previous value back. now access to an open row. So, once the row is already open, then there is no need for other further activate just perform a read or write command activate command is not required.

(Refer Slide Time: 15:20)



So, let us now try to understand the operations in DRAM, kindly follow the animation that has been shown in the screen. So, you have your DRAM which consists of rows and columns. So, we have a row buffer which can accommodate 1 full row at any point of time initially the row buffer is empty. Consider the case you got a request to rows 0 and columns 0. So, row 0 number has been given through row decoder.

And based upon that, once you give the row address decoder the contents of row 0 is being transferred. So, now row 0 is there in row buffer and then you apply column 0 there. So, the contents which have of column 0 is been transferred through the data bus. Let us say the very new request is to row 0 column 1, if the new request is row 0 column 1 already row 0 is there. So, we call it as a row buffer hit.

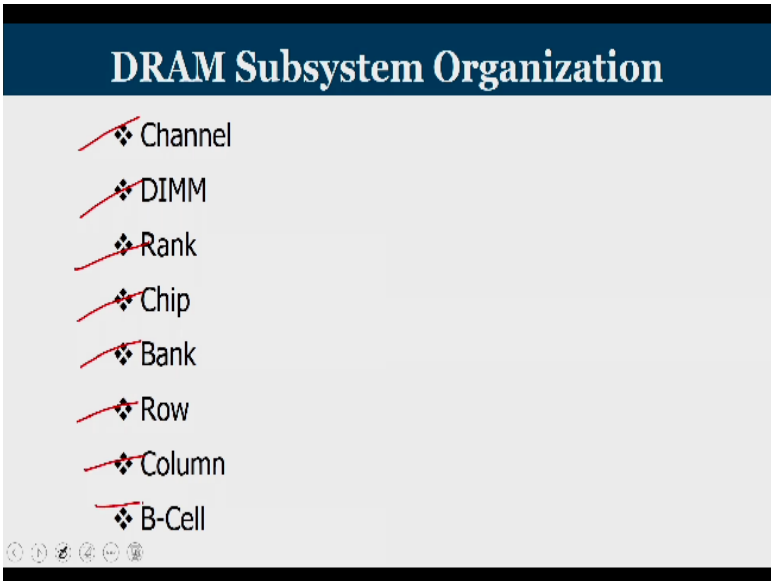
So only column number to be given. So you give column address 1. So the content of column address 1 is been transferred as a data. Now imagine you got the very next request to row 0 column 85. Again, it is known as a row buffer hit because row 0 is already open. So you need to give only column number. So these will actually take less amount of time to transfer the content from memory when there is a row buffer hit.

So give column address 85 to the contents of column 85 has been transferred. Now consider the case the new incoming request is to row 1 column 0. So row 1 is a different row when you look

at the contents of the row buffer and that is what is known as row buffer conflict. So, in the case of row buffer conflict first I have to precharge the contents of rows 0 back into row 0 and then put row address number 1, so, the content of row address 1 is been transferred.

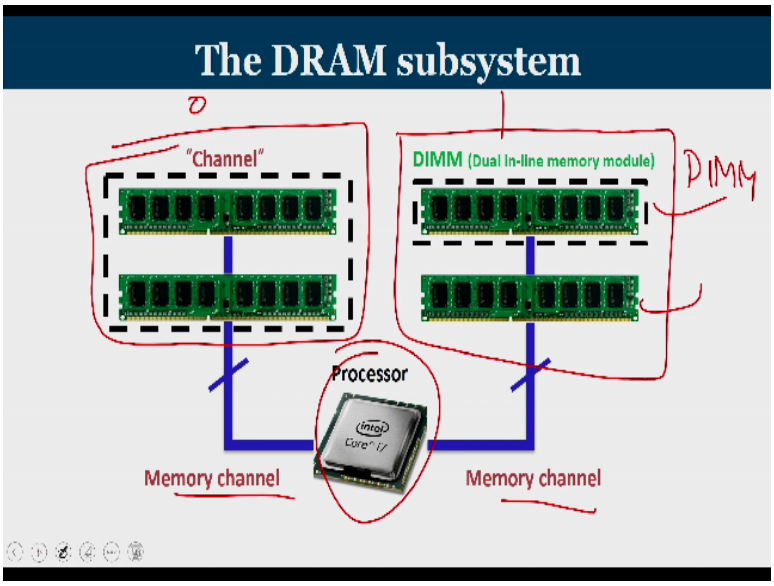
So, row 1 is now residing insert your row buffer and then you apply column address 0. So, when you apply column address 0, the content of that gets transferred to data.

(Refer Slide Time: 17:30)



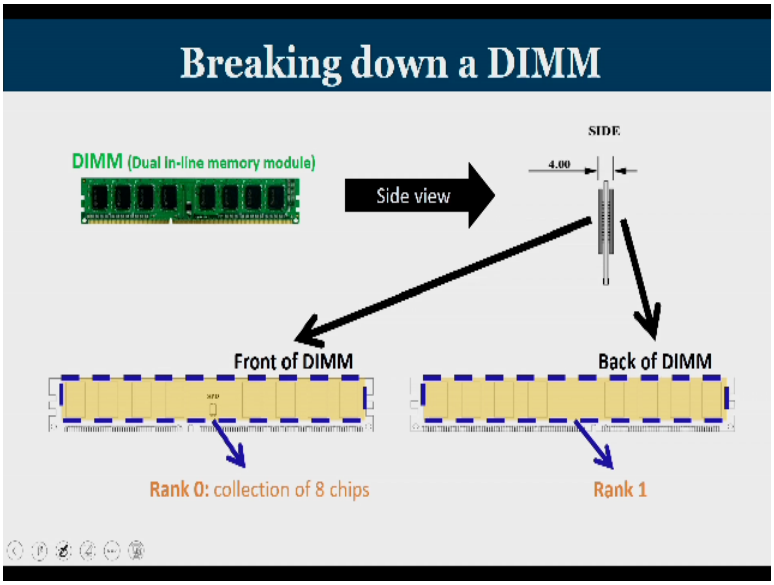
This is the way how generally your DRAM works. Now, let us try to see the organization of the entire DRAM. The entire DRAM structure in modern multicolor processor is divided into multiple sub hierarchies. They are channels, DIMMs, rank, chip, bank, row, column and the B cell.

(Refer Slide Time: 17:54)



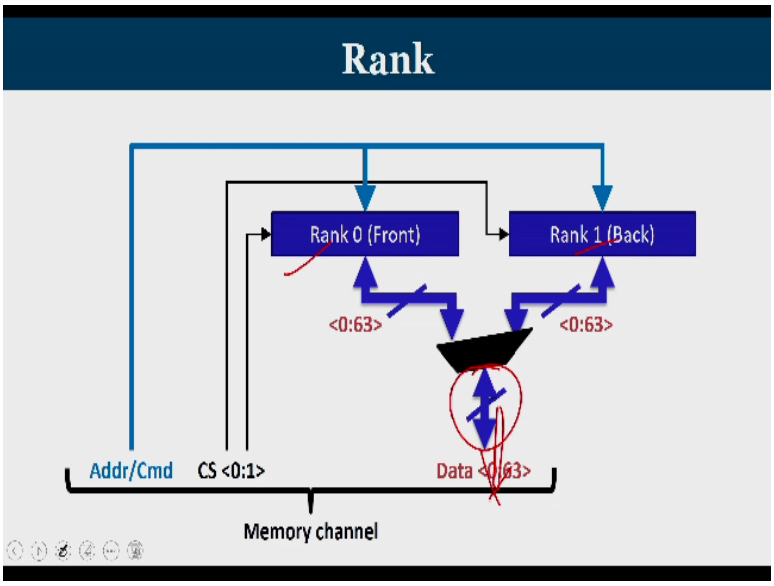
So, when you talk about processor, the latest processor from Intel Core i7 it is having multiple memory channel. So, each memory channel can be defined as an independent bus or independent communication point. So, each channel consists of multiple DIMMs. So, this is one channel, this is channel 0, this is channel 1 and each of the channel has separate DIMM. DIMM stands for dual in-line memory module. So, channel consists of multiple DIMMs.

(Refer Slide Time: 18:22)



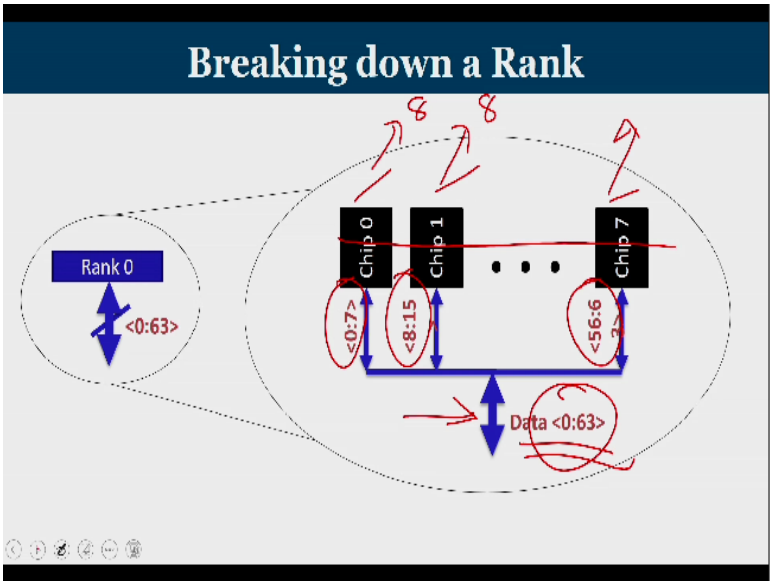
Now, let us take 1 DIMM, this is what typically you see it as a memory module known as a DIMM, the front surface of the DIMM is known as rank 0, and the back surface of the DIMM is known as a rank 1.

(Refer Slide Time: 18:37)



So, this is rank 0 and rank 1 both are connected to the same data bus using a multiplexer. So only data from one of them can move at any point of time. So this is your data bus that has been connected, so you have a multiplexer which will choose. So 1 bit in the address will tell you whether it belonged to rank 0 or rank 1, if that bit is 0 rank 0 is enabled, if that bit is 1 rank 1 is enabled and the remaining bits will tell you what is addressed within that rank.

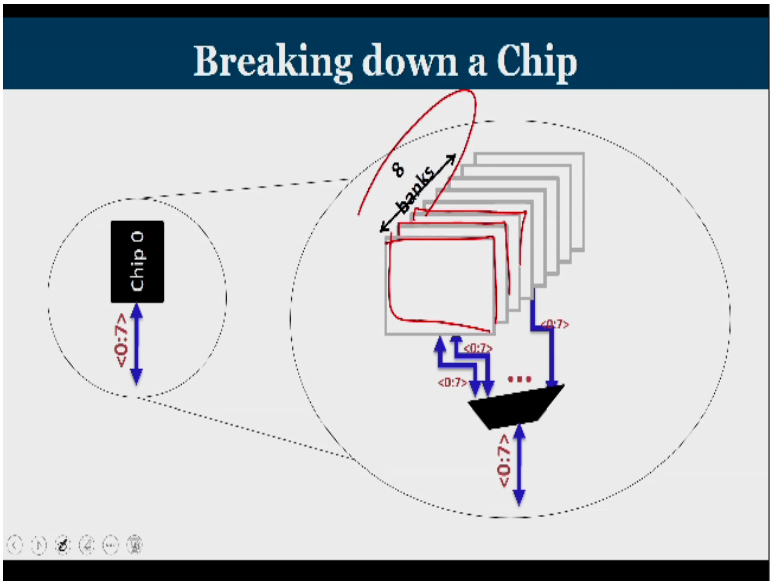
(Refer Slide Time: 19:06)



Now, if you break down a rank, you can see that there are multiple chips and each chip is going to store some fragment of data. So, when you have total 64 bit of data and you have 8 chips number as 0 1 2 3 etc. up to 7, each of this chip is going to give me 8 bit of my data. So, out of the total 64 bit, which is my word, all these chips contribute to 1 by 8 of that word. So, when I

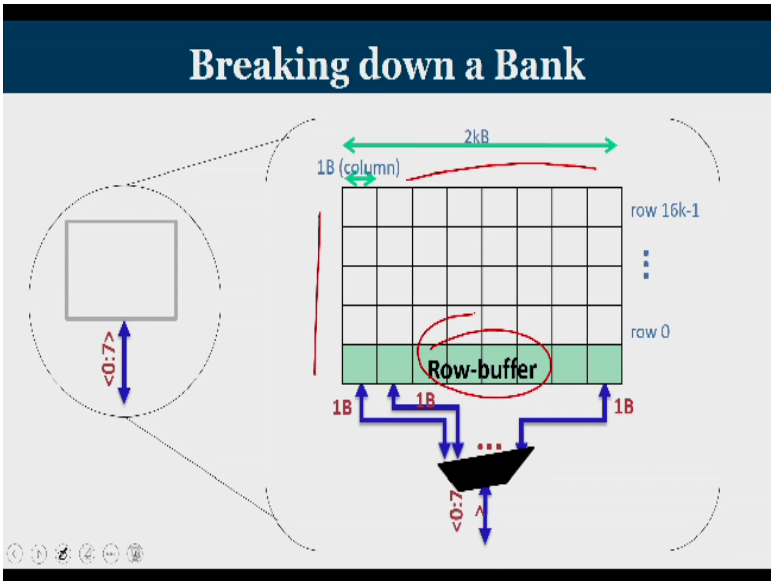
store a number in 64 bit, it is not stored only in 1 chip, it is stored across all the chips, that is the idea of breaking down of the rank.

(Refer Slide Time: 19:48)



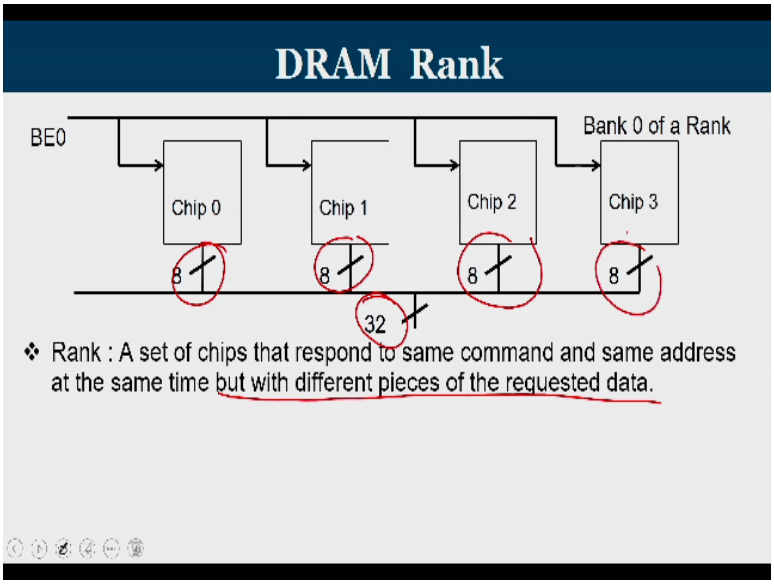
Now if you take 1 chip had a 3D hierarchy and that is what is known as your bank structure. So bank 0 is the first surface behind that we have the next layer, which is called bank 1, bank 2, bank 3 like that and each of the bank has its own row buffers.

(Refer Slide Time: 20:07)



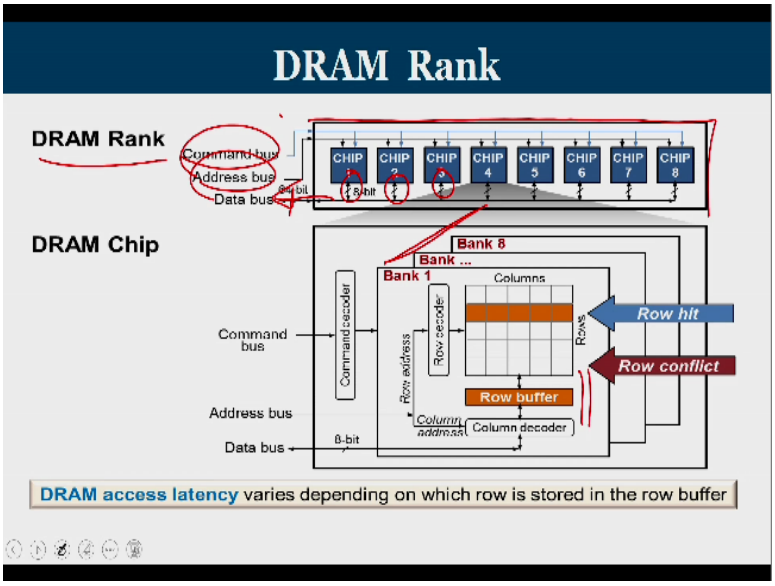
And then you are going to break down further banks. That is what we have seen in the previous animation, you have now rows and columns, you have row buffers. And from the row buffers I am going to transfer the data.

(Refer Slide Time: 20:20)



Now we look into what is there a DRAM rank, that is what we have seen. DRAM rank is nothing but a set of chips that respond to the same command and same address at the same time, but with the different pieces of the requested data. See let us say it is a 32 bit data bus, each chip is going to return me 8 bit of information.

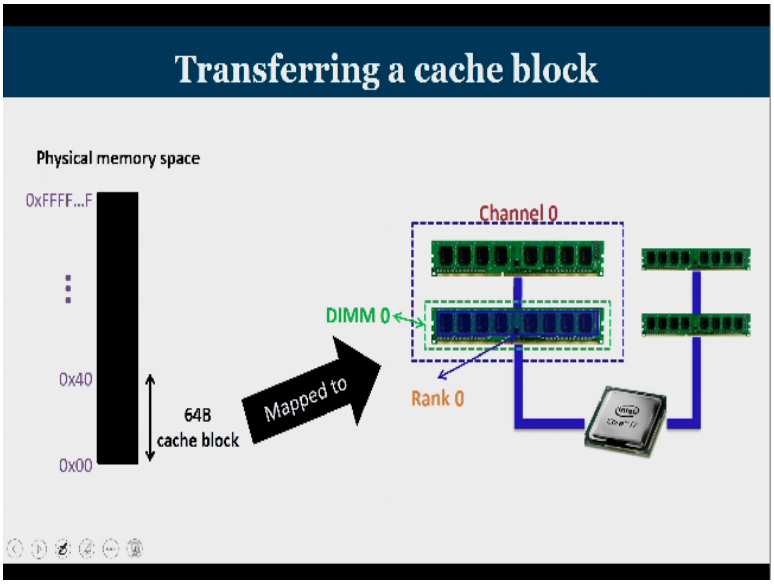
(Refer Slide Time: 20:40)



So this is the logical view that you have seen, this is what is called your rank DRAM rank. So you send your commands, your address, all into this and all of them will go to all the chips, but fragment of data has been given by all of them, that together makes your data bus and bank

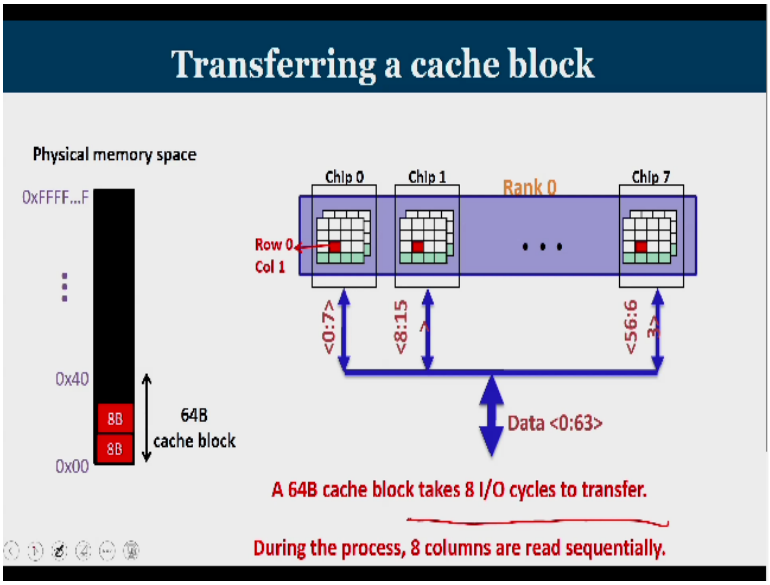
concept is something like a 3D concept, your each of the bank is having rows and columns, and it has shown row buffers.

(Refer Slide Time: 21:07)



Now, we will see just how are you going to transfer data from main memory to cache, we know that it is from the main memory you are going to transfer a block of data. Now, caches generally work in the order of blocks. So any miss that happens in the cache, you have to bring a block of data. And it is not possible to bring a block of data in one stretch, it has to be broken to multiple smaller words, at any given point of time, you can transfer only 1 word at a time.

(Refer Slide Time: 21:34)



So consider the case that we were talking about 1 rank 0 at any given point of time, your data bus can take only 64 bits. Imagine your cache block capacity is 64 bytes. So you are a data bus bit is 8 bytes, you need to transfer 64 bytes of data to make your 1 cache block full. So what you do is you view row 0 and column 0, and that is going to give me 8 bytes of data. Each one will supply me 1 byte of data, so there are 8 such chips together it is 8 bytes of data.

Similar to that, then I moved to row 0 column 1 that also will supply me another 8 bytes of data that is a second word like that is 64 byte cache block takes 8 such I/O cycles to transfer data from the main memory into the cache. During this process 8 columns are read sequentially. This is the way how which the main memory is going to interact with the cache.

(Refer Slide Time: 22:33)

So with this we come to the end of the introductory material of DRAM, we learned about the differences of SRAM and DRAM, it is working principles, how reading and writing happens. And then we are trying to see the organization of DRAM the hierarchy split up of the DRAMs into channels, DIMM, rank, chip, bank, rows, columns and B cell. We will see in about 3 different command, the activate command which will transfer the condense from a row to the row buffer.

The column command or read or write command by which a column number is given, then you perform read or write, and the precharge command that will transfer the contents of the row buffer back into the corresponding row. With this we conclude the DRAM introduction we will

look more into the working of DRAM controllers and how memory mapping happens in the subsequent lecture, thank you.