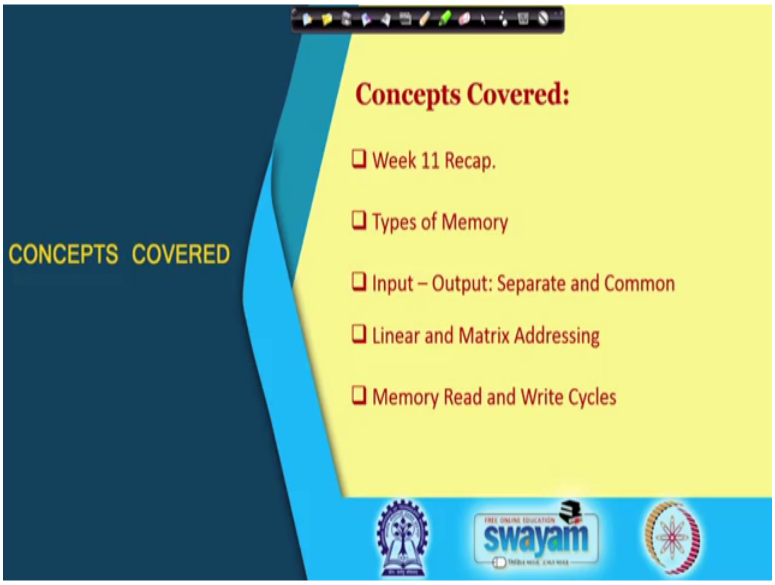
Digital Electronic Circuits
Prof. Goutam Saha
Department of E & E C Engineering
Indian Institute of Technology, Kharagpur

Lecture – 56
Introduction to Memory

Hello everybody. We are in week 12 the last week of this particular course. So, in this week we shall discuss Memory.

(Refer Slide Time: 00:23)



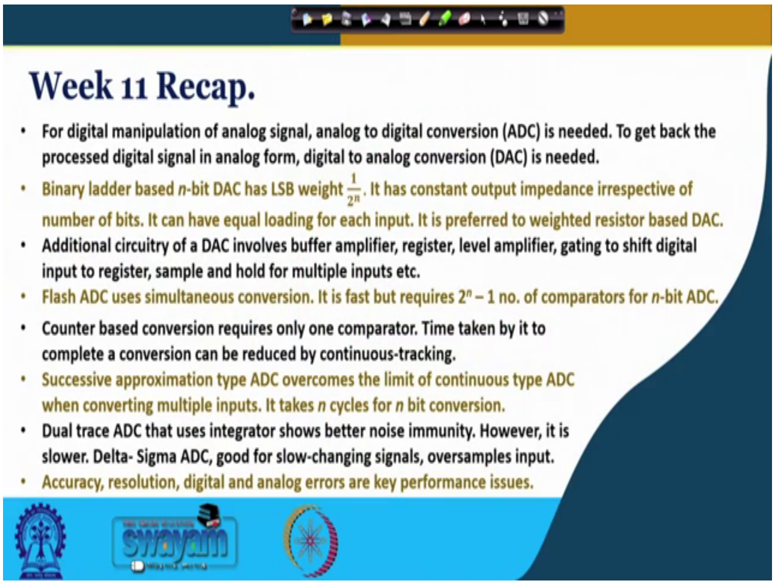
CONCEPTS COVERED

Concepts Covered:

- ❑ Week 11 Recap.
- ❑ Types of Memory
- ❑ Input – Output: Separate and Common
- ❑ Linear and Matrix Addressing
- ❑ Memory Read and Write Cycles

So, before we begin with memory its different types, how input output is arranged the addressing methods and some in discussion on memory read and write cycles we shall have a quick recap of what we discussed in week 11 that is last week.

(Refer Slide Time: 00:41)



Week 11 Recap.

- For digital manipulation of analog signal, analog to digital conversion (ADC) is needed. To get back the processed digital signal in analog form, digital to analog conversion (DAC) is needed.
- Binary ladder based *n*-bit DAC has LSB weight $\frac{1}{2^n}$. It has constant output impedance irrespective of number of bits. It can have equal loading for each input. It is preferred to weighted resistor based DAC.
- Additional circuitry of a DAC involves buffer amplifier, register, level amplifier, gating to shift digital input to register, sample and hold for multiple inputs etc.
- Flash ADC uses simultaneous conversion. It is fast but requires $2^n - 1$ no. of comparators for *n*-bit ADC.
- Counter based conversion requires only one comparator. Time taken by it to complete a conversion can be reduced by continuous-tracking.
- Successive approximation type ADC overcomes the limit of continuous type ADC when converting multiple inputs. It takes *n* cycles for *n* bit conversion.
- Dual trace ADC that uses integrator shows better noise immunity. However, it is slower. Delta- Sigma ADC, good for slow-changing signals, oversamples input.
- Accuracy, resolution, digital and analog errors are key performance issues.

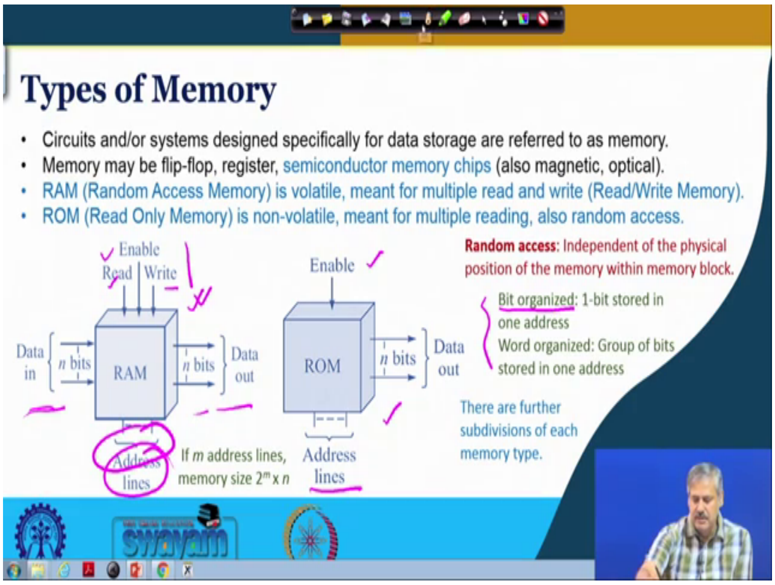
So, in the last week we discussed analog to digital conversion and digital to analog conversion. And we started with digital to analog conversion and we found that binary ladder based DAC is preferred over the other method which is weighted register based DAC from the loading point of view, the output impedance point of view; so there are certain advantages and this is you know more common. And we had seen that other than these weighting part that we get where the binary weights get mapped to the corresponding analog output.

So, there are other circuitry in the form of power amplifier register level amplifier all those things are required to make a DAC work. And then we move to ADC and we found that flash ADC the simultaneous conversion method that is very fast, but the number of comparators required is very large ok. And in the counter based method we found that the number of comparator reduces to 1, but the time taken to convert either an analog signal to corresponding digital equivalent is quite large.

And to reduce the time we discussed continuous tracking type of ADC, using both up counter and down counter. Then we moved on and we discuss accessing the approximation type ADC which is very popular and it has its advantages specially when multiple inputs are to be converted one after another, in the continuous tracking type the tracking gets lost the locking get lost once you move from one input to another. We also discussed dual trace ADC which within it gives us integrator for which we have an advantage of that ADC providing better noise immunity.

And further we discussed delta sigma ADC which is good for slow changing signals and it uses over sampling of the input and it can give very high resolution I mean the number of bits that we can get in a to d conversion is quite large. And we also discussed some of the performance metrics accuracy resolution and there are errors involves both from digital type and analog type and that is how we concluded week 11 ok.

(Refer Slide Time: 03:35)



Types of Memory

- Circuits and/or systems designed specifically for data storage are referred to as memory.
- Memory may be flip-flop, register, semiconductor memory chips (also magnetic, optical).
- RAM (Random Access Memory) is volatile, meant for multiple read and write (Read/Write Memory).
- ROM (Read Only Memory) is non-volatile, meant for multiple reading, also random access.

If m address lines, memory size $2^m \times n$

Random access: Independent of the physical position of the memory within memory block.

Bit organized: 1-bit stored in one address
Word organized: Group of bits stored in one address

There are further subdivisions of each memory type.

Moving on we begin with the types of memory that we encounter in digital electronics or digital circuits. And here we are talking about some specially designed data storage units as memory and we have seen flip flop register they are being used as memory and in this particular week we shall look more into semiconductor memory chips.

Specially designed semiconductor memory chips and also there are memory elements which is coming in the form of you know magnetic you know disk ok, optical disk that we have used we are using in computers, but that is outside the purview of these particular discussion. And we make use of them in the when we connect them as a peripheral to a computing platform.

So, in semiconductor memory chips we have mainly two kinds of them, one is called RAM; Random Access Memory; this is volatile means if power goes on the value gets lost and it is easy to write and it easy to read. So, multiple read and write operation is possible and if that for which it is also called read write memory ok. Contrast to that we have got another type of memory called ROM, Read Only Memory, which is non-volatile and it is made for multiple reading ok.

So, that what does it mean? I mean it has to be written once with some effort. So, it requires some special efforts special mechanism right. So, from the user point of view the final user end user will be able to read it many times ok, but the developers from developers point of view when it is required it will be able to the developer will be able to write it by some special mechanism. And to be noted that though this called ROM it is also a random access memory by nature ok.

So, random access means the access of the information the storage stored data digital information, it is independent of physical position of the memory within the memory block ok. It does not really matter where it is there, but if it is a sequential axis then it has to move you know sequentially from one place to another. So, that will take if it is further then it will take more time like if it is in a tape.

So, the one which is closer it will you can fetch it quickly, the a one that is further from the head which is reading the tape then it will take more time to reach that particular location ok. And of course, within this RAM and ROM there are further subdivisions like in RAM that is called Static RAM, which does not require refreshing that is dynamic RAM which requires periodic replacing each one has it is you know strengths and weaknesses. Similarly in ROM we have got prom we have got e prom all those varieties we shall take up in the subsequent classes so, these are the two major variety.

And other than that the memory of course, we understand that in a memory block there will be an address lines ok. So this what you see over here? This is a RAM representative RAM block. So, address lines so, this indicates a location in the memory ok. So, if there is a 2 bit address line say a naught and a 1 then 2 to the power 2 4 such places in the memory can be addressed right and in that you can write data right.

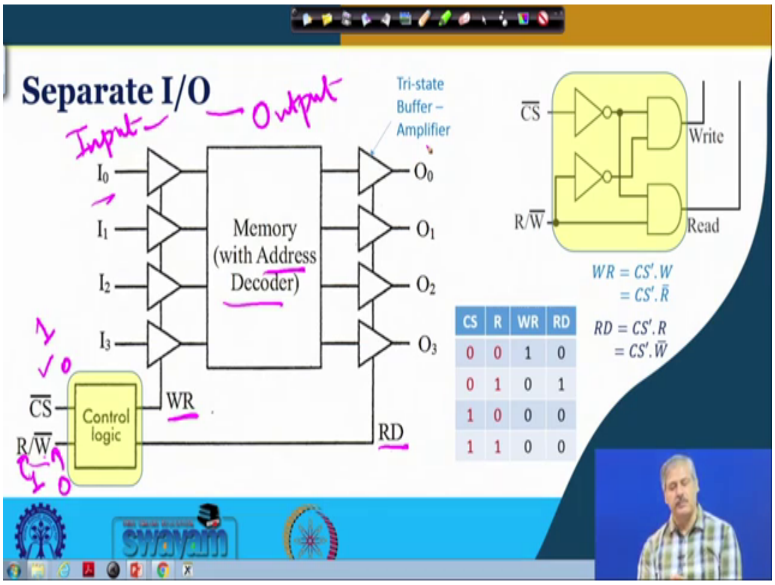
So, if the data is of n bit it could be 1 bit data, it could be 4 bit data, it could be 8 bit data and all. So, this is the data in and this is the data out that is as we said random access memory read write is possible at the user end. And then we have the other kind of you know control inputs in the form of read write enable. So, sometimes these three or you know control inputs can be clubbed into two and all and subsequent output can be generated from those two inputs. So, those schemes we shall see little later.

So, enable means this particular RAM block is being accessed and for which this is the corresponding address what you see over here and then when you say read operation so, you are inhibiting right. So, you are reading whatever is already stored and when you are saying right, then you are over writing the current input so, you are writing a new information ok. So, the old information is lost so, then you can again read by invoking read. So, these are the standard you know read, write and enable these control inputs that you will see right and when you take it to wrong.

So, basically in this particular case we will see that there is a enable whenever it is enabled. So, you know it is the reading operation only that you are doing and from the developer side something is has already been written into it which we can read multiple times. And similarly where from you are reading that is defined by this address lines and memory can be another subdivision could be there bit organized or word organize. So, in a particular location address location if only one bit information is there; if that is the way memory is organized then it is called bit organized and in that location number of bits are there ok.

So, a group of bits are there then it is called word organized and it is the size of the word depending on which you will be having 4 bit, 8 bit, 16 bit, 32 bit or so on and so forth depending on your requirement, is it clear. So, these are the way we broadly define memory and then we shall see further subdivisions going forward ok.

(Refer Slide Time: 10:10)



Separate I/O

CS	R	WR	RD
0	0	1	0
0	1	0	1
1	0	0	0
1	1	0	0

$WR = CS'.W$
$= CS'.\bar{R}$

$RD = CS'.R$
$= CS'.\overline{W}$

Now, as I said the control mechanisms we shall see right. So, here we see a particular memory block is a very simple memory block right, which is having a control you know input CS and read stroke write.

So, R for read and W for write; write is you know compliment. So, basically if it is 1 then read operation will take place and if it is 0 then write operation will take place ok. So, then 2 inputs is clubbed into 1 and CS means chip, select bar means if it is 0, this chip is selected and if it is 1 then this chip is not selected, is it clear. And this is the memory unit and there are physical memory cells more about that we shall study later right and that particular cell is addressed by some address bits for which certain address decoder is there so, these we shall take up later.

So, once this chip is selected and we know that the address lines are pointing to a particular location of the physical location of the memory right, then two operations are possible right one is read operation, another is write operation.

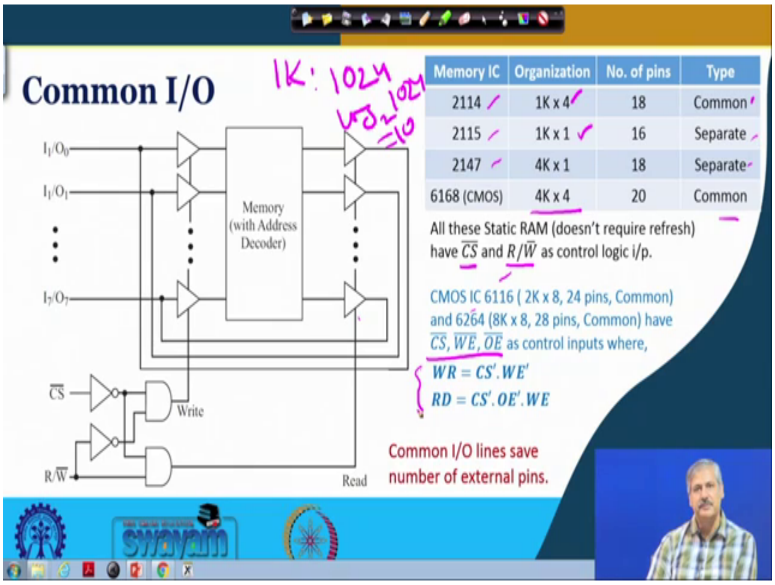
So, in this particular case what we are showing that for this memory block we have got separate input and output. So, this I stands for input ok, in bit we often say that it is I O; I means input and O means output right. So, then this control logic is generating a write signal, write or a read signal. So, whenever it is so, this is a tri stated buffer. So, if it is not invoked so, if it is 0 ok. So, then this output will be high impedance. So, basically these input is not going to the output. So, it does not get related is it fine? And whenever it is 1 then this I naught goes to a specific its location I 1 it is location, I 2 it is location and I 3 corresponding location is it fine.

So, these are separate this 4 you know input lines are there and for our read operation so, at that time definitely write should not be there then there will be you know mix up all right. So, at the time write will not be happening. So, whatever is already stored the memory address is in you know this invoke the location is invoked chip select is enabled then whatever is there. So, this will be 1 and then it will go to the output, otherwise it will be remaining tri stated high impedance state.

So, this is one a simple organization right with separate input and output and how you can generate you know this logic this control logic? We can see in that what we are looking at. So, if CS is 0 ok, then if R is 0 ok, R is 0 means you are not doing the reading operation you are doing the write operation. So, WR should be 1 and RD is equal to 0 right and chip select it is enabled 0 right, these R is 1; that means, now you are doing the reading operation so, WR should be 0 and RD should be 1 ok.

And when chip select is 1 right so, it is not selected right, this memory chip this block is not selected then whether it is 0 or 1, read and write should be 0, is it fine. So, if you then convert it to corresponding Boolean expression. So, this is the expression you get right. So, W and R body is you know the same you know so, we can use one of them and then if you translate it to a corresponding digital logic block this is what you get very simple is it fine, CS prime and this is going as W, that is R bar and this is CS prime R.

(Refer Slide Time: 14:34)



Common I/O

Memory IC	Organization	No. of pins	Type
2114	1K x 4	18	Common
2115	1K x 1	16	Separate
2147	4K x 1	18	Separate
6168 (CMOS)	4K x 4	20	Common

All these Static RAM (doesn't require refresh) have $\overline{CS}$ and $R/\overline{W}$ as control logic i/p.

CMOS IC 6116 (2K x 8, 24 pins, Common) and 6264 (8K x 8, 28 pins, Common) have $\overline{CS}, \overline{WE}, \overline{OE}$ as control inputs where,

$$WR = CS'.WE'$$
$$RD = CS'.OE'.WE$$

Common I/O lines save number of external pins.

Now, we can see memory blocks memory IC's in which these input and output are not separate if they could be common also ok. So, these comes from the fact that when you are doing a reading operation; you are not doing write operation and when you are writing you are not doing the read operation; isn't it? So, that gives us an opportunity to have a common input and output line ok.

So, this is one such arrangement what you can see over here right this is the you know input output is common. So, we can see earlier in the previous arrangement this was going out and this is a separate output that was there right, this is what we had seen. Now what we are doing? Since the when this is used so, these input blocks are not used right. So, what we are doing over here we are just connecting it to the I O line over here right and these control logic remains the same, for which when you are doing the right operation this is 0 so, it is try stated. So, this is very high impedance so, it is not affecting the input data that is getting written in anyway ok.

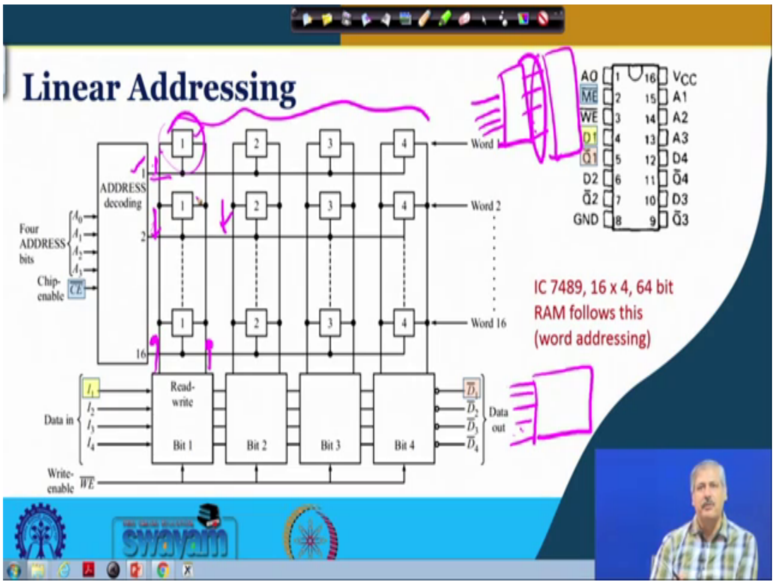
So, this is what we can use in the common I O configuration is it fine right. So, we have memory IC s so, some examples I can show you that 2114, 2115, 2147 these are all BJT based memory ICs right. And they have got these either common or separate this kind of arrangement D 1 that we have just discussed ok. And these organization 1K cross 4 what does it mean? That 1000 such locations are there right, physical locations are there. And to invoke this 1000 location so, 1000 1K we know in digital logic or digital discussion binary arithmetic so, 1K stands for 1024. So, how many address bit will be required? So log 1024 to the base of 2.

So, 10 address bits will be required. So, this particular IC will be having 10 you know address bit to point to a particular location of the memory and in that how many in each location how many bits are stored? So, 4 bits are stored so, it is word organized right and these are these input and output this writing is this is common. And all these are using two control logic CS and R read stroke ah, write bar the one that we had discussed just now right.

Now, what about this one 2115 this is bit organized again 1K, but in each location you are having only 1 bit that is stored right. And in this case you have got separate you know input and output you know likes right and 6168 this is CMOS based IC right memory IC. So, we shall see both you know BJT based you know memory cell and CMOS based memory cell little later I mean in subsequent classes right. So, this is 4K cross 4 and it also has common and there are other memory ICs in which the control inputs as I was discussing we have got instead of 2; 3 control inputs chip select write enable for writing purpose, output enable this is for reading purpose.

So, these kind of three kind of you know control inputs are there. So, this is one such examples this is another example these are CMOS based IC and these are the corresponding size of the memory right and this is the way the logics are generated in those cases fine. So, we move on.

(Refer Slide Time: 18:54)



Linear Addressing

IC 7489, 16 x 4, 64 bit RAM follows this (word addressing)

Now, this addressing memory; so just now what we had discussed if you try to visualize how it looks like. So, in you know memory block so, this is one such example.

So, here this is the address decoder to which in these particular example 4 address bits are coming and that is getting decoded. So, you have got 16 lines generated. So, these are 16 lines so, 1, 2 to 16. So, if you have looking at this memory block so, this is the memory block right. So, these are 16 lines that are generated this is coming from the row decoder ok; so there are 1 2 3 4.

So, these are the decoding decoded output and then this is the memory. So, 16 such lines to the memory after the decoder and from each location right each location you are having four such bits right. So, these are the four cells that you can see more about this we shall discuss in the next class how these cells are formed and all how they are you know. So, this is connected right. So, when you are addressing this one this particular line of course, the others ones are not addressed.

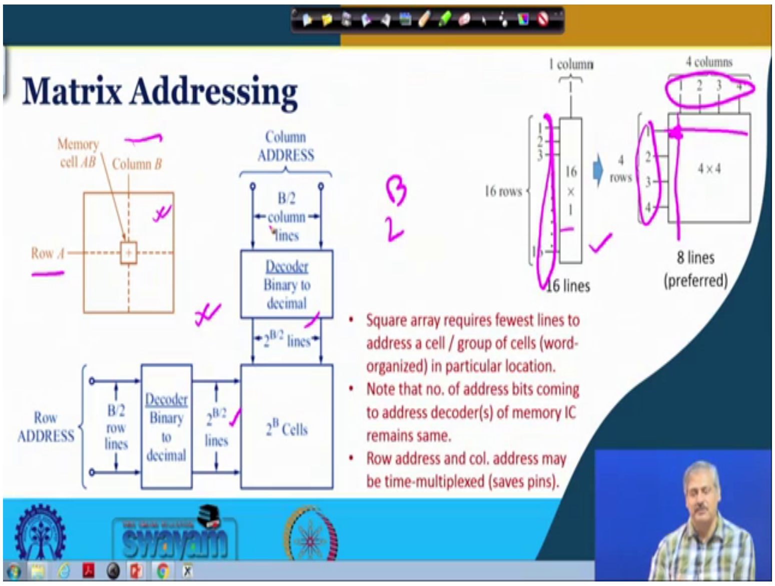
So, whatever is information is there in this cell how why I mean this is one side this is another side both are coming over here. So, this organization we shall see little later and then we are reading it during the reading when we are doing the read operation and when you are trying to write it. So, information will go from here to here and change the value of the cell and during the reading operation information will come from here to here if it is the first location and then other locations are tri stated so, they are not contributing ok.

And similarly the next location is addressed so, this one. So, information will come from here for read operation and we will go to it during write operation and otherwise it will remain tri stated ok. So, this is for bit 1, bit 2, bit 3, bit 4. So, this is word you know organized right and this is the data coming in and this is the data going out and we have some this particular thing a practical IC, IC 7489; if possible you can work with it in the lab.

So, this is a 16 cross 4 so, 16 such locations and 4 means 4 bits in each location 64 bit RAM ok. And so, this CE that you see over here chip enable or chip select in our earlier discussion. So, in the IC data sheet if you see you can see that it is written as memory enable ok. So, write enable, write enable it is there which is inverse of read that we have already seen, the other thing that you see the data in I 1, I 2, I 3, I 4 in this particular diagram the corresponding thing a thing in the datasheet is D 1, D 2, D 3 D 4.

And this is the corresponding output final output in this diagram D 1 bar, D 2 bar, etcetera, etcetera, which corresponds to Q 1 bar, Q 2 bar and then you can do this mapping fine. And this kind of addressing where the address bits are coming and one after another; these row decoding I mean through a row decoder the locations are being accessed; so this is also known as linear addressing ok and in contrast to that we shall see another kind of addressing which is called matrix addressing ok.

(Refer Slide Time: 22:38)



Matrix Addressing

- Square array requires fewest lines to address a cell / group of cells (word-organized) in particular location.
- Note that no. of address bits coming to address decoder(s) of memory IC remains same.
- Row address and col. address may be time-multiplexed (saves pins).

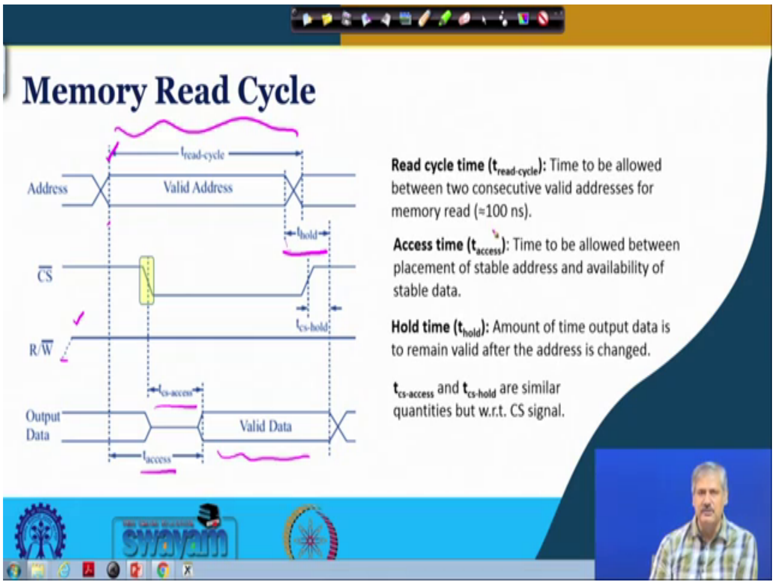
So, what we do there? So, in this a particular location is accessed particular location is accessed through a row decoder and a column decoder ok. So, how does it really work? So, let us look at one example first and then we shall go come to this particular block. So, this is 16 lines we have already seen right see that 7 4 8 9 example the previous in the previous slide. So, these are the 16 lines that are going to the memory after the row decoder. So, this is the memory say block so, say it is a 16 cross 1, it could be 16 cross 4 also ok.

So, this is there whereas, in this case what you see? If you are having in each location say one such thing, one such bit you are trying to address. So, you can have 4 rows here and 4 rows there and at the cross point; at the cross point over here you can store 1 bit right. So, it is a 2 dimensional plane now and if you understood 4 bit. So, then it will be coming in the z direction, if this is x and this is y, then it is in the z direction. So, 4 such cells will be placed, is it clear or if it is just one bit then it is over here. So, how many such things will be required so, this is 4 line and this is 4 line.

So, 8 lines to the memory instead of 16 lines going to the memory ok. So, this is the usefulness of matrix based addressing and most of the you know advanced to memory chips you know you will see that it is using matrix based addressing. So, generally if you want to generalize it so, this is what I was talking about these you know cross point will be having the memory cell, to generalized we see if it is a square arrangement I mean if you equally divide then the number of rows will be the number of lines going to the memory will be the list ok.

So, if we have you know this 2 to the power B lines in total. So, half will be here half will be there, but note that the number of this address will always be the going to the chip will be of course, this you know B by 2 here and B by 2 there total B that is what will be required that we cannot avoid of course, ok, but the way it is going to the memory physical block we can you know make you know different kind of arrangements over there.

(Refer Slide Time: 25:32)



Memory Read Cycle

Read cycle time ($t_{read\text{-}cycle}$): Time to be allowed between two consecutive valid addresses for memory read (≈100 ns).

Access time (t_{access}): Time to be allowed between placement of stable address and availability of stable data.

Hold time (t_{hold}): Amount of time output data is to remain valid after the address is changed.

$t_{cs\text{-}access}$ and $t_{cs\text{-}hold}$ are similar quantities but w.r.t. CS signal.

Now, we would look at, we have seen certain control inputs we have understood that there is a memory cell in which it is getting something data is getting written or we are trying to read from it. So, how one memory read cycle will look like ok. So, if you talk about these timing diagram based you know understanding then we look at one such cycle over here.

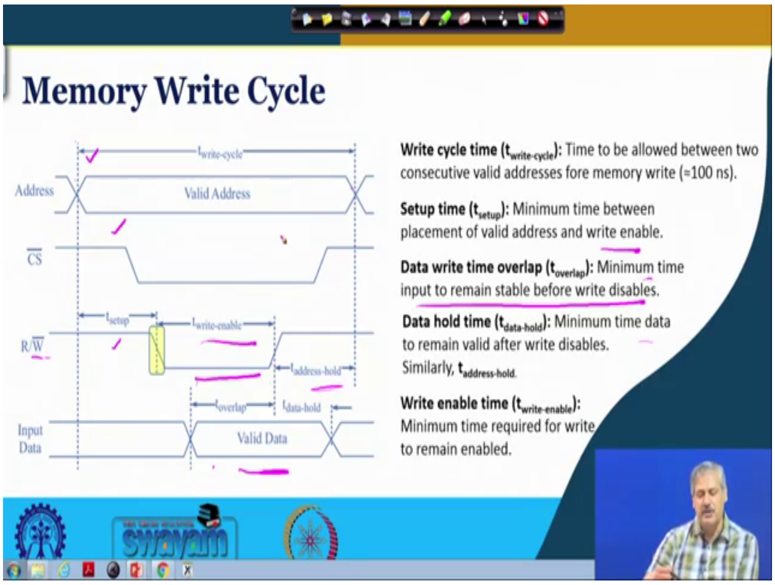
So, here this is high means we are trying to so, this is low, this is high; so this is we are trying to do a read operation. And then we have floating the a memory address a valid address, the physical location from which it has to be you know rate right and then chip is chip select is enabled chip select high means it is disabled. So, it is going low means it is enabled and then you see the valid data that we are reading it is coming after some point of time ok.

So, these are certain delays associated with you know this memory read operation ok. So, from address line being made valid to a valid data comes the time need to be provided. So, that is known as access time is a very important parameter of you know into you know data sheet and other things will be able to see that some such you know value is provided. And this some chip select to valid data, so this is called chip select access and the other important parameter is this whole time right so, that is when the valid address has been removed right and but output remains valid.

So, even after the address is changed so, this is the time that you see this is the hold time and in comparison if it is done with respect to chip select then it is called chip select hold time. And this is one memory read cycle after that another address can be floated and the minimum time that is required that defines the speed of how this memory read operation and typically it is of the order of 100 nanosecond.

So, each of these parameters if you look at the data sheet will be around say 13 nanosecond or you know different values for different such things ok. But altogether if you just look at it then it comes around 800 nanosecond and you know faster memory means smaller this value slower means memory means larger these value is larger.

(Refer Slide Time: 28:28)



Memory Write Cycle

Write cycle time ($t_{write-cycle}$): Time to be allowed between two consecutive valid addresses fore memory write (≈100 ns).

Setup time (t_{setup}): Minimum time between placement of valid address and write enable.

Data write time overlap ($t_{overlap}$): Minimum time input to remain stable before write disables.

Data hold time ($t_{data-hold}$): Minimum time data to remain valid after write disables. Similarly, $t_{address-hold}$.

Write enable time ($t_{write-enable}$): Minimum time required for write to remain enabled.

So, if that is a memory read cycle we will be having a corresponding memory write cycle ok. So, how memory write cycle you know; what are the sequence of events taking place. So, what you see that, first address is made you know a valid address is put on which where the writing needs to be done ok, then that particular chip is selected because that address could can belong to some other you know chip also ok.

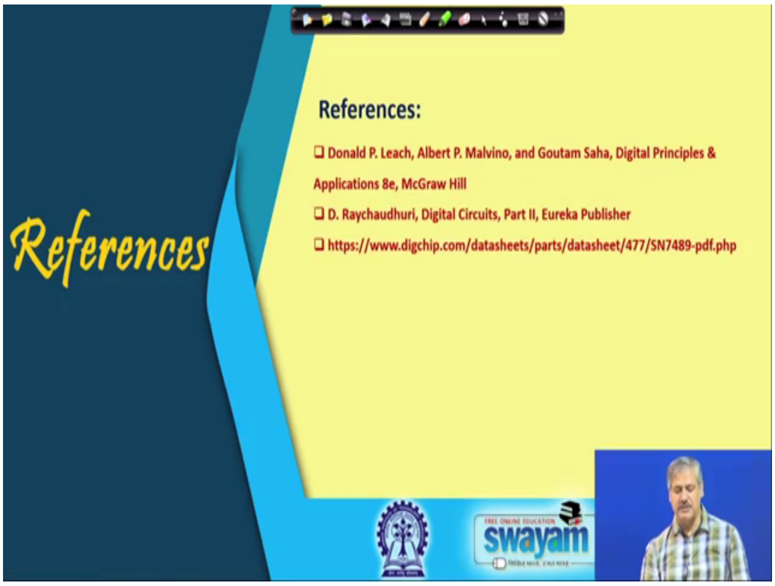
So, whether it is meant for that specific chip or not and then the write operation. So, this particular control input goes low; that means, it is write is to be done writing is to be done and after some time a valid data becomes available. And then this write is disabled then this chip select is disabled and valid data becomes unavailable after some time.

So, this is the valid data where these actually you are reading the information from the memory location right, similar to read cycle we have some of these important parameters right. So, the whole of it from one address to next valid address that is the write cycle time the minimum time that is required again it is of the order of that 100 nanosecond typically. And the setup time is minimum time between placement of this valid address and the write enable so, this is your set up time right.

So, these are some other important parameters. So, this is the time in which write enable need to remain valid and data write time overlap. So, this is the minimum time input to remain stable before right disables ok. So, this is the minimum time required for this after this valid data is available it is need to be remain stable. So, these this write enable need to remain stable and then comes data hold time. So, data hold time is the minimum time data to remain valid after write disables. And similarly this address hold time these write disables, but after this much of time this address should remain valid ok.

So, these are some important timing parameters in association with memory read and write operation ok.

(Refer Slide Time: 31:06)



References

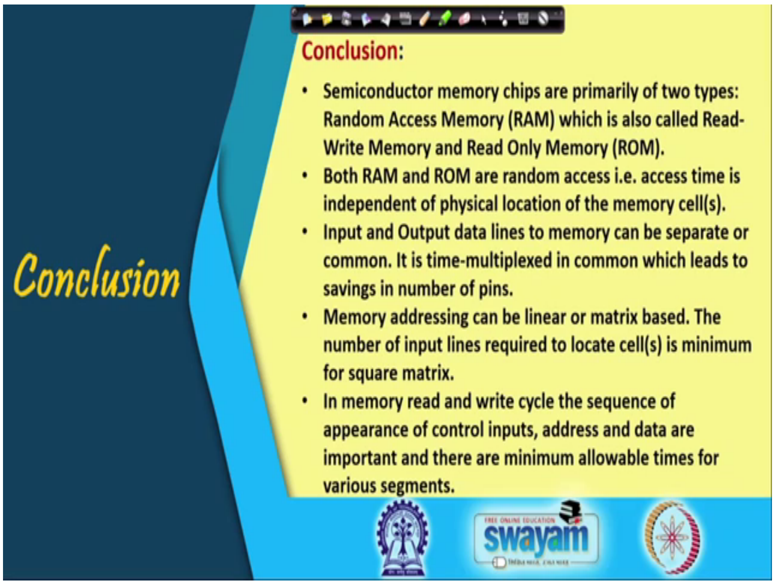
References:

❑ Donald P. Leach, Albert P. Malvino, and Goutam Saha, Digital Principles & Applications 8e, McGraw Hill

❑ D. Raychaudhuri, Digital Circuits, Part II, Eureka Publisher

❑ https://www.digchip.com/datasheets/parts/datasheet/477/SN7489-pdf.php

(Refer Slide Time: 31:07)



Conclusion

Conclusion:

- Semiconductor memory chips are primarily of two types: Random Access Memory (RAM) which is also called Read-Write Memory and Read Only Memory (ROM).
- Both RAM and ROM are random access i.e. access time is independent of physical location of the memory cell(s).
- Input and Output data lines to memory can be separate or common. It is time-multiplexed in common which leads to savings in number of pins.
- Memory addressing can be linear or matrix based. The number of input lines required to locate cell(s) is minimum for square matrix.
- In memory read and write cycle the sequence of appearance of control inputs, address and data are important and there are minimum allowable times for various segments.

So, with this we conclude the introduction to memory and what we have seen very briefly that semiconductor memory chips are primarily of two types, RAM and ROM both are actually random axis, where the access time is independent of physical location of the memory cells and input output data lines to memory can be separate or common ok.

And memory addressing can be linear or matrix based and of course, for matrix base the number of lines input lines to locate a cell can be minimum and it is minimum for square matrix ok. And in memory read and write cycle the sequence of appearance of control inputs address and data are important and there are minimum allowable times for various segments and as a whole that gives us an estimate of how quickly we can read or write into memory ok.

Thank you.