Digital Voice and Picture Communication

Prof. S. Sengupta

Department of Electronics and Communication Engineering

Indian Institute of Technology, Kharagpur

Lecture - 29

Audio Coding: AC - 3

We will continue with the discussions on the audio coding and in the last class I had talked about some of the very basic concepts pertaining to the audio coding and today we are going to take up one specific audio coder and that is the AC-3 audio coder, this we are going to discuss in some details today. Before we go in to the audio coding techniques adopted in the AC-3 let us first spend little time on what any general audio codec must contain.

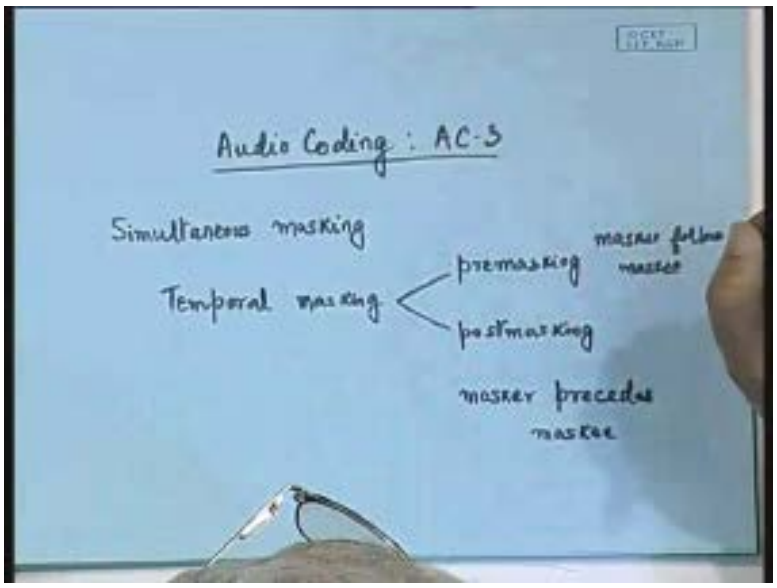
In the last class we had got certain concepts and as I had told that the phenomenon of masking is really very important in terms of the psychoacoustic bit allocations. So there what we actually do is that depending upon the signal some specific maskers the tones which tend to mask the other signals those maskers are identified and then those maskers will serve as the masking signal to mask any other signal which are rendered inaudible. So essentially what we should do is that the bit allocation should be done in such a way that the signal to noise ratio resulting out of this or rather to say the noise level that is incurred in the process of bit allocation resulting out of the quantization noise that noise level should be never exceed the masking level and as long as it is kept just below the masking level we need not have to allocate any extra bits so that is why some amount of bit saving could be done by taking advantage of the masking phenomenon.

Now the masking phenomenon that we described in the earlier class is also referred to as what is known as the simultaneous masking and why simultaneous because there it was assumed that the maskers and the maskees they are as if to say appearing at the same time instant that is why they are called as the simultaneous masking. But it has also been observed that other than the simultaneous masking there is also a kind of masking that occurs and that is known as the temporal masking. By temporal masking what we mean to say is that the masker and the maskee may not happen exactly at the same instant.

In general, one can say that the maskees, I mean, the once which get masked should precede the masker. Means, if the masker happens now then its effect would be such that the maskees can have the effect of the maskers or some masking effects will be felt even after the masker has occurred so that is actually called as post masking. But you may also be surprised to know that there is also a thing which is called as pre-masking.

So post masking, post masking means what is more logical that is to say that the masker precedes maskee in time and there is also what is called as pre-masking and in case of pre-masking the masker follows the maskee. Looks little awkward that the masker appears later than maskee but it is indeed so, I mean, it has been observed experimentally what actually happens is that the masker being a very strong signal and the maskee being a relatively weaker signal, in the psychoacoustic phenomenon that is to say as far as our perception is concerned it is seen that even though the maskee may follow little earlier but to a limited extent the masker can also mask the maskees by this phenomenon what is known as pre-masking.

(Refer Slide Time: 6:12)



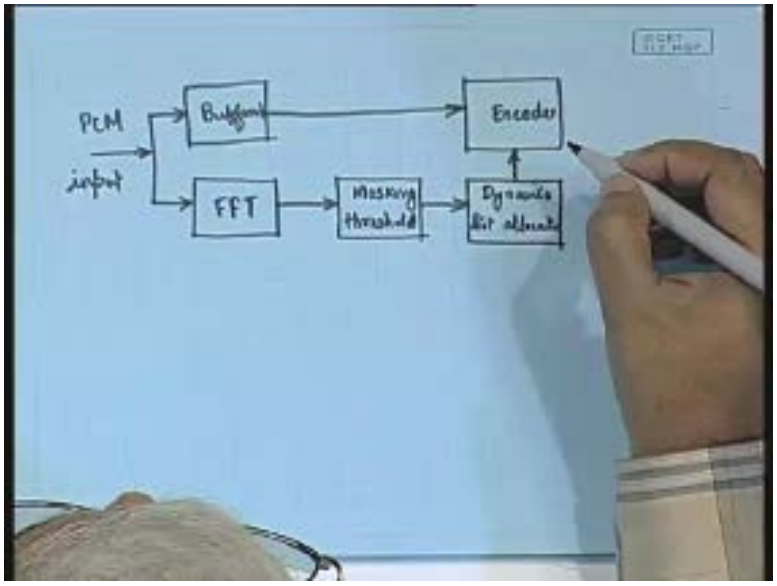
So temporal masking also is an important phenomenon although there the post masking happens to be more prominent as compared to the pre-masking but some effect of pre-masking is also

there. Now this is about the masking phenomenon; already we have discussed in sufficient details; it is about the masking phenomenon and then also talked about the various definitions like the signal to mask ratio, mask to noise ratio and how it is related to the signal to noise ratio, these aspects we have seen.

Now let us go over to the general perception based coder. Now a general perception based coder should look like this that this is where we have the PCM input the pulse code modulated input of the audio samples so this is the PCM audio samples and this bifurcates into two parts: one in which it is buffered and the other in which its FFT is taken.

Now the part for which the FFT is taken the FFT outputs are considered in order to compute the masking threshold. This I have already talked in the last class that what is the masking threshold. Masking threshold is actually computed based on the psychoacoustic observation and then the masking threshold basically decides the number of bits which are to be allocated so there is a block which is called as the dynamic bit allocator dynamic bit allocator and this dynamic bit allocator will decide on the bits which will be encoded. So there will be an encoder so all the buffered samples they will be encoded but how many bits will be allocated to the encoded samples that will be decided by the dynamic bit allocator.

(Refer Slide Time: 8:57)



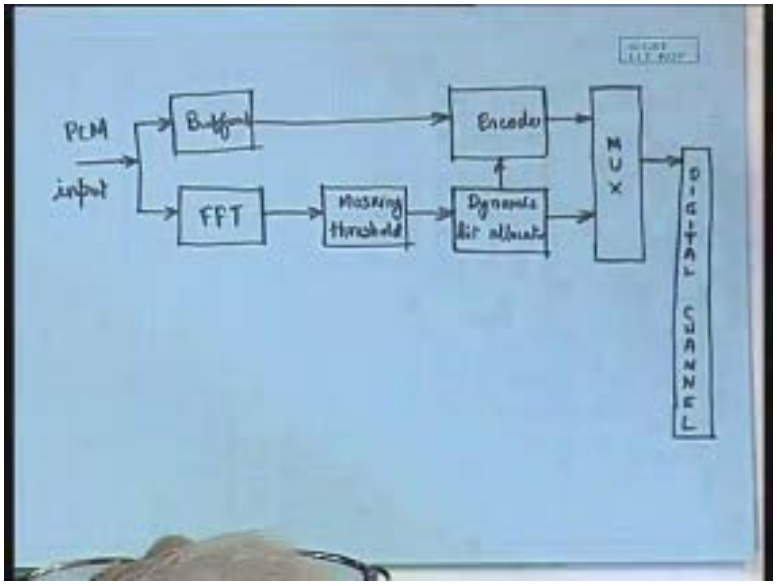
Now, when the information goes into the channel that time not only the dynamic bit allocator information but also the encoded samples that means to say the encoder bit information both should go into the bit-stream. So we must have a multiplexer and the multiplexed output will be fed to the channel and the channel we can assume it to be a digital channel. Although there is no problem if we assume analog channel in which case we have to convert from the digital to the analog and then transmit it but in general one can assume a digital channel and at this point of time we need not have to consider the losses that is involved in the channel.

Assuming that the channel is lossless, this is the digital channel, this part means from the PCM input up to the generation of the bit-stream multiplexed bit-stream this forms the total encoder part. Now what I want to point out at this stage is that (Refer Slide Time: 10:15) why is this FFT being done. This is a very important aspect to understand. You see, the FFT block basically picks up that what are the tonal components that is present in the signal. Like say for example, in a signal we may be having a presence of a single tonal component supposing 1 kHz supposing it is a pure 1 kHz tone then the FFT is going to give us a peak at 1 kHz and no signal or very weak signal in the frequency other than 1 kHz so it is ideally supposed to give an impulse at 1 kHz.

If we have 1 kHz as well as let us say one point seven kHz then we will be having two tonal components: one at 1 kHz, one at 1.7 kHz. Now actually in an audio frame we will be having presence of several such tonal components, it is never going to be a single tone but always going to be a combination of different tonal components. So depending upon those tones the FFT block will identify that which are those tonal components and there should be a processing block which I did not show in the block diagram in the general block diagram and that processing block will identify the tonal components and will reject the non-tonal components from it. So there may be presence of some weaker signal components also which will give rise to some local maximas in the frequency spectrum.

You can obtain..... basically through FFT what you were doing is to obtain a frequency spectrum so the frequency spectrum could have some prominent maximas which correspond to the tonal components and it could also contain some local maximas and we would like to eliminate those local maximas so some amount of post processing has to be done on the FFT samples but nevertheless once the tonal components are identified then we find out that in which frequency band..... by frequency band I mean to say the critical band which I have already mentioned in the earlier class it is identified that in which critical band the tonal component belongs to and depending upon that band and the characteristic masking characteristic that is specified in the psychoacoustic model. So we have to consider we have to consult a psychoacoustic model so depending upon the strength of the tonal component and the frequency band or critical band to which it belongs we will be computing the masking threshold and it is this masking threshold based on which the allocation is done. So it is a complete encoder where the perceptual bit allocation is considered.

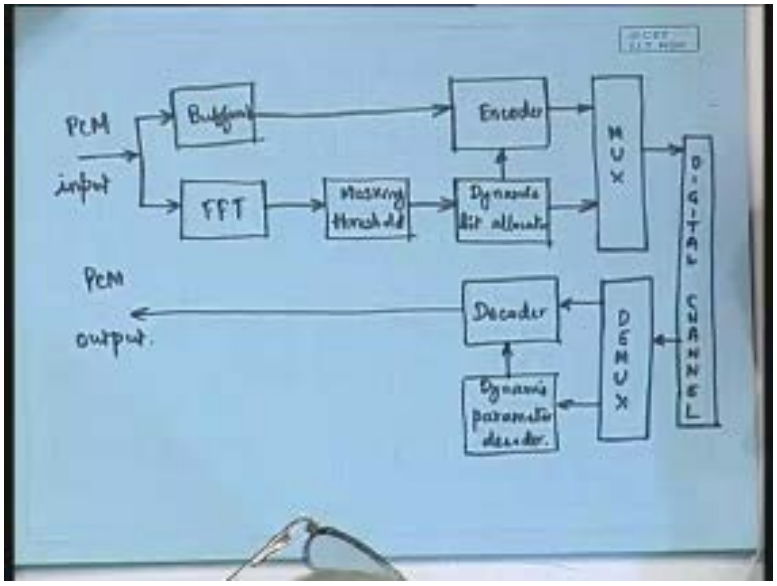
(Refer Slide Time: 13:35)



Now, in the decoding process we have to do exactly the reverse of this. So definitely a decoder has to start, I mean, if it receives the signal from the same digital channel then the decoder has to start with a block which is performing a de-multiplexing. So it must have a de-multiplexing and the de-multiplexing should separate out these two components because here we have got two components one is the encoder information and the other is the bit allocation information. So these has to be separated out so the demarks will have two parts: one is the actual coded signals they are to be decoded so we must have a decoder that will be applied on the samples but we cannot decode exactly till the time we obtain the bit allocation information.

The bit allocation information also has been put through this multiplexer in a coded form so definitely we must get it back. So since it is in the coded form we must have a dynamic bit allocator decoder dynamic parameter decoder we should say a dynamic parameter decoder which will extract the bit allocation parameters and after extracting the bit allocation parameters it has to signify it has to notify the decoder accordingly because based on this the decoder will extract the samples; the actual extracted samples will be obtained and the decoder output will be the PCM output which will be used for the reconstruction of the signal.

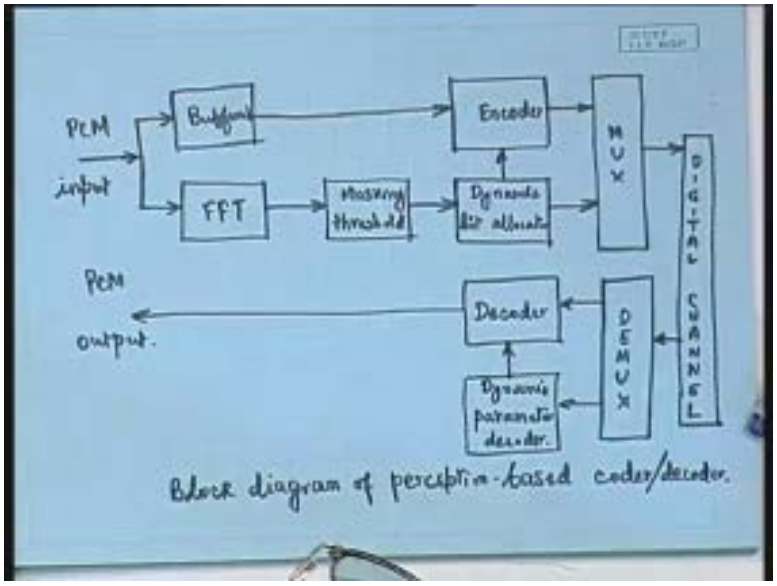
(Refer Slide Time: 15:55)



This is where we expect the PCM output to be exactly same as the input provided the model is absolutely lossless. but nothing is lossless because in order to achieve some amount of compression even this encoder what I have shown as a bit block is also containing a quantizer so there will be some quantization and more importantly you will also have some channel noise that is associated with it. So the PCM output only under ideal condition could be the same as that of the PCM input.

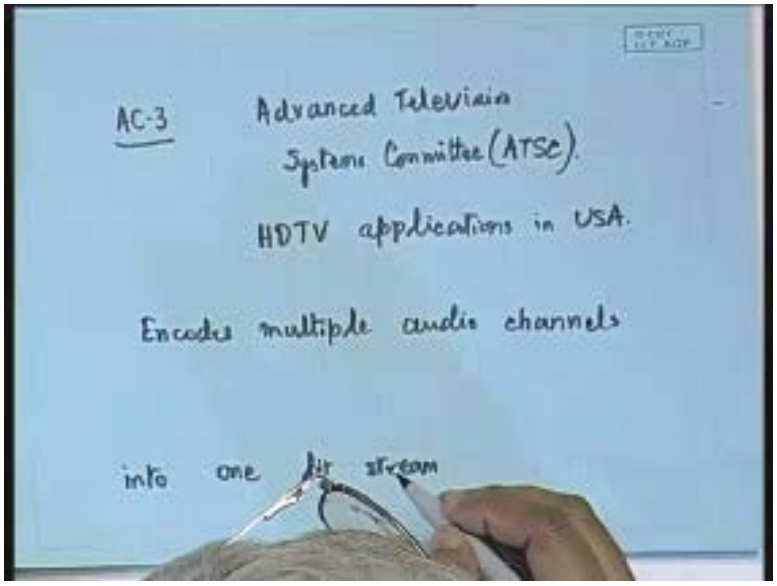
This is the generalized..... so I can say that this is the block diagram of a perception based coder decoder. So block diagram of perception based; perception in this case the audio perception so this is perception based coder and decoder and in fact this is the general structure which is followed in the MPEG audio coding and also to a great extent this same model is followed in the AC-3 codec which I am going to describe to you now.

(Refer Slide Time: 17:23)



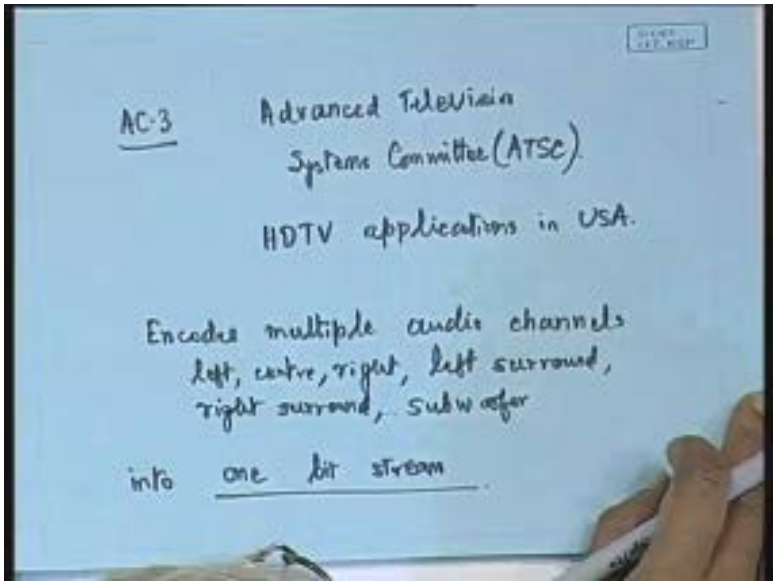
Now talking about the AC-3, AC-3 as I had mentioned is the coding standard that was actually proposed by the advanced televisions systems committee, so this was by the Advanced Television Systems Committee. In fact it was developed by the Dolby Advanced Television Systems Committee or in short form what is known as the ATSC ATSC and ATSC had adopted this AC-3 standard for the high definition television or the HDTV applications in United States. But other than this being adopted as the HDTV standard the AC-3 is also adopted as an audio coding standard in many applications in in the digital audio. This is a very popular digital audio encoding scheme and here basically this supports.....I mean, the beauty of the AC-3 is that it supports multiple audio channels so it encodes multiple audio channels into one bit-stream into one bit-stream.

(Refer Slide Time: 19:25)



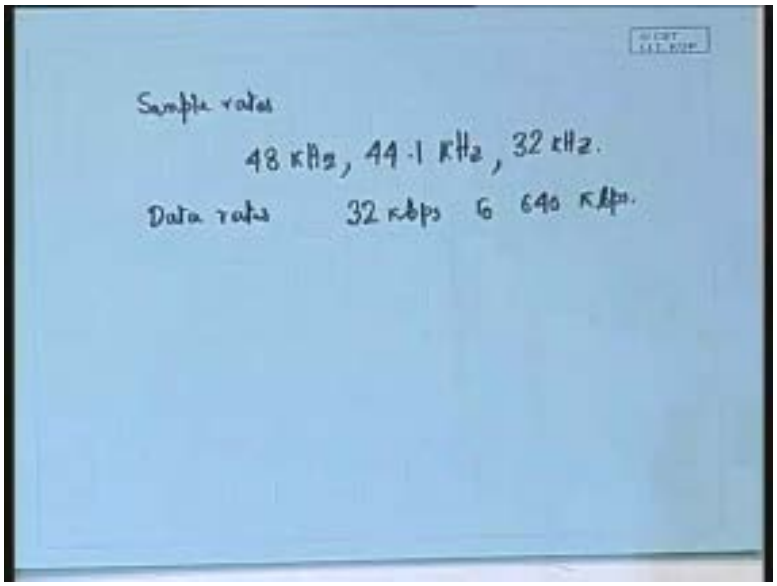
Now what are the multiple audio channels; why is multiple audio channel really necessary?
It is seen that in order to have the best effect of the sound what you do you normally adopt a stereo where you have the left channel and the right channel but not only these two channels are important but if you want to create a better effect of the sound then you not only require the left, you also require the center, you require the right, then you require the left surround; you will find that in many of the advanced audio equipment the left surround and the right surround they are available as the channels so left surround right surround channel and the subwoofer channel that makes six different channels, so six different channels are there; six discrete channels are there and this could be encoded into one single bit-stream.

(Refer Slide Time: 20:56)



Now the AC-3 bit-stream specification actually permits the sample rates; the sample rates which are specified there are 48 kHz, then 44.1 kHz and or 32 kHz and it supports data rates that is ranging from 32 kbps to 640 kbps.

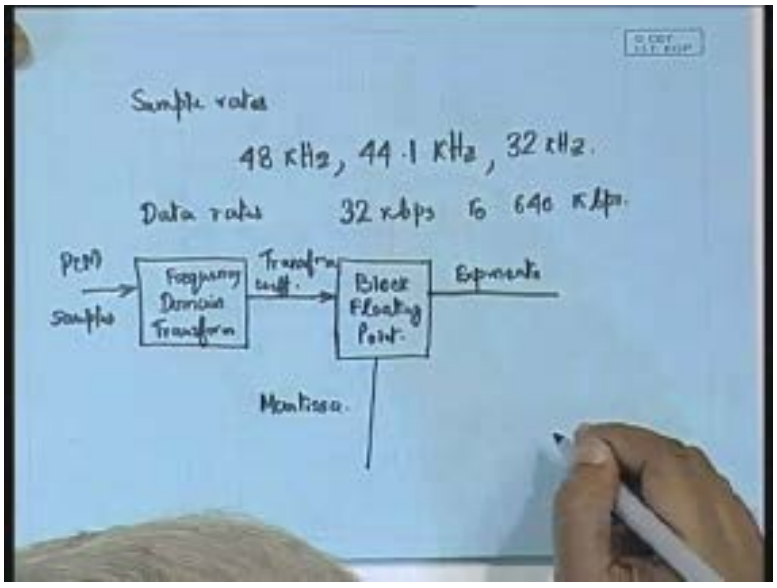
(Refer Slide Time: 21:45)



Now let us have a look at the block diagram of the AC-3 encoder. Now here we have the PCM samples at the input to the system as before, as we had seen just now with the perceptual audio coding, then there is a frequency domain transform which is applied. This is a frequency domain transformation (Refer Slide Time: 22:25); I was just now saying that here the FFT was performed in order to identify the tonal components but of course FFT involves a complex computation. So computationally it is quite involved and that is the reason why we had seen that in the case of the images and videos we were adopting the discrete cosine transform which is having a real kernel.

But actually speaking, for the audio encoding, the DCT is not applied directly in fact what is applied is a Modified Discrete Cosine Transform MDCT which basically permits a time domain alias cancellation so that aspect I will be talking shortly or may be in the next lecture I will come to that and the frequency domain transform this basically takes up the transformed coefficients and the transformed coefficients basically goes through a block floating point unit because interestingly in the AC-3 the computation is done in the floating point mode involving the exponent and the mantissa. So this is block floating point (Refer Slide Time: 24:13) which separates it out into the exponents and the mantissa part.

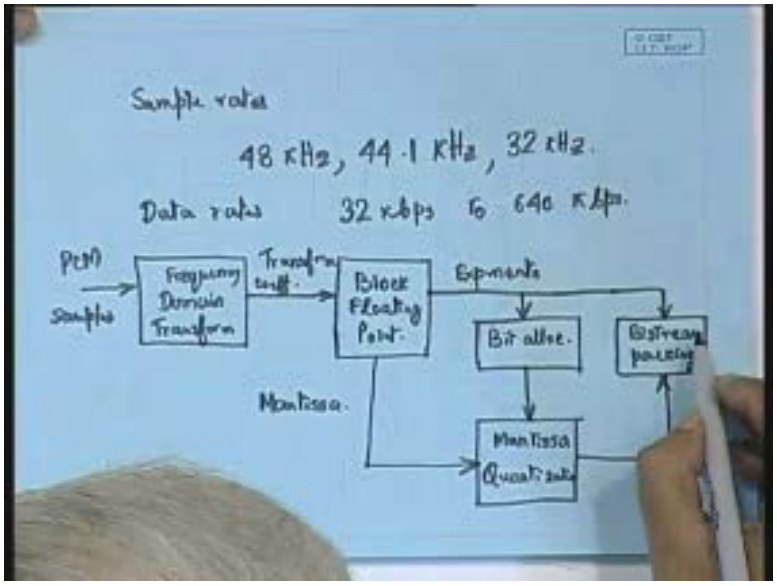
(Refer Slide Time: 24:25)



Now the exponent part basically does the now this these mantissa components they are basically put through a quantization process so there is a mantissa quantization and the exponent part goes through a bit allocation. There is a bit allocator and the bit allocation model basically controls the mantissa quantization.

For the mantissa components how many bits will be allocated that will be decided by this bit allocation model. Basically this is a variable length encoding which is being done for the mantissa whereas a fixed length encoding is done for the exponent part. then what happens is that this exponent as well as mantissa along with the bit allocation information that should go into the encoded bit-stream because all three are important; we not only need the exponent but we also need the quantized mantissa so there should be a block which we call as the bit-stream packing.

(Refer Slide Time: 26:20)



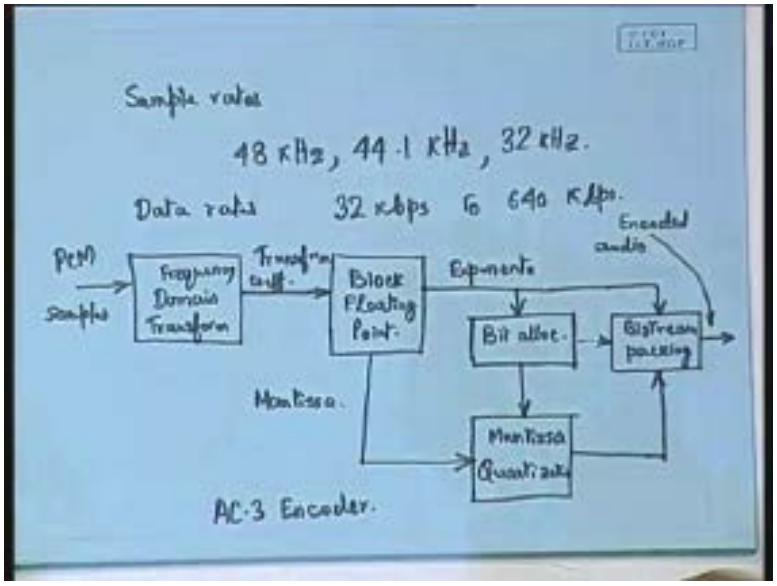
This is the bit-stream packing and then we have the encoded audio output so this is where I do not have space to write so I am writing here so this is where we generate the encoded audio.

Then not only should this information go but also the bit allocation information which should be put into the bit-stream.

Now, in the case of the AC-3..... so this is the AC-3 encoder block diagram so this is the block diagram of the AC-3 encoder. now instead of sending the bit allocation information directly, in the AC-3 encoding what is being done is that the parameters that first the parameters are extracted which are used for the bit allocation and those parameters can be encoded with fewer number of bits. So the bit allocation information is not sent directly into the bit-stream but in fact a parametric approach is taken. So the encoder constructs the masking model based on the transformed coefficient exponents and a few signal dependent parameters. So the whole idea will be to extract the masking information because the bit allocation has to extract the masking information which should depend upon the exponent part.

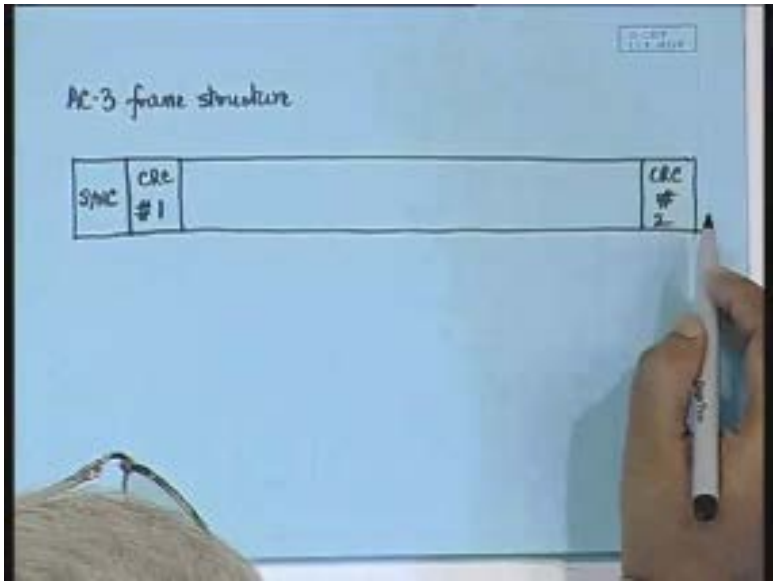
You notice one thing that here (Refer Slide Time: 28:09) the transformation is done only at one stage whereas you will be seeing that in the MPEG audio standards MPEG-1 and MPEG-2 audio standards you will find that there we require one audio coding; there we have not only the FFT computed separately but also for the encoding of the signal we basically encode the signal in two different bands and in order to process those bands basically the MDTC is taken so both the type of transforms are considered there; the MDTC as well as the FFT whereas here this is a single point transformation which is being done, so PCM samples converted to the transformed domain and then on the transformed coefficients we act and the bit allocation or rather the masking information is extracted from the exponent part of the signal and that is what makes the feature mode advantageous for the AC-3; although computationally it is somewhat more involved because it is a floating point computation but because of its adoption in the standard and the popularity, several leading companies several leading semiconductor companies have designed chips to encode the AC-3 in real time.

(Refer Slide Time: 29:23)



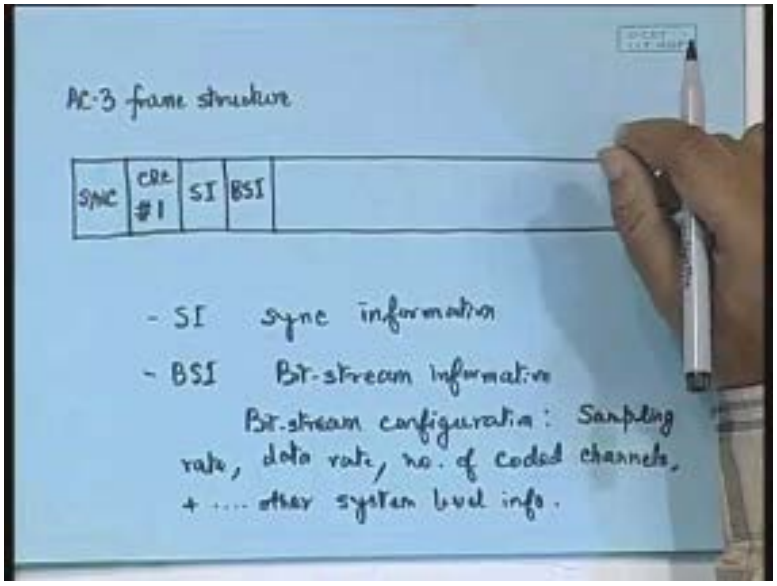
Now how the AC-3 encoding is actually done? Let us have a look at the AC-3 frame structure. The AC-3 frame would look like this (Refer Slide Time: 30:33). An AC-3 frame will first have a synchronization character and then it will have the cyclic redundancy check so CRC. In fact there are two cyclic redundancy check blocks so this is called as the CRC 1 block and at the end of the AC-3 frame we will be finding yet another cyclic redundancy check so there will be another CRC block at the end and we call that as the CRC 2. So we have got CRC 1 here at the beginning just after the SYNC characters and the frame ends with another block of CRC characters which is the CRC 2.

(Refer Slide Time: 31:23)



After this CRC 1 we have the SYNC information. The SYNC information or rather to say the SI is followed just after the CRC 1 and the SI is followed by the bit-stream information. There are two fields SI and BSI. So what are these? SI is the sync information and the BSI is the bit-stream information. Now these two fields basically they describe the bit-stream configuration so they specify the bit-stream configuration and as a part of bit-stream configuration what you need to specify is the sampling rate; the sampling rate that has to be specified, the data rate the data rate then the number of coded channels and several others plus other system level information. So this will go as you can say that these are sort of header information to the AC-3 frame structure.

(Refer Slide Time: 33:21)

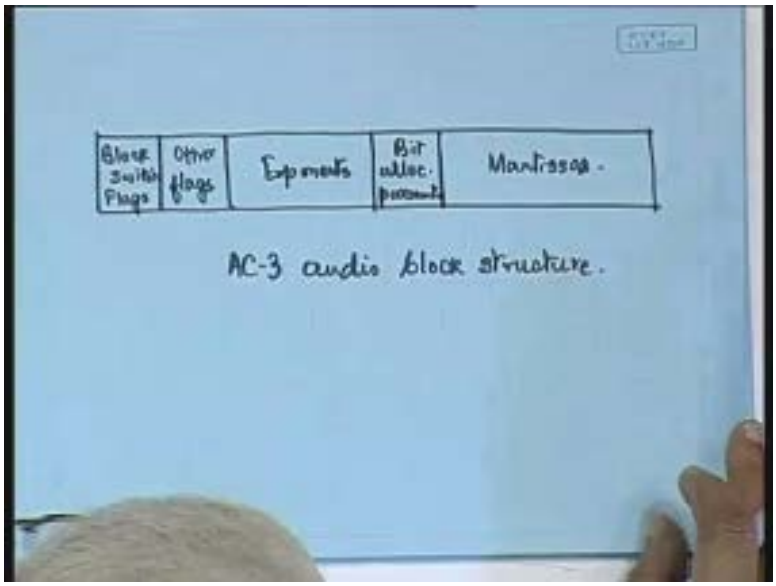


Now followed by the SI and the BSI we have the actual audio blocks and this standard basically accommodates up to six audio blocks in any channel. Each channel can have up to six audio blocks and in fact the AC-3 bit-stream this AC-3 frame each frame consists of 1536 samples per channel 1536 samples per audio channel and each audio channel actually is divided into six up to six blocks. That means to say that whenever you are having 1536 samples into six blocks means for each block we have 256 samples we have 256 samples per block then we can write down the individual blocks as..... so this will be divided (Refer Slide Time: 34:35) subdivided into six parts let us do that 3 4 5 I could not make equal partition so this is audio block 0 so this is block 0, this is audio block 1 2 3 4, the last is the audio block 5.

Now the AC-3 block structure would be something like this that we have the AC-3, so within one block we must have this information so we have the block switch flags this is actually we are referring to the AC-3 audio block structure (Refer Slide Time: 35:30) where in the block switching flags we will be having few other flags; I am not going into the details of the other flags then what we will be having is the exponent part and the mantissa part so there will be the exponents and followed by the exponents we will be having the mantissa no but between the exponents and the mantissas we are going to have the bit allocation parameters so there will be

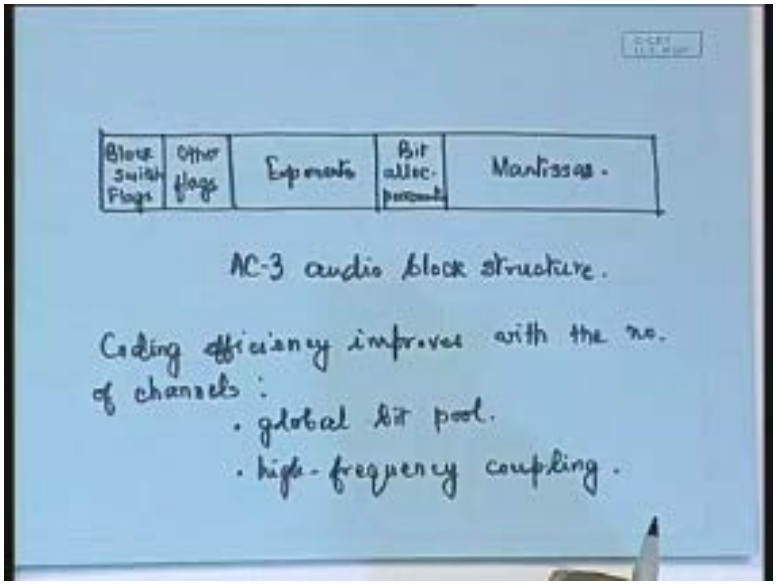
bit allocation parameters and then we are going to have the mantissas. so this is going to be the AC-3 block structure.

(Refer Slide Time: 36:51)



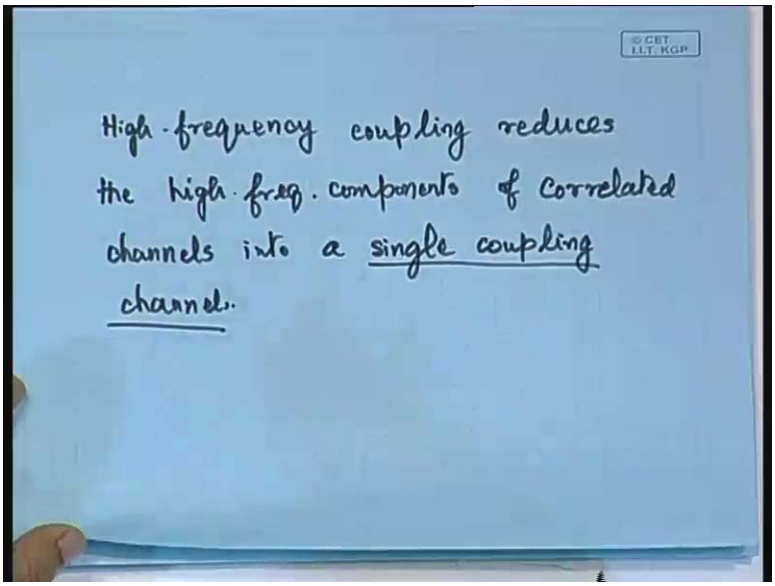
Now it is seen that the coding efficiency of the AC-3 that improves with the number of as the number of source channels increase. This point we must take into a note that the coding efficiency improves with the number of channels. Now it is due to two principle features. First is that there is a global bit pool. The global bit pool basically permits the total available bit pool into..... I mean, it allows the bit allocator to split this global bit pool into a number of different audio channels as needed. So, on a need basis the bit allocation could be done to the global bit pool so that if some channel does not require much of bit allocations then those bits can be allocated to the other channels so a dynamic bit allocation or dynamic bit adjustments in between the channel that is very much possible and that is a feature which makes the AC-3 coding very attractive. This is one part of it and the other important characteristic is the high frequency coupling that we get.

(Refer Slide Time: 38:42)



In high frequency coupling we mean to say that we have the presence of a number of high frequency channels. In the channels, between the high frequency information there could be some correlation and it is also seen that at the high frequency the ear cannot detect the individual cycles of an audio waveform and instead it responds only to the envelop of the audio waveform. So the coupling, I mean, if we are adopting a high frequency coupling then the coupling basically reduces the high frequency components of the correlated channels into a single coupling channel.

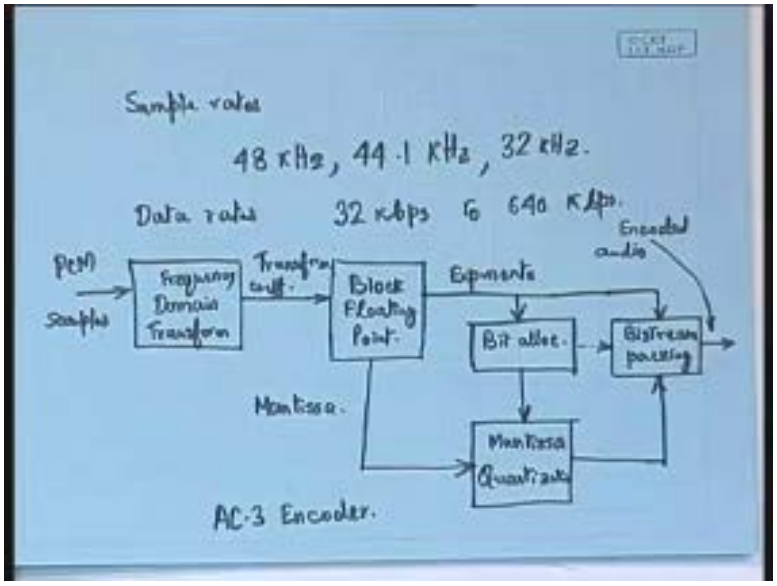
(Refer Slide Time: 39:39)



So, high frequency coupling reduces the high frequency components of correlated channels into a single coupling channel and this is intuitively obvious also that after all what are these channels; these channels are derived from the same sound source; it is not that these are independent time these are independent sources. There is one basic audio source but we are capturing that audio through different channels and those channels could be the left, the right, the center, the left surround, the right surround, subwoofer as I already mentioned and since they are coming from the same source that is why some amount of high frequency commonality will be there and basically what we are doing is that instead of sending the individual information for the high frequency components for each of these channels we are embedding them into one single coupled information; a single coupled information is what we are giving. So this is what makes the AC-3 encoding or the coding efficiency improves with the number of channels as I have said.

Now in terms of the decoder what one has to do is just to see that the reversal of what we have done can take place properly. After all what is being done is that if you are looking at the block diagram you will see that we are encoding the information as the exponent, the mantissa and also the parameters; all three are very important.

(Refer Slide Time: 42:20)

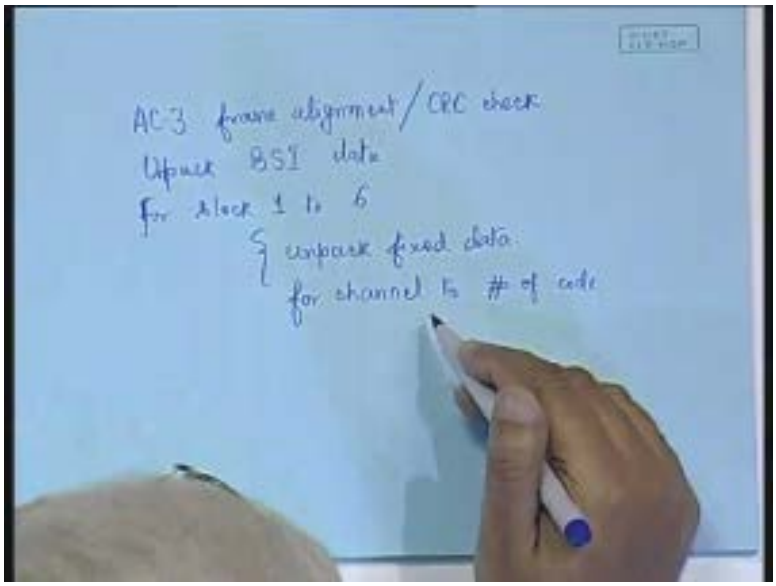


And as you can see in the block structure also (Refer Slide Time: 42:30) the exponent, mantissa, bit allocation parameters these are the three information that goes in to the encoded form so in the decoder what we have to do is to decode this information out and decoding has to take place first from the block level. From the frame structure itself first we have to identify the CRC, we have to identify the SI BSI information and then only we will be in a position to unpacked the audio blocks and once the audio blocks are unpacked then the exponent, bit allocation part and mantissas they can be unpacked.

Thus, the AC-3 decoding that can be described in the form of a pseudo code like this. We describe the pseudo code. First is that we require an AC-3 frame alignment as also the CRC check. This basically ensures that we have been able to identify the beginning of the audio frame and then we unpack the BSI data unpack BSI data and after the BSI information is unpacked then we have to unpack the block information. So, for block 1 to 6 for block 1 to 6 we unpack the fixed data; what is the fixed data it corresponds to the flags which I was mentioning; so for channel 1 to the number of coded channels..... see, in each block, mind you this is very important, in each block we could be accommodating more than one channels. Hence, after unpacking the fixed data which is actually given in the block structure the fix data will be given

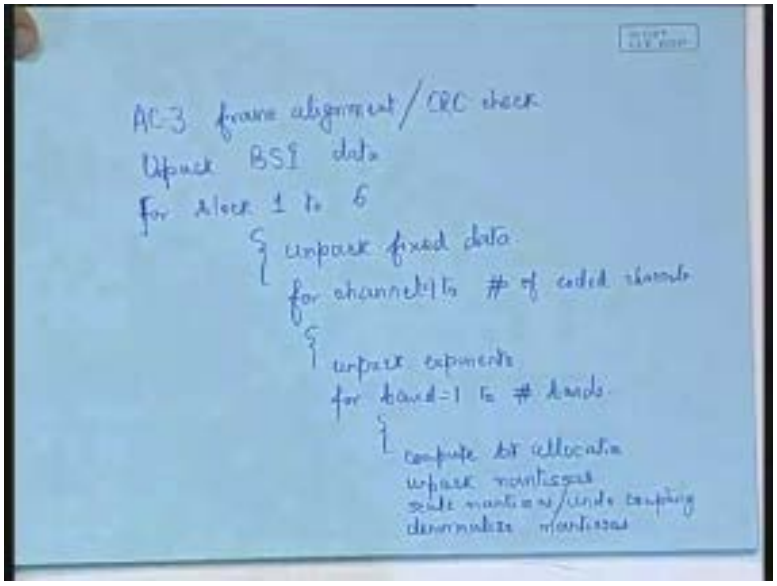
by this flag information (Refer Slide Time: 45:18) blocks with flag other flags etc and now these information exponent mantissa this will be there for more than one channels in fact.

(Refer Slide Time: 45:37)



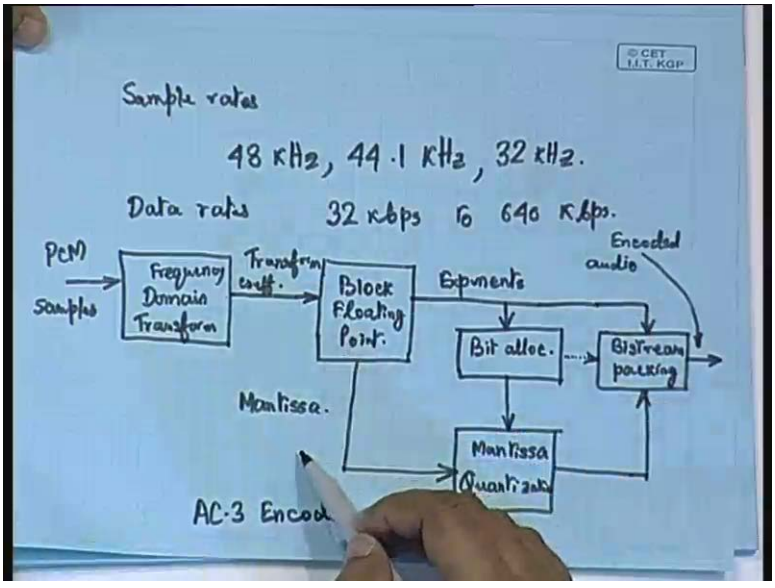
Therefore, for channel number for channel 1 to for channel is equal to 1 to the number of coded channels what we have to do is to unpack the exponents and then these exponents basically correspond to each of the frequency components. Now these frequency components they belong to different bands; by bands I mean to say the critical bands of the psychoacoustic model. for band number 1, so after unpacking the exponents for band number 1 up to the number of bands what we have do is to compute the bit allocation means extract the bit allocation parameter so compute the bit allocation then you have to unpack the mantissas. Then if you have done high frequency coupling as I have mentioned just little while back then you have to undo those coupling so scale the mantissas. After unpacking, you have to scale the mantissas or undo the coupling and then you have to de-normalize the mantissas.

(Refer Slide Time: 47:46)



After doing that what we have to do is..... see, these data are in the transform domain because so far we have extracted only the transformed coefficients.

(Refer Slide Time: 48:20)

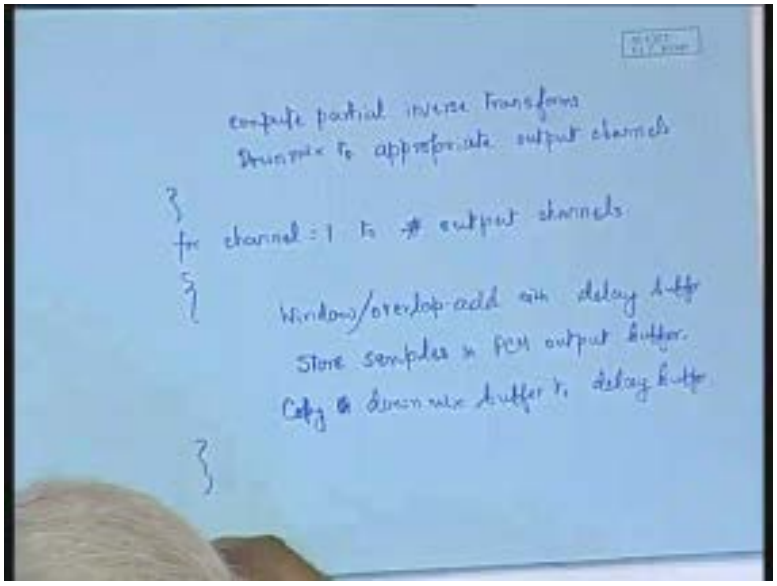


Remember the block diagram that in the overall block diagram we have it like this that frequency domain transformation and then the transformed coefficients are spilt into the exponent and mantissa. So whatever you have obtained by unpacking the exponent and mantissa they are still in the transformed coefficients. So to extract the PCM samples what you have to do is to compute the inverse transformation. So the pseudo code has to continue, so after this bit allocation and unpacking of mantissas what you need to do is to compute the partial inverse transformations.

In fact the partial inverse transforms are calculated because the technique that is adopted in the transformation is the MDCT so there is all always a partial overlap and add method which is used to in order to get back the coefficients get back the original time domain or PCM samples and then what we have to do is to down mix these transformed coefficients to appropriate output channels.

In fact, this I will be explaining that why is this input and output channels are important. Basically the number of input channels and the number of output channels they need not be the same. In fact, what the down mixing block does is that it matches the number of input channels into a different number of output channels. This I will be showing you a little later. But after down mixing to the appropriate channels then what we do is that for channel number 1 to the number of output channels we do the windowing and overlap and add and this extracts the PCM samples, so store the samples in the PCM output buffer and then copy the down mix buffer to the delay buffer. This is a kind of a pseudo code that one forms.

(Refer Slide Time: 52:08)



Of course whichever routines we opened we have to meticulously close everything so this we have to take care; about the matching of that part. Now forget about the down mixing part which I will be explaining you separately. That is why this down mixing and delay buffering this may not be very clear to you immediately. But all that I want to say is that the basic idea is to encode the mantissa and the exponents for each of the channels for each of the audio channels and basically the synchronization timing that one follows for the AC-3 is something like this that whenever we receive the first two third of frame first two third of the AC-3 frame we need not have to wait for the complete AC-3 frame to be received for decoding; just the first two third of the frame is enough.

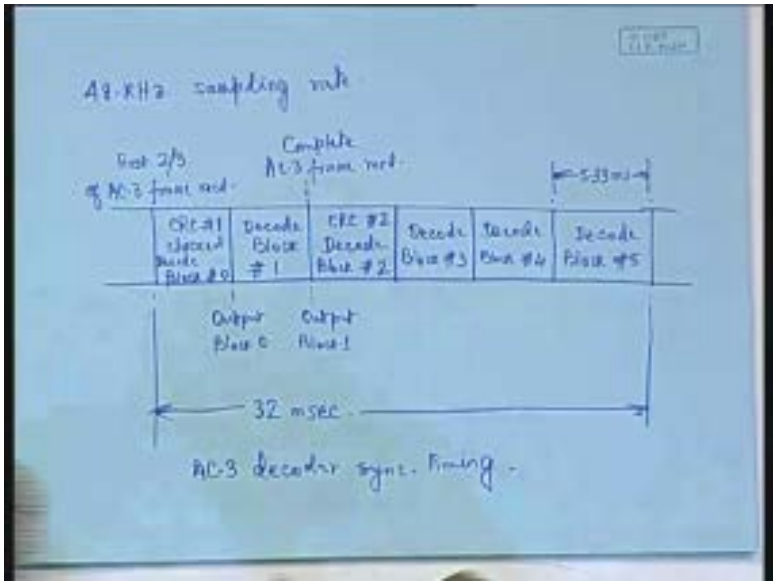
Means you see, if you look at the AC-3 frame structure it is the CRC 1 followed by this synchronization information followed by these blocks and the last one is the CRC 2 information so it leads us to think that after the CRC 2 after we receive CRC 2 then only we should start the decoding but that is not mandatory; we can start the decoding when first two third of the AC-3 frame is received and they are the CRC 1 and block 0, audio block 0 information that is enough.

Therefore, what we do is that first we check the CRC number 1, decode the audio block 0 then we also decode the audio block 1 and by then the rest one third of the AC-3 frames will be received and then we can look for the CRC 2 and decode the remaining audio blocks. So the synchronization timing goes like this. this is assuming a rate of 48 kHz sampling rate so with 48 kHz sampling rate the timings are like this that the total time for a frame happens to be 32 milliseconds and this is the instant where first two third of AC-3 frame is received. Now, totally we have here, from here up to if the audio block five ends over here then we have total 32 milliseconds (Refer Slide Time: 55:41).

Then after the first two third of the AC-3 frame is received we do the CRC check number 1 so CRC number 1 is checked and also block zero is decoded. So decode block zero so decode block zero and then next we decode block 1 and then next we check CRC 2 so CRC number 2 is checked and then we decode block 2 which means to say that when CRC 2 is checked that means to say that at this instant you must have the complete AC-3 frame received. Then this is decoding of block number 2.

Now here after decoding block 0 you can output block 0 so this is output block 0 then after decoding block one here you can output block 1 and so on. So block 2 then comes the block 3 so this is decode block 3, decode block 4 and then decode block 5 and here we have a total time of 5.33 milliseconds 5.33 milliseconds. You see this is one sixth of the total time roughly; every block time is one sixth of the total time. Then you can see that that is so because 5.33 if you multiply by 6 then it is 32 the total frame time and up to the two third if you receive then the rest of the decoding can commence. This is the decoder synchronization timing so this is the AC-3 decoder synchronization timing.

(Refer Slide Time: 58:55)



So we will continue with the discussion on the AC-3 encoding in the next class; till then thank you.