Design and Analysis of VLSI Subsystems
Dr. Madhav Rao
Department of Electronics and Communication Engineering
International Institute of Information Technology, Bangalore

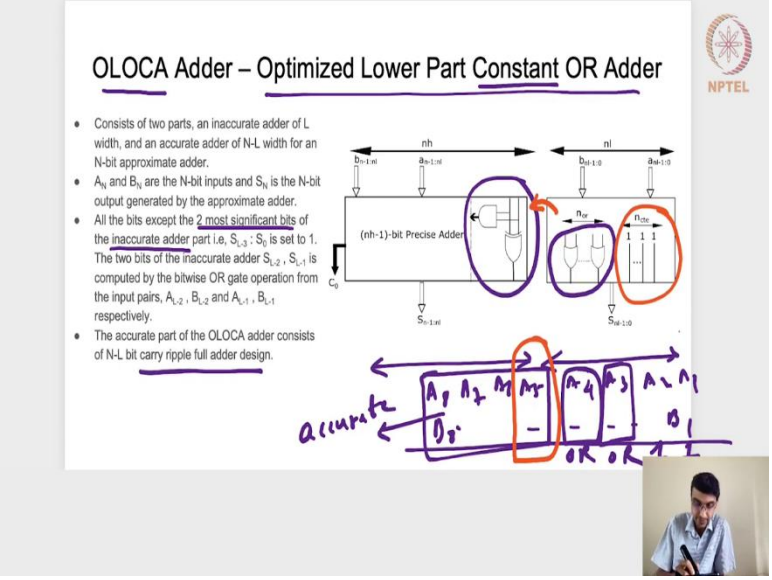
Lecture - 94
Approximate Adder

(Refer Slide Time: 00:16)

Hello students, welcome to this particular lecture. The title is the Approximate Multiplier Designs using the approximate adders, in the sense, it is kind of novel, because I think this is the first time I think, we have designed this, we in the sense I and my students, the research students have worked on this and then developed this Novel Approximate Multiplier Designs using the approximate adders.

In the last lecture what we have seen is, an approximate multiplier design using, the approximate compressors. Here, we are using making using of the adders itself for the partial products addition. I think, the reference paper can be taken from this, I have put the reference paper, the title is Novel Approximate Multiplier Design for the Edge Detection Application and the, this is available in the proceedings.

(Refer Slide Time: 01:13)



Let us take a look into the approximate adders itself. What we have used is OLOCA adder, which is nothing but Optimize Lower Part Constant OR Adder. It is pretty simple in the sense that there are 2 parts, 2 components to this particular adder, one is the MSB side and then the other one is the LSB side. If I have a 16 bit adder or an 8 bit adder. The 4 bits of the adder is kind of segmented into the LSB side and then the remaining 4 bits could be considered as the MSB side.

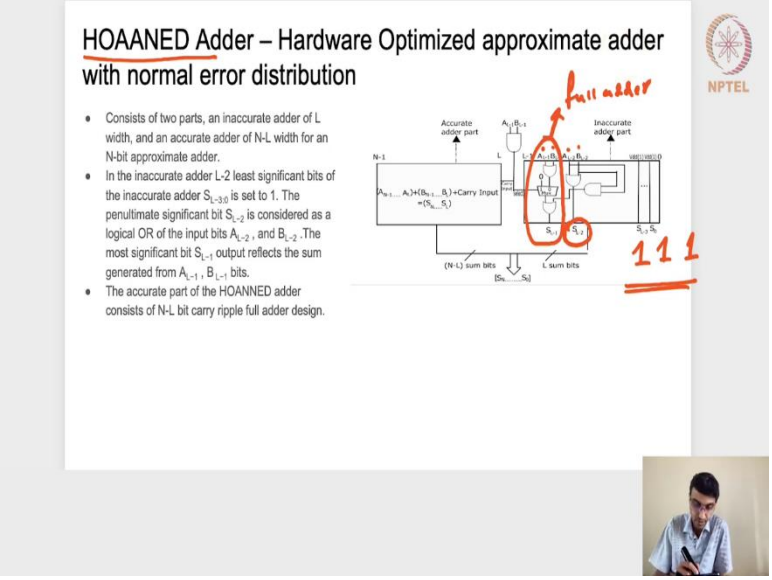
In the LSB sides except the 2 bits to 2 MSB bits, the 2 most significant bits in the inaccurate part in the LSB side, we will have an OR gate. The other set of bits here can be directly considered as a constant 1, that is why it is called as a Constant and then an OR gate. What we are saying is if I have an 8 bit here, A_8 A_7 A_6 5 and then 4 3 2 and 1 and if it gets added with another B_8 to B_1 here.

What we are saying is, irrespective whatever the value is A_2 and A_1 or B_2 and B_1, we will consider it to be directly a constant 1 here. Then, this 2 bit addition here what we can do is, we can do an instead of an XOR operation here to get the addition done, we directly use A_4 or with of B_4 and A_3 or with that of B_3 to get the results, this is an OR output and the remaining a 4 bits, we can consider it to be an accurate adder. This is an exact adder, where we can use the carry ripple full adder design or any of the faster adder designs. The first 4 bits we can make it an approximate adders. This one, the diagram here represents that the bit here, the least significant bit in the accurate portion of the adder.

Since there is no carry output here, there is nothing like that. We use the half adder here and then the carry out of the half adder gain gets propagated or the carry here propagated to the carry ripple adder, using the full adder designs here, in the remaining 3 parts of the accurate addition part.

In the inaccurate portion, on the lower LSB side, everything is 1 here, except the 2 bits. The 2 bits output will be nothing but, coming from the OR gate output. It is pretty simple in the sense that, we do not use any kind of a gates in this lower portion and then we use only the OR gates instead of an AND gate or an XOR gate. The hardware in terms of the area or the power of the deal can be benefited as lot using this particular the approximate adder design.

(Refer Slide Time: 04:20)



The next approximate adder design is, Hardware Optimize approximate adder with normal adder distribution. Here also, it is been segmented or divided into 2 portions, one is the accurate portion, the other one is the inaccurate portion. In this except the 2 bits here, all other portions, all other bits coming out from the inaccurate adder is actually considered as 1.

These 2 MSB bits is getting from the S_{L-2}, this sum coming from this 2nd bit 2nd most significant bit is nothing but, coming from an OR and then from an OR gate here. This is basically an OR representation very very similar to that of the OLOCA adder.

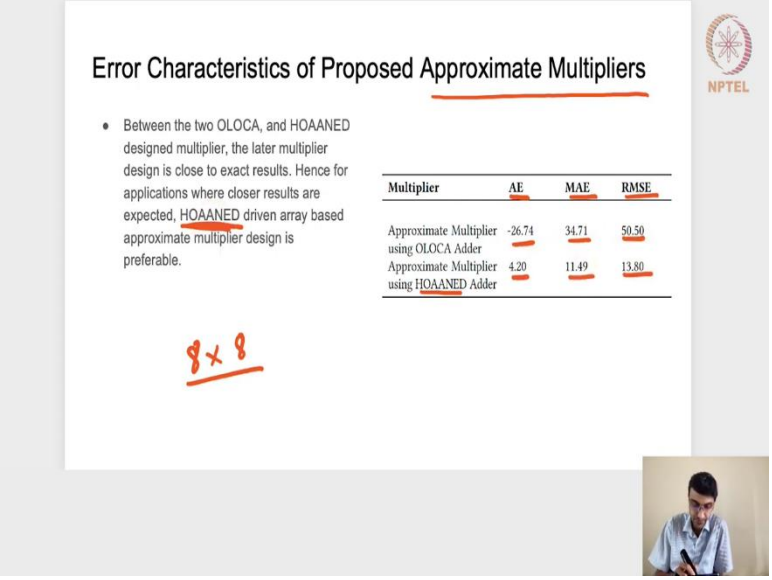
But it's the AND gate is there which will help it in carrying carry forward. This AND gate represent the carry out of the addition of this 2 bits, A and B in this 2nd most significant bit in the inaccurate portion.

This particular carryout, is put into this particular portion of the design. If I consider this particular portion of the design, it is actually the exact adder. Its basically a full adder design. Although it uses a multiplexer to make it little bit more faster and then the A and B will give me the carryout signal, which will be fed into the accurate portion.

Remember that in an OLOCA there was no carryout, coming from the inaccurate portion and fed to the accurate portion that was not at all there and we had 2 OR gates here. The first the most significant portion, the most significant bit in the inaccurate portion is having a full adder design, little bit more hardware is there compared to that of the OLOCA design.

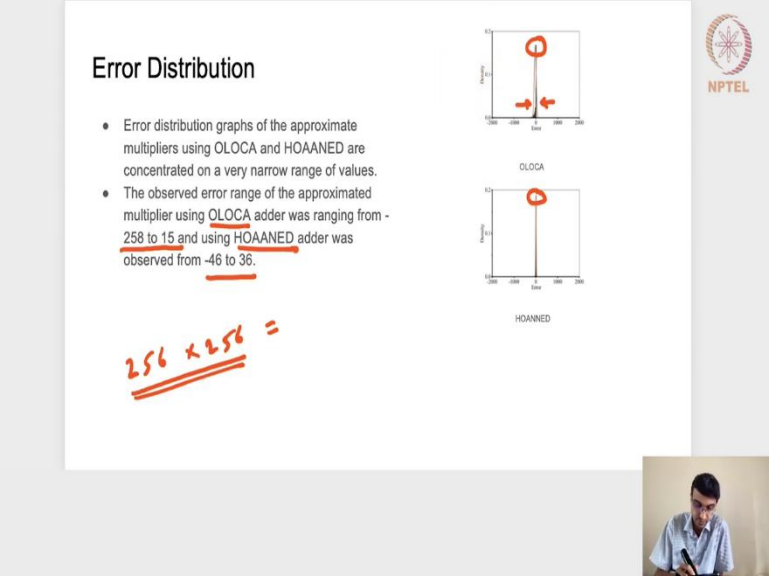
What we should expect is of course, we will expect an approximate value, because the inaccurate portion we have used some of the bits directly a considered to be 1 and this one is an OR gate. It is not an XOR gate, there will be inaccurate research. But it should be closer to the accurate results of an exact adder, because we have utilize little bit more gates here. So, the power, area and delay of this particular adder HOAANED which is a Hardware Optimize approximate adder with normal error distribution, for this particular adder it will be slightly more, than that of the previous OLOCA adder.

(Refer Slide Time: 06:46)



If I use this particular 2 proposed adders, to approximate adders into an multipliers we will get the approximate multiplier. If I actually do an 8/8 bit multiplication using these 2 bit adders, then this is the results we are likely to get the absolute error, the mean absolute error and the root mean square error turns out to be this particular the 2nd approximate adder is giving the lower results, compared to that of the 1st approximate adder, because the 2nd approximate adder was anyone in the inaccurate part, the most significant bit was giving the right results. As compared to the OLOSCA adder, where we had only the OR gates, in the most significant bit of the inaccurate adder. In that sense we were expecting the 2nd approximate adder which is HOAANED, to give us the results closer to the accurate value, thereby the error values are reduced.

(Refer Slide Time: 07:50)

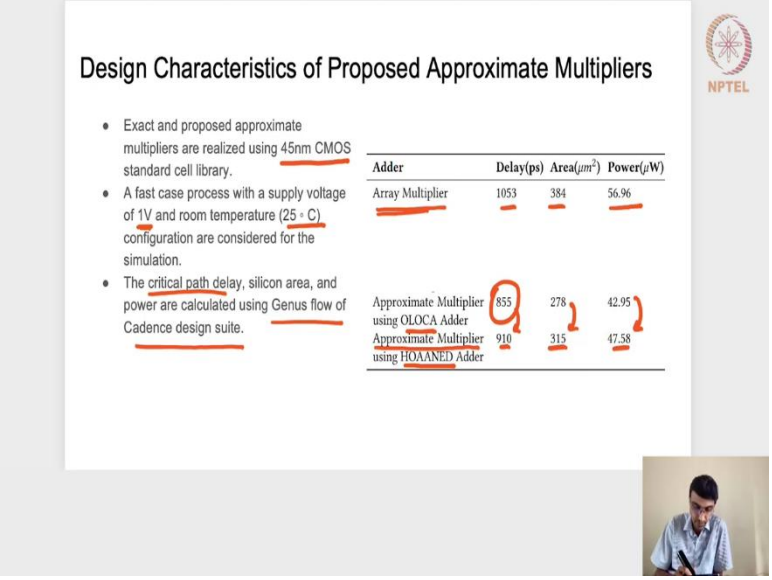


In terms of the error distribution, we can see that the OLOCA has slightly more width here compared to the HONAANED which is very very sharp. The accurate results has a density more and here also in the OLOCA the density of the accurate results is more. The width is slightly more, in the sense that OLOCA adder was has an error ranging from 258 to 15 and HOAANED has an error ranging from -46 to 36.

The way we design this particular error distribution and then the error rate for an 8 bit multiplier we can have 256 x 256. Out of all this combinations, what is the error we are getting and then, get this statistics if we have all this particular values compare that with that of the accurate or an exact values and whatever is the error and then we actually define

the statistics of the error. Whatever is the error, we can actually plot it in terms of the probability distribution.

(Refer Slide Time: 08:55)

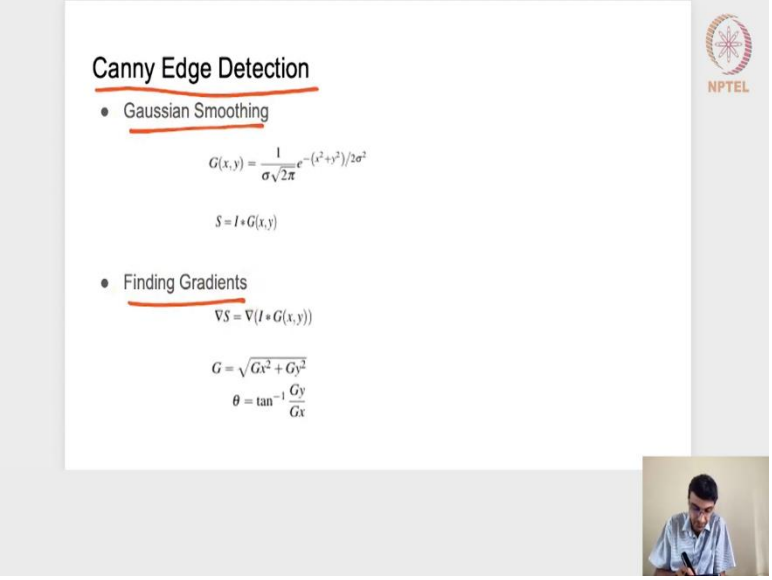


In terms of the design characteristics, if I use the 45nm technology, CMOS standard cell library and then if I use it a regular ESSIC process, where it is 1 volts and then 25^{o} Celsius, if the temperature is configured in such sense and in the simulation.

The critical part delay, the silicon area and then the power are calculated using the Cadence Genus flow then we will get this kind of an statistics. The adder multiplier which is an exact multiplier takes around this particular values. Whereas, an approximate multiplier defined driven by OLOCA adder, is giving us much reduced results. The 855 compared to 1000, 278 compared to 384 and 42.95 as the microwatt compared to 56.96.

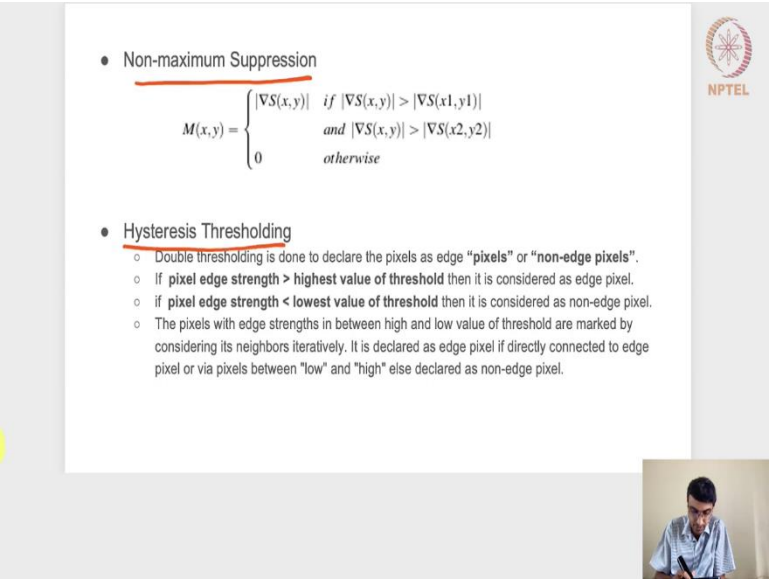
HONAANED which is an approximate adder again which if you use it to design our approximate multiplier using this particular adder. The results are much better compared to the exact multiplier. But the results are slightly higher here, on the 3 parameters are where analysis parameters are more than that of the OLOCA adder, in the sense that there are little bit more gates that are being involved in the inaccurate portion especially in the MSB part of the inaccurate adder portion and thereby it is likely to give us slightly more hardware parameters.

(Refer Slide Time: 10:21)



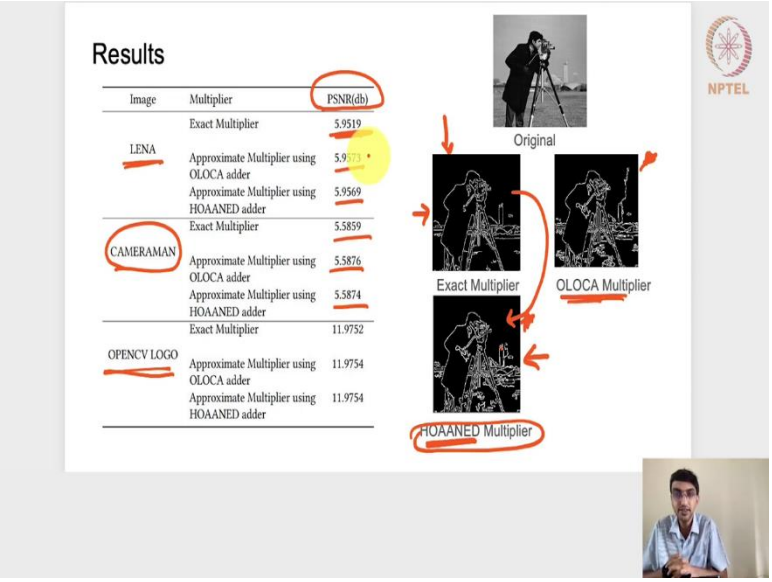
Now approximate multiplier as such it is very very useful in some of the image processing application and one such image processing application is identifying the edges. Edge Detection Algorithm and then Canny Edge Detection is one such very very popularly used algorithm. It involves a few steps. One is the Gaussian Smoothening and another is the Finding the Gradients.

(Refer Slide Time: 10:44)



Then the Non-maximum Separation and then the high Hysteresis Thresholding. The Gaussian smoothening is where I think the multiplier will be heavily used and if I use those 2 approximate the multiplier designs this is what the results we get.

(Refer Slide Time: 10:58)



The original results this is the original image of a cameraman and exact multiplier if I use a to build the Gaussian smoothening and all the other steps to get the Canny edge detection, we will get this particular image.

Using the approximate multiplier using the OLOCA adder, we will get this particular image that is been rendered and using the approximate multiplier or the multiplier design using the approximate an adder here of this particular type we will get this particular image. Looking at those particular image is not much is large, what we are comparing is this particular image, we have to compare with this particular image and then this particular image.

The HOAANED shows everything very very similar to that of what is there in the exact multiplier, even this particular portion is kind of showing it properly. Whereas, the OLOCA has little bit of errors and that is kind of intuitive in the sense, OLOCA uses much more approximation in terms of the gate designs. Whereas, the HONAANED uses a little bit more gate designs to bring the results closer to the accurate ones.

The HONAANED multiplier is actually preferred in terms of the error characteristics. But in terms of the area, delay and power, the OLOCA multiplier is little bit more preferred. If I want to actually get into compare between the exact multiplier output with that of the other approximate multiplier output. There is a term there is an index called as PSNR which is Power of the Signal Noise Ratio in terms of dbs.

We can clearly see that the exact multiplier and then the approximate multipliers gives us very very similar results for all the images the Cameraman images is one which is shown here, 5.58 even the approximate multiplier shows the same results.

If I use a different image, OpenCV Logo, I will get the similar results and if I use a different LENA image, also we will get the similar PSNR values. Using the approximate multipliers does not degrade the PSNR that much. It gives us the values similar to that of using the exact multiplier.

In this particular lecture what we have seen is 2 approximate adders which have been used to design the approximate multipliers and then we also went through the error characteristics as well as the hardware parameters characteristics. Hope you are able to understand this particular lecture.