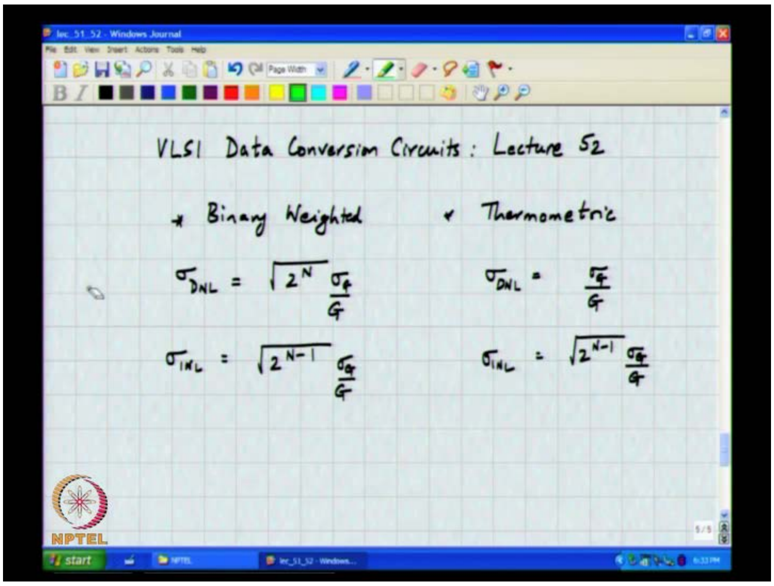
VLSI Data Conversion Circuits
Prof. Shanthi Pavan
Department of Electrical Engineering
Indian Institute of Technology, Madras

Lecture - 52
Segmented DACs

(Refer Slide Time: 00:17)



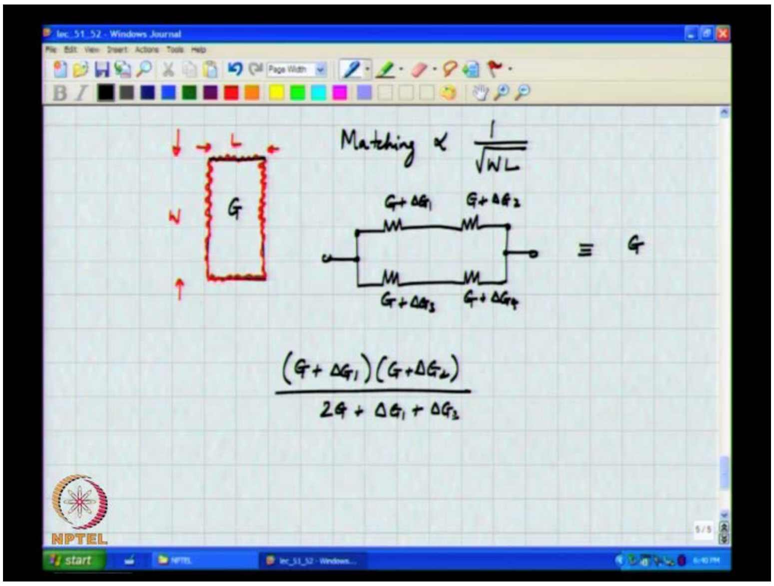
So, this is VLSI data conversion circuits lecture fifty two; in the last class we were discussing the relative merits of binary weighted and thermometric DACs. The sigma DNL in the binary weighted case was approximately square root of 2 to the N times sigma G by G whereas, in the thermometric case was square root of 1 and sigma G by G; whereas, the sigma INL is the same in both cases. The upper result makes intuitive sense because in the binary weighted case the worst case DNL.

Please note these are all worst case numbers in the binary weighted case the worst case occurs at the major code transition and it is during the major code transition that you are getting the next step by removing or disabling whole a bunch of current sources. I mean in this case conductances and enabling another set of conductances, which is only marginally greater than the one that you have disabled. So, on a more common sensical level it is trying to get a very small number by subtracting two large numbers each of which is uncertain, which means that the result which is the difference is very uncertain because the uncertainty of both the random numbers both the big and the small.

I mean both the big numbers adds giving you a lot of variance. On the other hand the INL is simply measuring the deviation of the code from the ideal code and that if you wanted a code of m it means that whether you do it binaurally or through a thermometer m current sources must be m conductance's must be active, which means that the INL must be the same in both cases. Because you are basically measuring the deviation of m conductance from m times G and whether that is binary or thermometer remains the same t.

So, which means that as a designer if you want to hit a certain target DNL a certain worst case DNL, then if you implement it as a pure binary weighted system you will need to have a much tighter matching requirement on the individual elements. I mean clearly to get a certain sigma DNL for the binary weighted case the sigma must be square root 2 to the N times smaller than what it would be otherwise when compared to the thermometric case. Now one question that one might ask is.

(Refer Slide Time: 04:10)

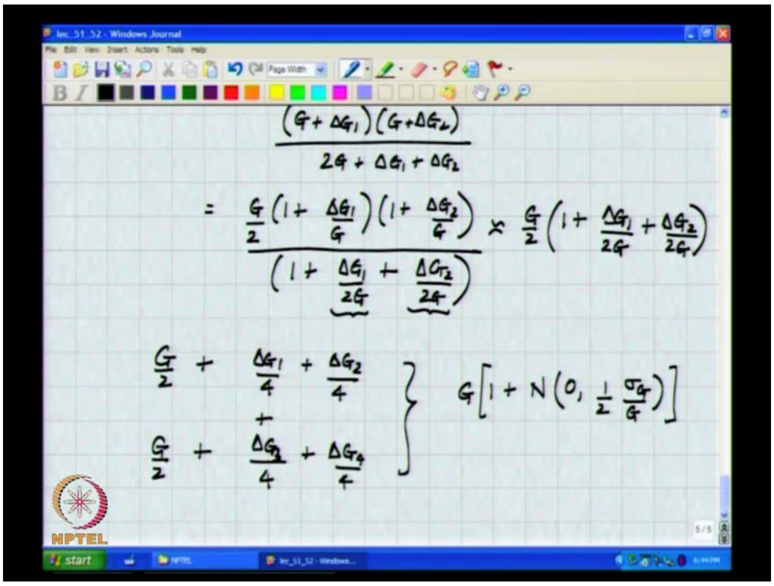


Let us say I implemented a resistor or a conductance and its nominal conductance is G. Now, there are variations due to random mismatch and the factors that cause mismatch are the following, one in a resistor what happens is that these edges are not sharp and that is a one source of variation. Then there will be some tolerance to the way you can define the length of the resistor and there will be some tolerance to W. So, W and L being random variables and there is an edge roughness and so on. All these things basically when you put them altogether you can model the whole thing as some Gaussian random variable with some variance.

Now if you want to get a conductance G, but with improved matching what do you think you can do? Increase W and L and why does it make sense. Relative variance should be through the absolute errors you mean the same relatively. So you can think about in couple of ways one argument you can make is that the delta l and the delta W probably remain the same, but if you make one much larger the delta l becomes a much smaller fraction of L. Therefore, you can expect improved matching and that is generally correct. So, the matching is dependent on 1 over square root WL the sigma G by G will be proportional to 1 by square root WL and this is not just noticed with resistors it is also notice with capacitors and transistors.

One more way of looking at it is to say that if I took four identical resistors conductances and connected them up like this G G G and G what is the conductance now? This as this is equivalent to single conductance G; however, what can you say about there the variance of this composite conductor conductance each of these as say this is delta G 1 this is delta G 2 delta G 3 and delta G 4; conductance of the top branch is what G plus delta G 1 into G plus delta G 2 divided by 2 G plus delta G 1 plus delta G 2.

(Refer Slide Time: 07:32)



That is where I made a mistake correct which is 1 which is G times 1 plus delta G 1 by G times 1 plus delta G 2 by G divided by 1 plus delta G 1 plus by 2 G plus delta G 2 by 2. I am sorry I made a mistake this is G by 2 times approximately 1 plus delta G 1 by 2 plus delta G 2 by 2 G these are all small numbers. So, I can pull them up into the numerator and then subtract and so on. So, the top branch is a conductance which is G by 2 plus delta G 1 by 4 plus delta G 2 by 4. Similarly, the lower branch has a conductance G by 2 plus delta G 2 G 3 by 4 plus delta G 4 by 4 and when you add the 2 you get a conductance G plus a random variable, which is 4 and 16. So, it is G plus N of 0 sigma G by 2 by G times 1 half G times 1 plus.

So, variances add 16 and we have 4 them. So, it is one-fourth standard deviation must become square root of one-fourth, which is one half. So, we have a conductance which is 4 times larger and it is variance the matching is improved by a factor of the sigma has gone down by a factor of 2. So, in another words if you want increased matching you must be prepared to have a physically large element whether it is a transistor or whether it is a capacitor or whether it is a conductance improved matching means larger area. So, while I have discussed this binary versus thermometer decoding in the context of a resistive DAC the same is true for whether you use capacitors where then the quantity of interest is not I DAC, but charge Q DAC.

So, you have capacitors which are charged to some voltage and then you somehow take those take many of those capacitors and then add up or extract the charge from all these capacitors; just like how we are passing current though these resistors and extracting the current it is also possible to do that with charge all right and again just like how matching depends on a matching of the resistors, there it will depend on matching between capacitors and the same arguments for area and matching will hold and the same arguments for the LSB change or rather the DNL of the DAC will hold whether we implement a completely thermometer type design or a completely binary weighted design.

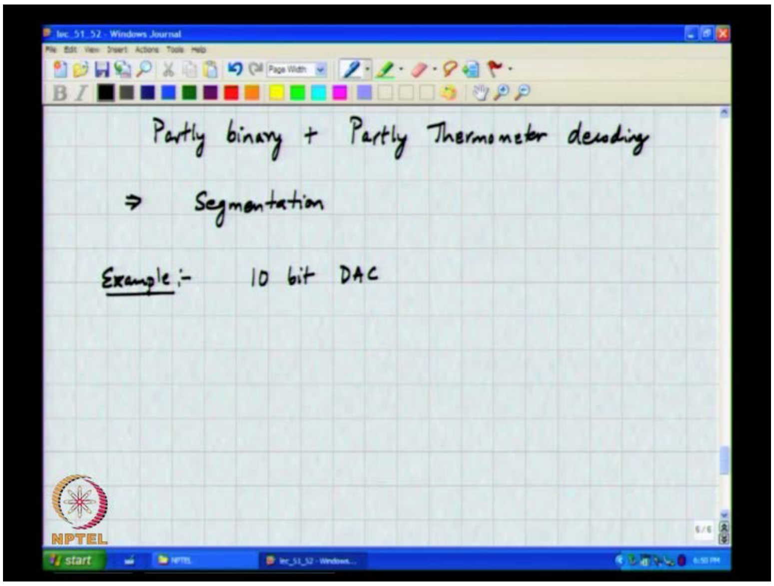
Since we have seen the two extremes which are the thermometer code where the DNL is very nice, but the INL, I mean the INL is also there is no difference between the thermometers or and the binary designs as far as INL is concerned. However, decoder is a lot less complex in a binary design, but you get a lousy DNL. So, in the forbid example that we have seen the decoder may not seem like a big deal, because it is only a four to fifteen decoders. Now, imagine you want to design at twelve bit DAC then the decoder complexity twelve lines coming out and four thousand ninety six wires getting out which makes routing and everything else a terrible nightmare.

So, one obvious thing is to say is there some middle path, where I do not have to deal with the enormous complexity of routing 2 to the N wires, which is what I should do in a fully thermometer design. And be the terrible DNL level I will get if I use the same kind of elements in the in a fully binary weighted design, where I have gotten rid of the decoder, but the DNL be terrible. So, in order to make the DNL come within speck I will have to chose elements with lot better matching, which means that the elements have to become much bigger as we just saw, which means that the design will become occupy a lot of space.

You understand and you know to add to this we are trying to meet the DNL specs by increasing the matching of the design. Unfortunately, the INL spec is way over design, because INL is only proportional to I mean as we saw before thermometer or binary. The INL worst case expression is the same. So, it is likely that if we go and design for good DNL in the binary case the sigma G by G will have to be chosen to be so low that the INL may be way over design.

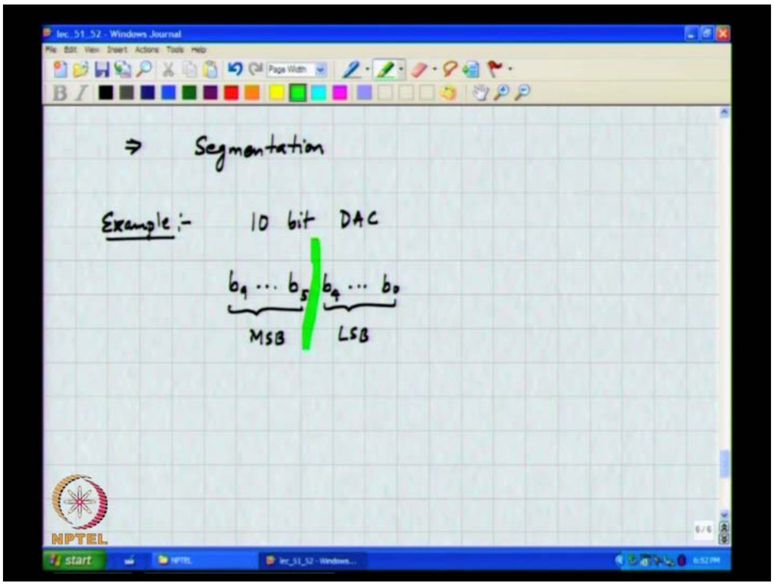
You may not really require such a good INL at all in the system. I mean the penalty of paying for easy decoding is this huge area as far as of the improved matching requirements of the elements. Now the question is can we find a middle path where you have neither the enormous decoding problems associated with a fully thermometer design not the terrible DNL associated with a binary design.

(Refer Slide Time: 17:06)



I mean as you guessed it is you do a partly binary plus partly thermometer decoding and this kind of that design is called segmentation a segmented DACs. So, as an example let us take a ten bit DAC and say I am going to have I am going to segmented such that half of it is binary and half of it is thermometer.

(Refer Slide Time: 17:44)



So, let us say b 9 through b 5 or rather b 5 will be decoded one way and b 4 through b 0 decoded in another way. So, do you think it makes sense to decode the LSB’s in a binary manner or in a thermometer manner? In other words should we decode the MSB’s in thermometer and LSB's in binary or vice versa?

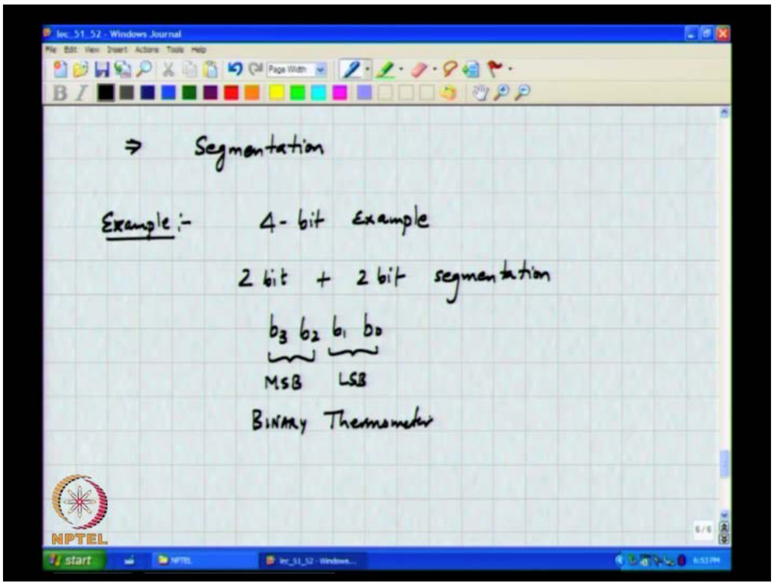
Student: If MSB is binary,

Why?

Student: It requires lot of ways.

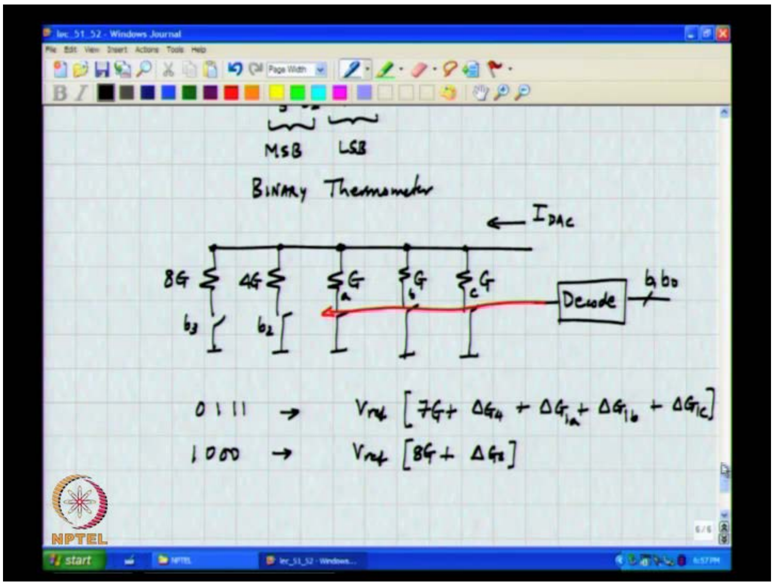
No, if I say that I am going to split the DAC into 2 parts: one I am going to decode 5 bits binary and 5 bits thermometer from a decoding point of view it does not seem to make any difference right if you decode one binary, the other will be thermometer and so on may be the ten bits is confusing you. So, let me go back to the four bit example. So, I could conceive of at least two ways of segmentation let me choose a 2 bit plus 2 bit segmentation. So, b 3 b 2 b 1 b 0 I am going to say I am going to as summary points out I am going to say let me try and do this binary and the LSB's I am going to do as thermometer. Let us see what happens?

(Refer Slide Time: 20:55)



So, in other words the DAC will look like this. So, I will continue to have what should I have if b 3 and b 2 are decoded in a binary fashion I should have 8 G and 4 G. So, this becomes gets controlled by b 3 this gets controlled by b 2 and what about b 1 and b 0 I will have 3. Three individual conductances all of value G and all I need to do is have a decoder which takes b 1 b 0 and goes and controls these switches and this is I DAC. So, what is happening with the biggest problem with the DNL is?

(Refer Slide Time: 22:01)

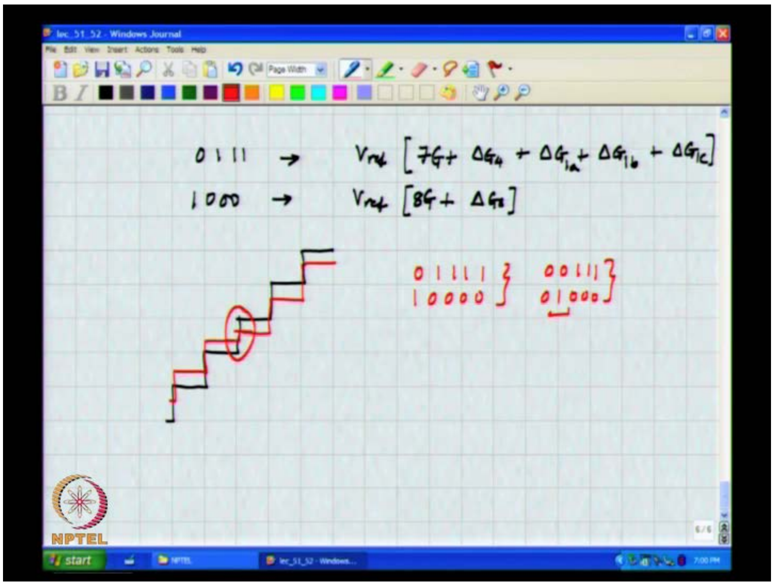


I mean this resistor mismatches and at what code transition is the problem the worst, zero followed by all ones to one followed by all zeroes. So, has anything changed when it is zero followed by all ones the current will be G times V ref times 7 plus delta G 4 plus delta G 2. I mean delta G plus delta G 1 a 1 b 1 c where I call this a b and c does it make sense. Now if it is one followed by all zeroes what is the output code, output value corresponding to this code G times V ref. 8 plus delta G all this must be or I should get rid of delta G 8. So, what is the variance of the step size?

Student: Sir, it is fourteen.

It is fifteen, please note that fundamentally should not make any difference because; however, we getting the step we are turning off all these characters and turning on the 8 G, which is exactly the same thing that we were doing before you understand. So, clearly making the MSB's get I mean decoding them in a binary fashion is not the right thing to do; I mean intuitively this is what is happening in the ideal DAC outputs must be like this.

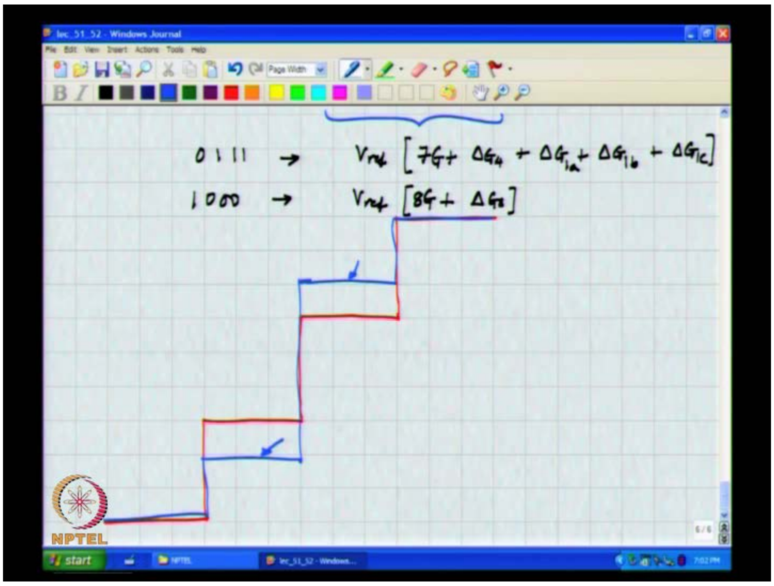
(Refer Slide Time: 25:01)



Where I just draw in the horizontal lines just to make sure that you know we see something there rather than just dots now with a binary weighted DAC. If this particular transition was the major code transition this will be main error. So, this step for instance can become like this. This is the major code transition and you will have large DNL's at not just I mean the largest DNL will be at the major code transition, but every you know sub major code transition if you will, whenever any of the lower bits I mean when whenever there is a change from zeroes followed by all ones to ones followed by all zeroes.

For example, there will be a peak in the DNL at zero followed by all ones and one followed by all zeroes, but there also be peaks local peaks in the DNL when you get 0 1 1 1 to 1 0 0 0 this the 0 variance of this peak will be smaller right, but there will still be a peak you understand. So, in another words the use of the binary decoder will make large amounts of because large the large amounts of DNL and does not help if the LSB's are decoded in a thermometric fashion does it make sense. So, another way of understanding this is b 3 b 2 is defines how many bit DAC is a 2 bit DAC with a large step size of 4 G times V ref.

(Refer Slide Time: 27:47)



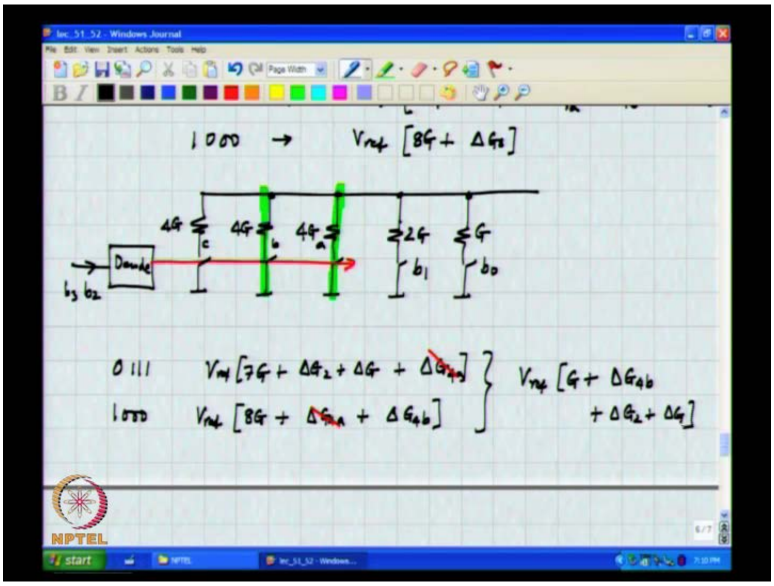
Now, if this staircase which is ideally supposed to be like this is made with a binary decoding, then what will happen is that this step will become very large or very small depending on the deviation of the individual elements. Within this step within this large step the small guys at jumping helping it jump from one large step to the other. Decoding them in a thermometer fashion will ensure that that small staircase which is joining which is helping you go from this step to this.

That staircase will be very uniform, but the big staircase has got a lot of variation at the major code transition right you understand and you can see that the variance is no better than the fully binary case. So, in another words when you want to segment therefore, you must segment the MSB's thermometric decoded which means that you can always think of a DAC as the DAC output has coming from two DACS, one which is decoding the MSB's and one which is decoding the L S B's.

You add the two up together that is you know mathematically that is perfectly valid point. Now the LSB size of the MSB DAC the step size of the MSB DAC is much larger than the step size of the LSB DAC. So, if you want to make an overall nice staircase would you want to make a good staircase for you know MSB DAC or for the LSB DAC. You want the big staircase to be as linear as possible and the small DAC the DAC which is decoding the LSB's is just creating another staircase between two levels of the MSD DAC.

So, if there is a jump in the DNL in the small staircase that jump is likely to be small because the staircase is itself is very small whereas, for the MSB the DAC decoding the most significant bits. So, this height of the staircase is large you want to make sure that that is a lot more uniform than the LSB. Ideally, you want to make everything you know uniform, but that as we saw will lead to a complete thermometer implementation, which makes the decoder a nightmare you understand.

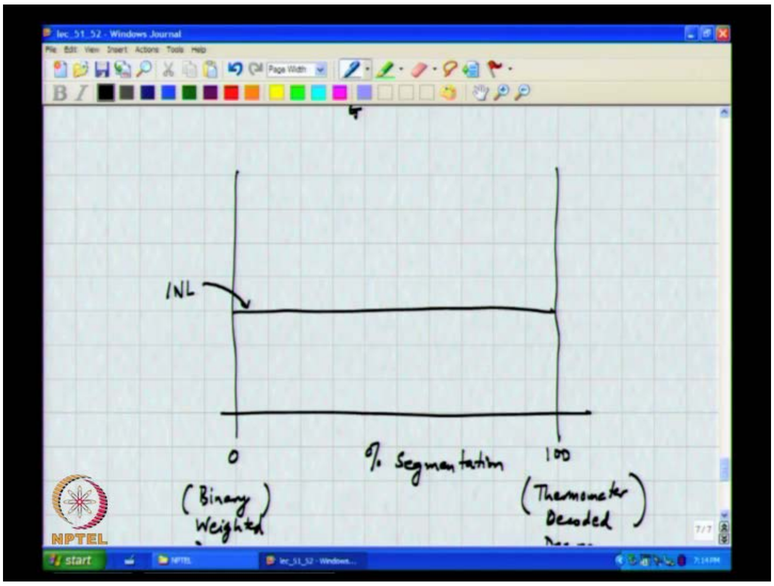
(Refer Slide Time: 31:40)



So, is this clear that the MSB's must be decoded in a thermometer fashion? So, if the MSB's are decoded in a thermometer fashion then we will have the decoder here, b 3 b 2 and what must each of these conductances be 4 G. The LSB's are decoded in a binary fashion. So, this must be 2 G and G and this will be controlled by b 1 and this will be controlled by b 0 does make sense. So, now, what is the within codes error in the step size at the major code transition; at the major code transition in other words when we go from all 0 followed by all ones what is happening which all are on this is on and one of these guys is on.

So, the step size will be or the output level will be 7 G plus delta G 2 plus delta G plus delta 4 G or delta G 4 a and when the code jumps to one followed by all zeroes it will be V ref times 8 G plus delta G 4 a delta G 4 a plus delta G 4 b. So, what is happening when the code becomes one followed by all zeroes this is getting turned on? These two are getting and those two are getting turned off. I mean here also there is differences between you know you just realizing that G by taking the part of it is realized as a difference between two large numbers. However, those two large numbers are not quite as large as they would have been if I did a complete binary decoding. That I will become apparent when you do the subtraction. So, what is the difference is V ref times G plus what variances I add time. Thankfully, this does not come in to the picture because it exists in both cases. So, this becomes G plus delta G 4 b plus delta G 2 plus delta G.

(Refer Slide Time: 37:03)



Therefore sigma DNL has now become searching square the root of 7. So, sigma G by G times square root seven what was earlier square root 15 become square root seven does it make sense. So, what can I say about say sigma INL? It should sigma INL remains the same. So, if you segment too little what does it mean? DNL. Either you have more complexity over here so, as you keep changing the degree of segmentation, which is defined as the number of bits; I mean clearly it is established that it makes sense it when you segment it makes sense to make the MSB portion thermometer coded and the LSB portion binary coded or binary decoded rather.

So, you can define the percentage segmentation as if it is completely binary you call it zero percent segmentation if it is completely thermometer you call it hundred percent segmentation and therefore, you know depending on the number of bits you code you decode in a thermometer fashion you define a percentage segmentation which ranges from all the way from zero to hundred.

As you keep increasing the percentage segmentation whether if it is zero. This is a completely binary weighted design and if it is hundred it is thermometer decoded design. And what happens to sigma INL as you change the percentage segmentation sigma INL remains the same and what happens to the sigma DNL? The worst case DNL sigma DNL goes down as you go on increasing the percentage of segmentation. So, we will continue with this in the next class.