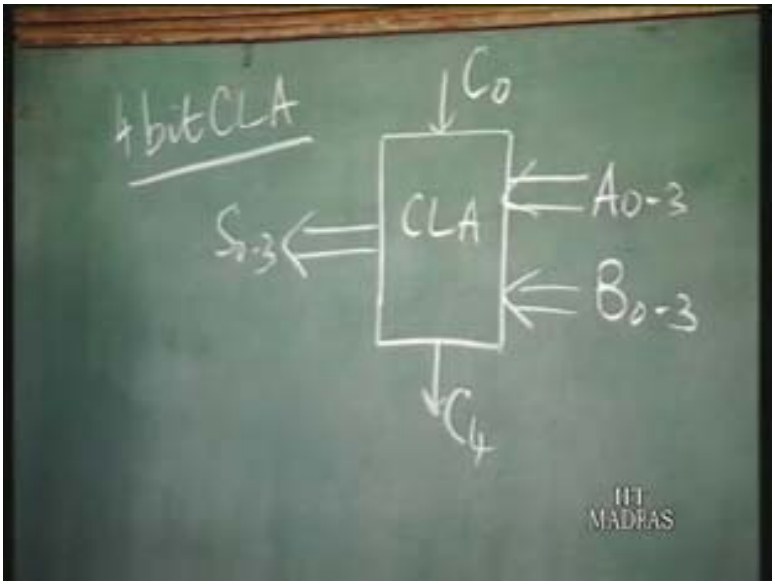
Digital Integrated Circuits
Dr. Amitava Dasgupta
Department of Electrical Engineering
Indian Institute of Technology, Madras

Lecture No # 31

We shall continue our discussion on the use of BiCMOS in larger systems and as an example we were discussing the 32 bit adder. We said that this adder can be realized by using 8, 4 bit carry look ahead adders. You have a carry look ahead adders, where you have A_{0-3} which are the 4 bits as an input, another 4 bits B_{0-3} you have the sum output S_{0-3} and the carry C_0 comes in and the output you have C_4. This is a four bit carry look ahead adder.

(Refer Slide Time: 01:53)



The advantage of using such a four bit carry look ahead adder is that it is a total independent structure and you can realize any multiple of 4 bit adders by just connecting few of these four bit adders. For example if you want to make a 16 bit adder, you just take 4 of them and just connect them together. For the next adder, the carry in is actually the C_4 and the carry out will be C_8 but it is going to be a same structure. If you want 32 bit adder, you have to take 8 such adders and you just connect them together.

That is very important in the case of VLSI because the philosophy in VLSI design is that you require repeatable structures because it is very difficult to actually design every transistor in a VLSI. You must have structures which are repeatable and so this is a

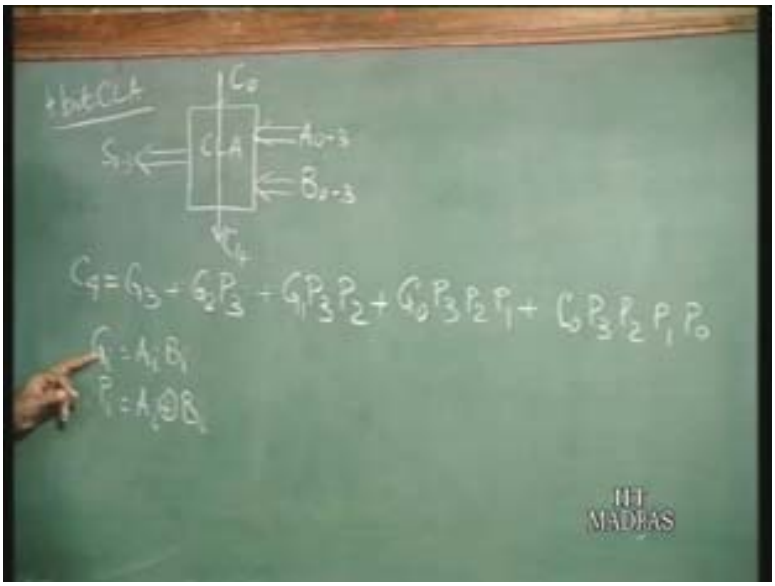
totally repeatable structure which we can go on adding to obtain for example a 32 bit adder or even 64 bit adder or any multiple of 4 bit adders.

Here we have already seen this structure and we had seen that this is the most critical path. That is the carry C_0 to C_4 path. Because when you have 32 bit adders, you have 8 such adders and this C_0 must propagate all the way to the last adder in order to obtain the full addition. This is a very critical path and the delay in this path must be reduced as far as possible. The carry propagation circuit must be fast enough and we have also seen the expression for this carry output. What is it? You have C_4 is equal to either G_3 plus $G_2 P_3$ plus $G_1 P_3 P_2$ plus $G_0 P_3 P_2 P_1$ plus $C_0 P_3 P_2 P_1 P_0$.

That is basically what are the G's and P's? G_i is equal to $A_i B_i$ and P_i is Exclusive OR of A_i and B_i. This is called the generate term that is if A and B are both high then you generate a carry and P_i is the propagate term. that is if the $A_i B_i$ mean one of them is one then only the output is one that is if you have a carry input and one of A or B is one that means there is going to be an output carry. Basically you propagate the output carry that is whatever is coming in as input, if the carry input is zero, the carry output will be zero. If the carry input is one, the carry output will be one if P_i is high, P_i is one. You propagate the carry, whatever is the carry input is propagated to the output.

Here C_4 is high, if you have generated the carry in the bit three term then C_4 will be high or if G_2 was high and it was propagated through PP_3 or G_1 was high that you generated a carry in the bit one stage and it was propagated through bit two and bit three or it was generated at G_0 and it was propagated through bit one, bit two, bit three or there was an input carry which was propagated through all the stages.

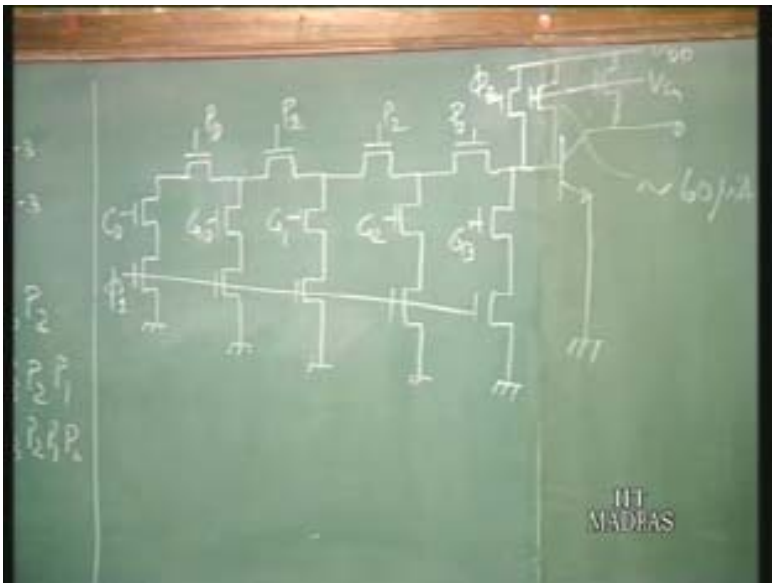
(Refer Slide Time: 05:11)



This is the equation which has to be realized in order to obtain this (Refer Slide Time: 06:35). Now the circuit which is of generally use, this is a very popular circuit called the carry chain actually which is realized to generate this. What we want to see here is how to use bi polar transistors as well as the commonly used circuit using MOS transistors together with commonly used circuit using MOS transistors to obtain higher speed.

Just let us look at the circuit for this. The circuit is something like this. I just write it like this $G_0 P_2 P_1$ plus $C_0 P_3 P_2 P_1$. Now the circuit is something like this. MOS transistors are mostly here, those of you who are familiar with this circuit using MOS transistors, this looks quite similar. Basically what you have is this is the clocked circuit. So you have say phi_1, one clock so this should go low when this G_3 is high then this line should discharge or $P_3 G_2$ or this is $P_3 G_2$ discharging $P_3 P_2 G_1$ or $P_3 P_2 P_1 G_0$ or $P_3 P_2 P_1 P_0$, this is C_0. Now the difference between this circuit and the CMOS circuit is that you have a bi polar transistor here and you have some MOS transistors here. This is a phi_2, so these are certain MOS transistors with certain fixed gate voltages. This is V_{DD}. Now the circuit works like this. This is the BJT output bi polar transistor, now when phi_2 is high, phi_2 and phi_1 are the two phases of the same clock.

(Refer Slide Time: 12:51)



When phi_2 is high, phi_1 is low and phi_1 is high, phi_2 is low. When phi_2 is high what happens is phi_1 is low, so if phi_1 is low that is all this MOS transistors are off. There is no path for the base charge to discharge. This n block, the entire block is off basically, it is not connected to ground. Phi_2 is on so there is going to be a base current for this transistor. This MOS transistor acts as a load resistance basically. This transistor is going to have a base drive, the output is going to be low and care is taken, one has to design it properly so that this effective collector resistance and the base resistance is such that the transistor does not go to saturation.

If the output is low enough but the transistor does not really go to saturation. There is another transistor here, this transistor actually provides a small base current usually of the order, in this particular design which I am talking of. This one is approximately say 60 micro amperes. I just give the reference where they have talked about this. This is very small current constant current is provided. I will come to that why you require a constant current.

In the phase when $phi_{2.}$ is high, basically output is low. So this is the pre charge phase. Now what happens is when $phi_{1.}$ goes high $phi_{2.}$ goes off so this $phi_{2.}$ is off, $phi_{1.}$ goes high so all this MOS transistors turn on. This transistor is going to turn off, if there is a discharge path to ground. There will be a discharge path to ground if $G_{3.}$ is high or $P_{3.}$ $G_{2.}$ is high or $P_{3.}$ $P_{2.}$ $G_{1.}$ is high or $P_{3.}$ $P_{2.}$ $P_{1.}$ $G_{0.}$ is high or $P_{3.}$ $P_{2.}$ $P_{1.}$ $P_{0.}$ $C_{0.}$ is high. Now what happens is even if this is not high so the output should remain low but because the transistor is going to discharge, basically the charges are going to re combine in the base of the transistors.

This transistor is going to go off, shut off so this is small 60 micro amperes current maintains the transistor in the on state. Of course if this discharge path is there, this current will flow through the discharge path and the transistor is going to be off. This small current is used to retain the transistor in the on condition. What is going to happen is normally if it was CMOS circuit, you would have a CMOS inverter in place of that you have a bi polar transistor.

Now what is the advantage of having bi polar transistor here? The advantage of having a bi polar transistor here is that this voltage along this line, at this point say if the maximum voltage you can have is around 0.8 volts or so because that is the base point of the transistor. You cannot have more than say 0.8 volts, the $V_{BE.}$ of the transistor. The voltage swing on this line is limited to around 0.8 volts whereas if you have a CMOS inverter here, the voltage swing would go all the way from 0 to 5 volts and obviously that would take more time.

Here by limiting the voltage swing, you are increasing the speed because the capacitance has to charge only through a smaller voltage whereas if you look at the output of this circuit here, the output voltage goes all the way from 0 to 5 volts. If $V_{DD.}$ is 5 volts, the output voltage will go all the way from 0 to 5 volts, I mean not exactly 0 but close to 0 to 5 volts. This output voltage has a full swing, so this output can drive the input of the next stage. That is if you have you have the series of this four bit carry look ahead adders, so the output of that can be fed in as the $C_{0.}$ of the next four bit carry look ahead adder because it is going close to 0 which means that is going to be less than the threshold of this particular MOS transistor and when it is 5 it means that the transistor is going to turn on. Whereas on the output side, you have the full swing. The input side the voltage swing is less and also because of the larger current drive capability of MOS of the bi polar transistors you get a higher speed.

You see that basically here you are using a BiCMOS but it is not in the form of actually a BiCMOS gate or anything but we have just replaced a CMOS inverter by a bi polar transistor. In fact there is only one bi polar transistor in the entire circuit which consists of so many MOS transistors. Basically the additional area requirement is very small. It's almost the same, you are not using a many bi polar transistors but you are just judiciously using required number of bi polar transistors but the amount you gain out of that is tremendous. I will just refer to this paper from where I took this circuit, T Hotta et al, IEEE journal of solid state circuits, volume SC-21, number 5, pages 808 to 813, October 1986.

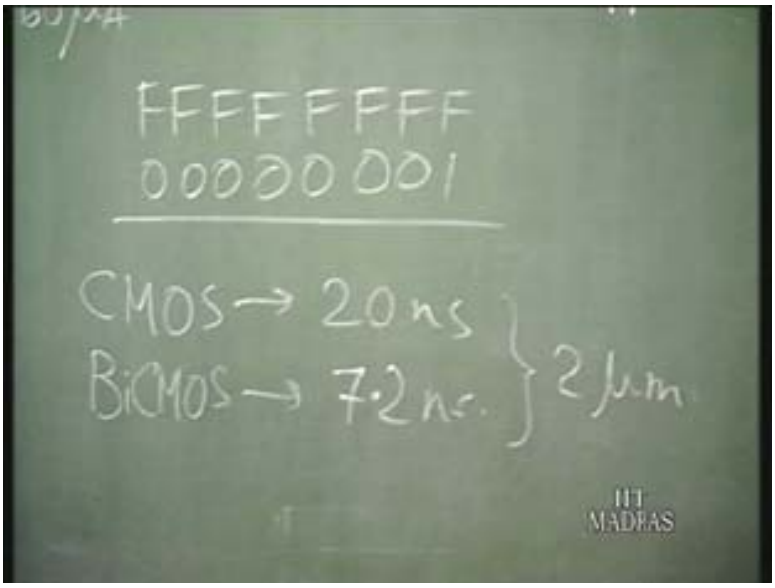
(Refer Slide Time: 17:22)

This is actually paper from Hitachi Corporation in Japan and they claim the first BiCMOS microprocessor which they fabricated. It is a way back in 1986 and of course they have subsequent papers where they have improved their performance but this is the original paper and many of the concepts are actually given in this paper. This is one circuit and they have made a comparison also. They compared with CMOS and they have added FFFFFFFF with 00000001 this particular addition where the carry has to propagate all the way. For CMOS it took 20 nanoseconds. For BiCMOS that is just using one bi polar transistor in the circuit which we have shown, it took just 7.2 nanoseconds.

You can see the amount of improvement one can get. This is for a two micron technology. We can see the amount of improvement one can really achieve by judicious use of bi polar transistors in basically a CMOS environment. We make a BiCMOS, you just have to introduce bi polar transistors in critical paths where you can speed up the operation and you get a tremendous improvement over the corresponding CMOS. Basically it is the CMOS environment but you just introduce bi polar transistors

wherever necessary. This is one example of use of BiCMOS. I shall also give another example, also from the same paper where again as I said they were discussing the microprocessor chip they had fabricated, BiCMOS micro microprocessor chip.

(Refer Slide Time: 19:09)



An important component in a micro processor chip is a ROM, read only memory. A read only memory is used in a microprocessor to store the micro instructions. That is basically whatever instructions you have and then to generate the control signals which is going to activate the different parts of a circuit. It is stored in the ROM. Because it is read only memory, it is just meant for reading you cannot alter the memory whatever is stored there and in a micro processor since the instructions are fixed, there is no need for changing them. So you actually use a read only memory.

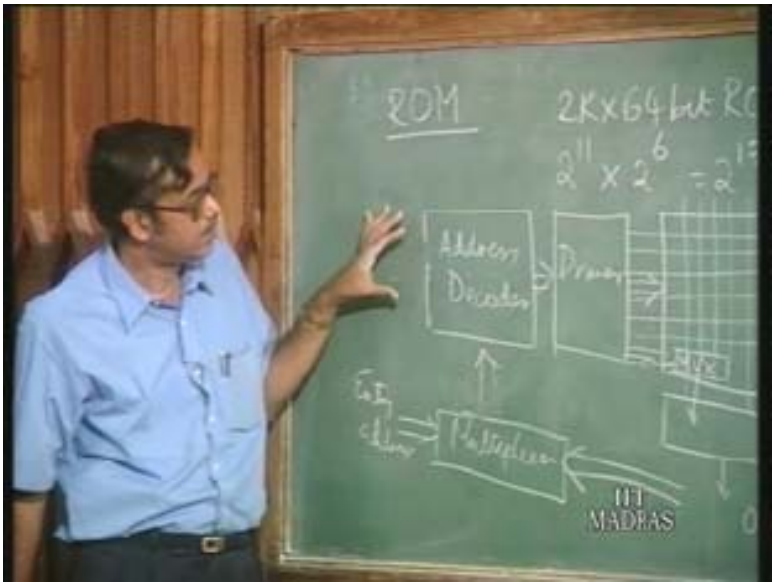
Again we just take it up and see how use of Bi CMOS that is bi polar transistors together with again the basic CMOS environment actually improves the performance. In that particular chip, they use a 2 k into 64 bit ROM. That is there are 2 k words and each of them has its 64 bit wide, the information stored in each word is 64 bit wide. This is actually true for any of these micro coded controllers. That is usually this bit

lengths are very wide in this case. Normally in a memory chip you won't have this type of wide 64 bits or so but anyway we will just discuss this. 2 k into 64 bit ROM, so how many bits of memory are there? 2 k would correspond to 2 to the power 11 and 64 is 2 to the power 6. 2 to the power 17 cells. You must have a memory which can accommodate so many bits of information. The basic structure of this memory is something like this. You have an address decoder, then you have a driver and then you have the array. This is the memory array or cell array and in this array the output goes to some buffers and from here you have the outputs here. In this particular case what happens is where you store micro code because any instruction may have more than one cycle.

The output part of it is fed back again to the next address, a input. So you have a multiplexer which either takes entry address, a new address or in this case 11 bits because its 2 to the power 11 will be fed in as input here to the address decoder. This is the basic structure. Anyway what we are interested in is the memory. How do you have a ROM memory as such in this case? What you have is you have the output of the drivers. Now you are having 2 to the power 17 cells here. Now you would like to have a sort of square array.

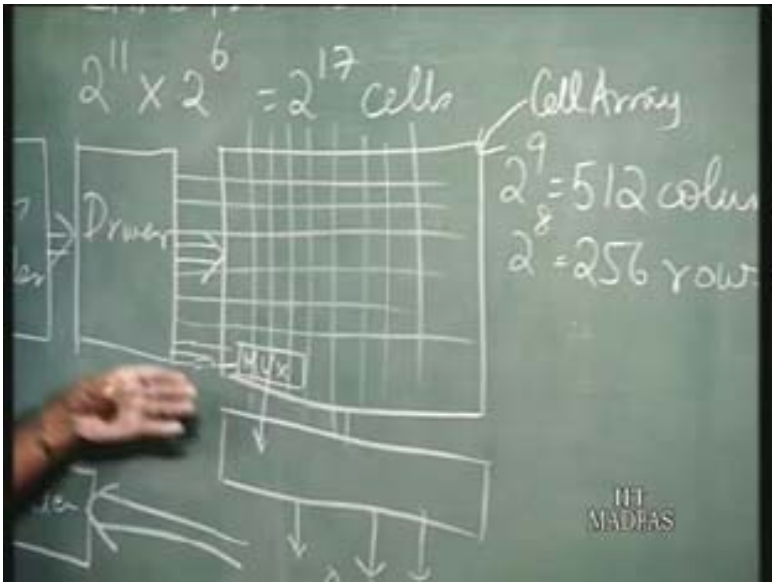
Now one way would be to have a 2 to the power 11 lines here and you have 64 bits. That means we would have a rectangular type of array but this is not good enough because this means that this side or the data output lines would be very long because you will be having 2 to the power 11 lines here and that would mean a lot of capacitance. Basically to reduce the capacitance of the bit line and the word line capacitances, we would prefer to have a square array.

(Refer Slide Time: 27:45)



If you have 2 to the power 17 cells, we would like to have the array, say 2 to the power 8 into 2 to the power 9, its 2 to the power 17. What is done is you basically have two decoders here, so one is you have 2 to the power 11 address lines.

(Refer Slide Time: 28:27)



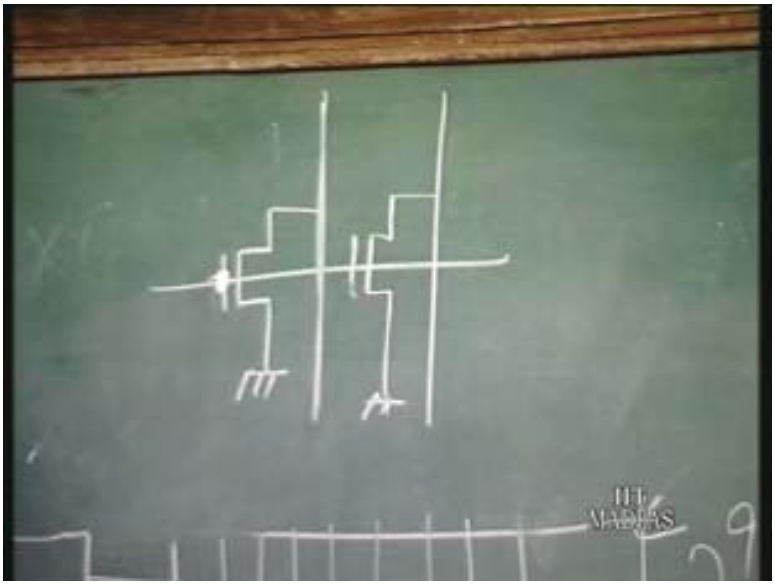
You take 2 to the power 8, this array will be 2 to the power 8 into 2 to the power 9. You have 512 columns and 256 rows. 512 means 2 to the power 9 and 256 rows means 2 to the power 8 rows. What you do is you have 2 to the power 8 rows here. So one decoder will be a 2 to the power 8 decoder. The 2 to the power 11 address lines, 8 of them are decoded to give 2 to the power 8 lines here. Another three of the address bits are again decoded to give 8 lines. There will be another 8 of them and here what you have is you have 64 into 8 columns depending on one of the 3 into 8 decoder which we have, so they choose one of 8 columns.

Basically what you have is you are going to have a multiplexer here, for every 8 columns you have a multiplexer. This multiplexer chooses one of 8 columns. The output of that multiplexer is the one of the 8 columns. You have 64 into 8 columns, the output which you get here is 64. Basically again just to repeat you have 2 to the power 11 address lines, 2 to the power 11 locations or addresses each is going to have 64 bits of data. In order to make it square, we make it 2 to the power 9 into 2 to the power 8, 2 to the power 8 rows. You have 2 to the power 8 rows here, so there is a decoder 8 to 2 to the power 8 in which the output of that 8 into 2 to the power 8 decoder is going to choose one of the 2 to the power 8 rows here.

There are 64 into 8 columns. There is another decoder 3 into 8 decoder which uses the remaining 3 address lines and one of the 8 output lines of the decoder is high. It chooses one of 8 columns, so you have 64 into 8 columns. At a time you choose basically one column. There are 64 output lines. That is clear now. Basically again we come back to the topic which we were discussing that how does bi polar transistor help in improving the speed.

Basically in this ROM, in each of the intersection of the rows and columns you must have a cell. What is the cell like? Basically if you have an intersection here, the cell consists of a MOS transistor which is something like this. So this is connected here. In another location, this may not be connected. Here the gate of this MOSFET is connected to the word line here. Now what is going to be the difference? Basically what is going to happen is suppose you pre charge this line. If this line is pre charged, now if this is connected and when this input goes high, this MOS transistor is on, this MOS transistor can discharge this line whereas if it is not connected here, this MOS transistor is not going to discharge this line. So this line is going to remain high.

(Refer Slide Time: 30:05)

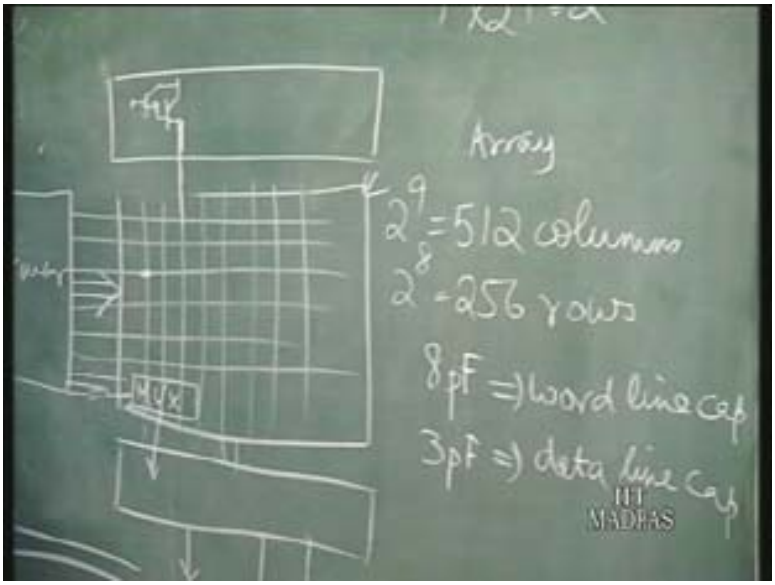


Depending on whether this connection is made or not, you are actually storing a one or a zero in that particular location. If you have a MOS transistor which is connected you are able to discharge that line, so that line voltage here goes to 0. If the MOS transistor is not connected, you are not able to discharge that line and so the line voltage is going to remain at one. What is done is you have a pre-charge circuit. Now this pre charge circuit, so I will just take one of the column which consist of bi polar transistors. In fact that they use Darlington pair type of configuration to pre charge the lines.

Now obviously because of the larger drive capability of the bi polar transistors. Once this line capacitances are very large, for this particular circuit the word line capacitance is 8 Pico farad and 3 Pico farad is the data line capacitance because these lines are very long, you see there are going to be so many transistors here. Say if you have 512 columns so on this line you can have 512 transistors. These capacitances are very large. So this is the word line, this is the data line, so the word line capacitances is 8 Pico farad and the data line capacitances is 3 Pico farad. If you want to pre charge, we have to pre charge this capacitance.

How fast you can pre charge depends on how much current, you can drive to charge the capacitance. For larger drive capability you use bi polar transistors here to pre charge the bit lines.

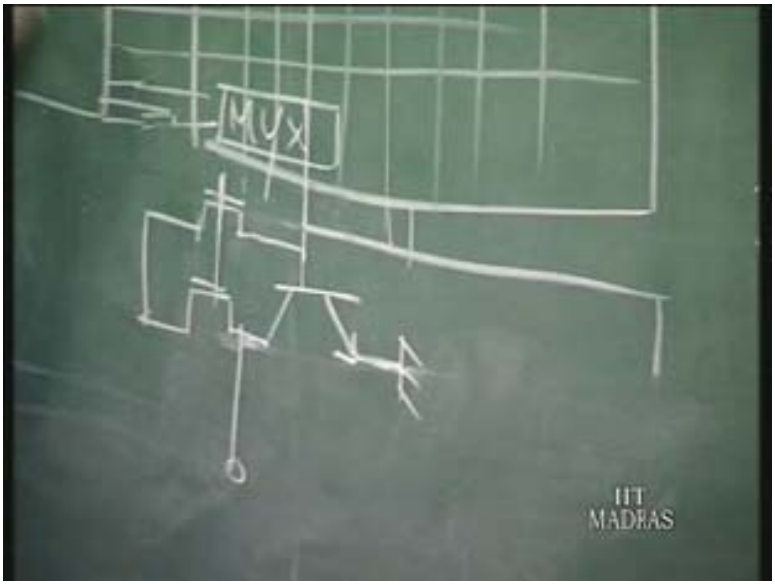
(Refer Slide Time: 32:37)



Then here at this end, you have a similar structure just like the previous case where we discussed this carry chain, similar structure like that. That is you are basically driving a bi polar transistor, I just draw it like this. This is a bi polar transistor. It is coming to the base of a bi polar transistor again and then again as load you have the MOS transistors just like the previous case. This is the output here, so this output is coming here to the base of the bi polar transistor.

Now what is the advantage here? Again this bit line the maximum voltage it can have here is around 0.8 volts. When you are pre charging or discharging the bit line, the swing voltage swing on that line is limited to 0 to 0.8 volts. So you don't have to charge or discharge the capacitance, this is a very large capacitance through a high voltage whereas if this was a CMOS inverter, it had to go all the way from 0 to 5 volts. That is an advantage, the voltage swing on this line is less. You can have faster operation whereas the output voltage here can go all the way from 0 to 5 volts, almost 0 to 5 volts. There is no problem and here the bi polar transistors are used for pre charging and also at this output here. This is just like a BJT inverter, the bi polar inverter here.

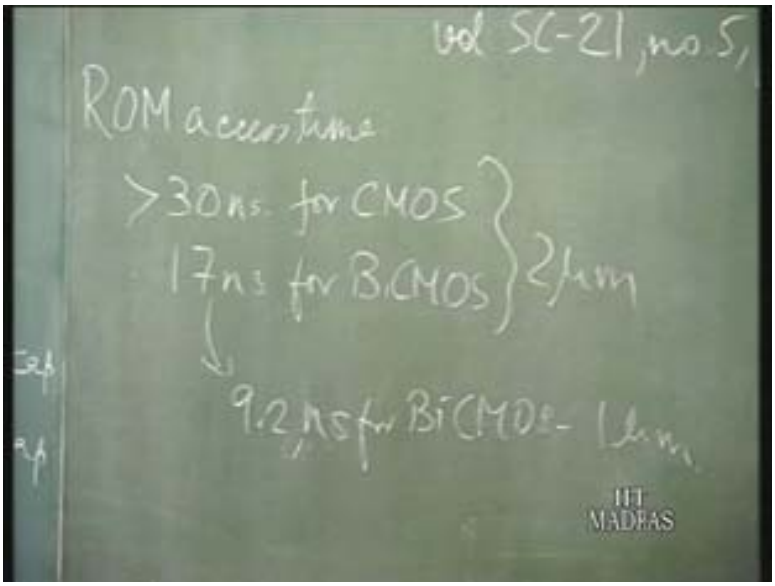
(Refer Slide Time: 34:53)



We have used a bi polar inverter here as well as in this particular circuit for the driver also, we have already seen that a BiCMOS is going to be very good as a driver because of the fact that they can drive large capacitances because you are using the bi polar transistor again. We have already seen that when you compare a CMOS with a BiCMOS, the BiCMOS has got that the variation of delay with large capacitance in which the capacitance is going to be much less. It is capable of driving large capacitance because the word line capacitances of are quite large.

Here again we use bi polar transistors in the driver stages. Here the output is actually from bi polar transistors. These are the three areas where you use bi polar transistors and let us again look at the results as given in this paper. So ROM access time. So memory delays are usually in the form of access times where access time is the time you get the output, after you have floated the address. It is greater than 30 nanoseconds for CMOS and it is 17 nanoseconds for BiCMOS. This is for two micron technology. 9.2 nanoseconds for BiCMOS, one micron.

(Refer Slide Time: 36:56)



You see that again the comparison is with CMOS and we see that again with judicious use of bi polar transistors, one can actually reduce the delays tremendously. In fact the next topic which we will take up for the next few classes is memories. We shall also see the advantages of using BiCMOS compared to CMOS, the different memories. Here again just given an example that bi polar transistors are very much useful in reducing the delays in memories and so BiCMOS as such where you use bi polar transistors judiciously to improve the performance. That is basically to reduce the delays or improve the speed. Here also another thing is if you look at it, the total number of bi polar transistors is going to be very much smaller compared to the total number of MOS transistors. You just use a few bi polar transistors, I mean judiciously. It is not that you are using equivalent number of bi polar transistors.

Actually it is not going to really affect the areas as such but at the same time you extract a lot of advantages. That is about this topic on BiCMOS where we have seen the BiCMOS as the advantages, where it retains the advantages of CMOS at the same time by introduction of bi polar transistors. The bi polar transistors when you compare with MOS transistors, you know they have certain specific advantages. That is the larger drive capability of bi polar transistors. That is they can drive much larger currents, if you compare with bi polar and MOS transistors because of the fact that the current

flows in a vertical way instead of compared to MOS where a current flows in a horizontal direction along the surface.

Bi polar transistors can provide lot of current, if you want to fabricate an equivalent MOS transistor driving the same current, the transistor size would be huge whereas in a bi polar transistor it actually can provide lot of current. At the same time you also know that the trans conductance of a bi polar transistor is much higher than that of a MOSFET because of the fact that in a bi polar transistor, it follows the exponential relationship between the input and the output. That is the output current is proportional to the exponential of the input, exponential V_{BE} by V_T. So only a small change in the input voltage is required to make a very large change in output current whereas in the case of MOS transistors, it is a square law relationship. The GM or the trans conductance is very low. You actually require a small change in input voltage to make the large change in output current in a bi polar transistor, so that is the advantage.

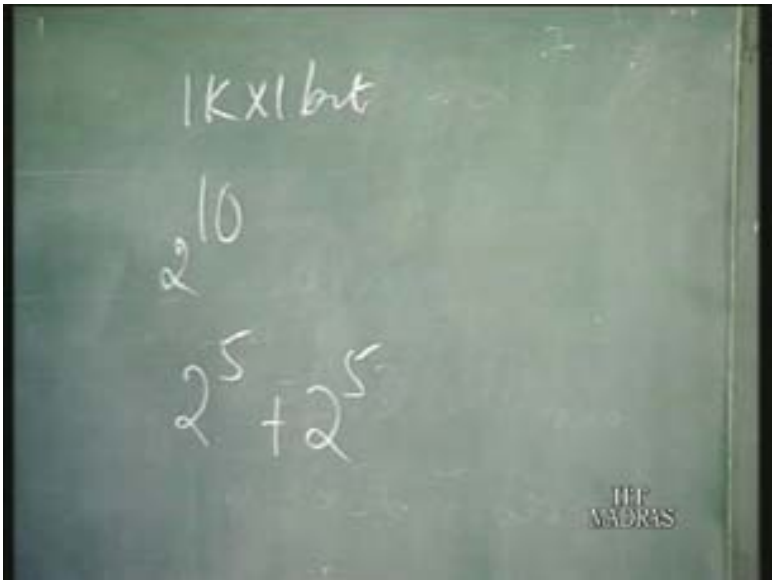
If the input line is having a large capacitance because the bi polar transistor, the input voltage variation requirement is much smaller. The time required is going to be much less to change the input voltage so that is the advantage. For all this reasons BiCMOS as such has a lot of advantages and it is used extensively in modern circuits where the problems related to technology has been removed because previously the problem was that CMOS and bi polar, it was difficult to fabricate on the same wafer but once that problem has been removed, people take both the advantages of bi polar and CMOS and make BiCMOS circuits realize better and better circuits. That is about BiCMOS.

The next topic which we shall take up is memories. We have been mostly discussing logics logic families and gates, etc but now we shall just move over to larger systems as such where memories form a big chunk of the circuits which are being fabricated now, the large scale integrated circuits. we shall take that up and in that also we shall see the advantages of the different types of memories that is we shall take up the static RAM, the dynamic RAM then the ROM and we shall also see how each of this cases for example static RAM, you just have the CMOS circuit or the BiCMOS how it is going to be better or i mean if you make a BiCMOS static RAM where you are going to introduce the bi polar transistors and how does it really improve the performance of that particular memory.

These are some other things which we shall take up in next few classes. I will just give you the block diagram of a memory chip, just as an introduction. A memory chip consist of, so you have the addresses, you have the address lines. A memory will have some addresses and in each address there will be some data. Suppose you say that you have a 2 k into 8 bit memory, so this refers to a 2 k addresses. 2 k is actually 2 to the power 11, 1 k is 2 to the power 10. So 2 to the power memory locations and each of them, you have an 8 bit information. Now memory chip would consist of the address inputs. Suppose let us consider a memory which is a one bit memory, there is a 2 k or

say 1 k into one bit memory. This would consist of 2 to the power 10 addresses and each of them has one bit.

(Refer Slide Time: 45:12)



What is done generally is you basically require a decoder because the address will be in the encoded form. So that in 10 bits and each of this combinations refers to an address, so you have to decode it. Now if you use a 2 to the power 10 decoder, you will have at the output 2 to the power 10 lines, they are far too much. What you do is, you do a two dimensional decoding that is you have 2 decoders each of 5 bits. You have one 2 to the power 5 decoder and another 2 to the power 5 decoder so you have just so many lines now, instead of 2 to the power 10 lines. So it is just 2 to the power 6 instead of 2 to the power 10, so that is the advantage. You have an x buffer and then you have a decoder.

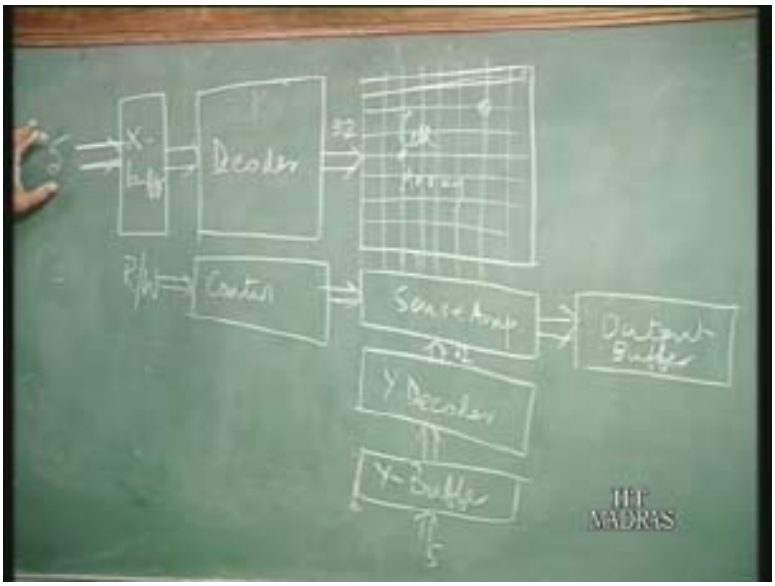
If you have 2 to the power 10, so this you have 5 lines say, so here at the output of the decoder you will have just 32 lines and then you have the memory array. You have an array and then this is the cell array. The memory cell, each cell stores one bit of information and then we have another set of similar y buffer, then y decoder and then

you have what is called a sense amplifier. The sense amplifier is used to sense whatever information is stored in that particular memory locations. Here you have 32 bits, if it is 5 here you have 32 bits. so here you have a sense amplifier. Basically for each of this columns, you will have one sense amplifiers. Here you have this array which is 32 into 32 and each of this intersections, you have a cell. At each of this intersections you have a memory cell. You have 32 into 32 memory cells so 2 to the power 10 cells, storing 2 to the power 10 bits of information. You have a sense amplifier for each of the cells.

Basically what is done is the y decoder does a multiplexing operation sort of and you choose the output of one of these columns. The x decoder what it does is, it chooses one of the rows, so it activates just one of the rows out of these 32 rows. In each of this columns, only one row is activated. So at each of this column output what you get is the information corresponding to that particular row which was selected.

Now what you do is here you choose one of these 32 columns so that basically you are choosing information corresponding to one particular cell and whichever output is there goes through an output buffer to the output and then you also have a control block which is fed to the sense amplifier in which actually you require information like whether you want to read or write and these sort of information. For each of them you have to generate the particular control signals which is going to drive all this different circuits.

(Refer Slide Time: 48:57)



Basically this is the block diagram of any memory. Just to repeat, you have an x buffer and a y buffer. If you have 2 to the power 10 memory cells, so you have 10 address bits. So you have 5 here, 5 here, so you have 32 rows. You are selecting one of the 32 rows and here you are selecting one of the 32 columns and at the intersection of each

row and column you have a cell. So corresponding to that if you are reading, the sense amplifier is going to sense what is there in that cell and you get the output. If you are writing, it must be able to write, if the write signal is high, you must be able to you write into that particular cell. Here what you have done is a 1 k memory.

Now suppose you want to have 1 k into 16 bit memory, what you have to do? You are going to have 16 such memories. Of course the problem is if you have 16 such memories, this is going to be again a rectangular array. What you can do is, if you wan to have square array that means you are going to have 16 k cells. Now 1 k into 16, 16 k cells. So what you do is instead of 5 address lines, you can have 7 address lines here. Then that means you have 2 to the power 7 rows and so you will also have 2 to the power 7 columns.

Basically giving you 16 k and then you again have to multiplex or choose or basically you get for each block of 16 columns, so you get a 16 bit information. That way you can make a square array. Basically if you know how to make a one bit memory, you can reproduce it, generally to make a 16 bit memory. These things we shall discuss in the next class. We shall see the different types of memories that is whether it is a bi polar memory or if you want to make a CMOS memory or a BiCMOS memory. These are basically ways of realizing the memory but also we can look at static RAM, dynamic RAM and for all this cases, we shall look into how to realize them using these forms.