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

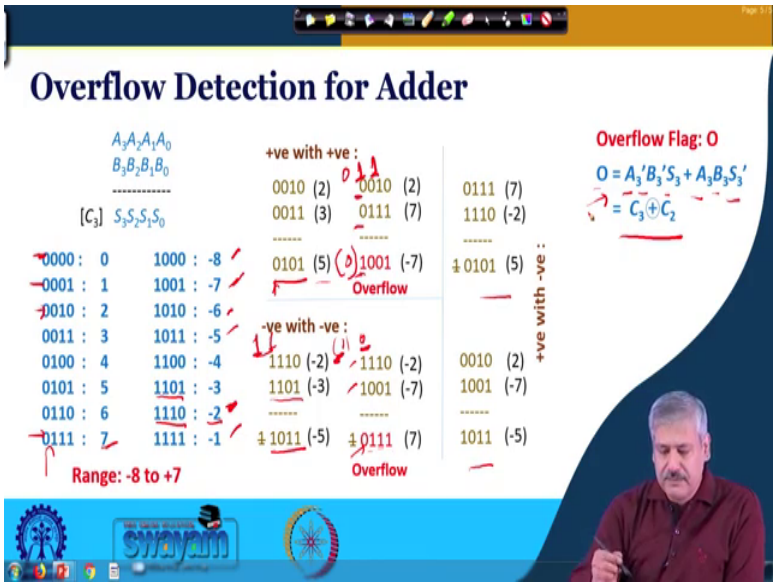
Lecture – 25
Overflow Detection and BCD Arithmetic

Hello everybody, we are discussing arithmetic building blocks. And in today's class we shall discuss Overflow Detection in adder and also how to perform BCD Arithmetic.

(Refer Slide Time: 00:27)

So, we shall see BCD adder and subtractor circuit and also here will have some idea about the underlying concept theoretical concept, that is 10's compliment. And, we shall also look at have a look at what would have been the case, if we had employed 9's complement for BCD subtraction. And, during that course we shall also look at the equivalent similar relationship for binary subtraction that can employ 1's complement. We have to use 2's complement subtraction; so, what could have been the case we had used 1's complement subtraction.

(Refer Slide Time: 01:07)



So, first we discuss the overflow detection. So, for that we again deficit the number representation; so, in this particular case we are looking at 4 bit number representation. So, that 4 bit considers also the sin bit ok. So, when we consider sin bit with 4 bit we remember that in 2's complement format we could represent a number from minus 8 to plus 7 ok. We remember that and how the numbers were there. So, these are the numbers so, 0 0 0 0 was 0 0 0 0 1 was 1 0 0 1 0 was 2 so, up to 0 0 0 1 1 1.

So, these basically behaves like a sin bit that was 7 and 1 triple 0 was minus 8, 1 0 0 1 was minus 7, 1 0 1 0 was minus 6, that way 1 1 1 1 was minus 1 we remember that, isn't it. So, if we are now looking at the possibility of the overflow; so, that occurs when a positive number is added with a positive number and the value of that is more than 7 ok. So, it goes out of the range and a negative number is added with a negative number. So, and the output becomes more than minus 8 I mean less than minus 8 from the negative number representation concept.

The magnitude is more than 8 so number is more than minus 8 ok. But, when you are adding a positive number with negative number the result I mean, if the positive number and negative number to begin with are within range. So of course, the result will be within the range because 1 is getting subtracted. So, magnitude will be within the range ok. So, we do not have a case of overflow when we are having a positive number added

with a negative number. So, positive number added with negative positive and negative number added with negative, there could be possible cases of overflow..

If we look at examples: so, 0 0 1 0 and 0 0 1 1 if we add them up simple binary addition. So, 0 1 0 1 sin bit is positive absolutely no issue 5 is within the range so, there is no overflow. So, whereas, when you are adding 2 with 7 ok; so, 0 0 1 0 0 triple 1. So, if you add them up you see 1 0 this is 1 1 1 0 here there is a 0, then there is a carry, then this 1 and 1 0, there is a carry ok. And finally, this is 1 and there is no carry, of course there is no carry here is it clear right; this C 3. So, there is no carry, that is what we see 1 0 0 1. Now, this is if you look at actually the meaning is 9 of plus 9 we know addition of 2 and 7, if you had been in you know binary coded this was not a sin bit right.

But, it would have been represented at 1 0 0 1, but we know that this is a a number representation where, the maximum positive number is plus 7 and 1 0 0 1 is over here it is minus 7. So, it is out of range so, there is overflow. Now, we look at the example of negative number with negative number. So, minus 2 and minus 3 say let us look at that. So, minus 2 is 1 1 0 from here you can see this is minus 2 and minus 3 is 1 1 0 1 you add them up. So, this is 1 this 1 1 1 0 right there is a carry over here ok. And, then 1 1 1 there is a 1 and there is a carry over here right and that carry comes here and we ignore that carry and this is 1 0 1 1. We see the result is 1 0 1 1 is minus 5 which is correct which is a valid number valid number and the result is also correct, it is within the range and if you have minus 2 minus 7.

So, what we see minus 2 is 1 1 1 0 and minus is 7 is 1 0 0 1 we have already seen that right. So, if you add them up 1 right this is 1 this is also 1 1 1. So, there is no carry here 1; so, that is basically no carry you can see no carry. Why I am writing this that would be cleared later and this is 1 1 0 and there is a carry. So, this is a carry over here, I can the ignore the carry 0 1 1. So, what is it? It is a it is plus 7 which is of course, not the right thing ok. So, this is the case of having a overflow. The answer would have been minus 9, but it is minus 9 is out of range minus 8 is the maximum that you can have ok. And, these are examples of positive with negative we can see that there will never be a overflow because, the magnitude is always within the range right.

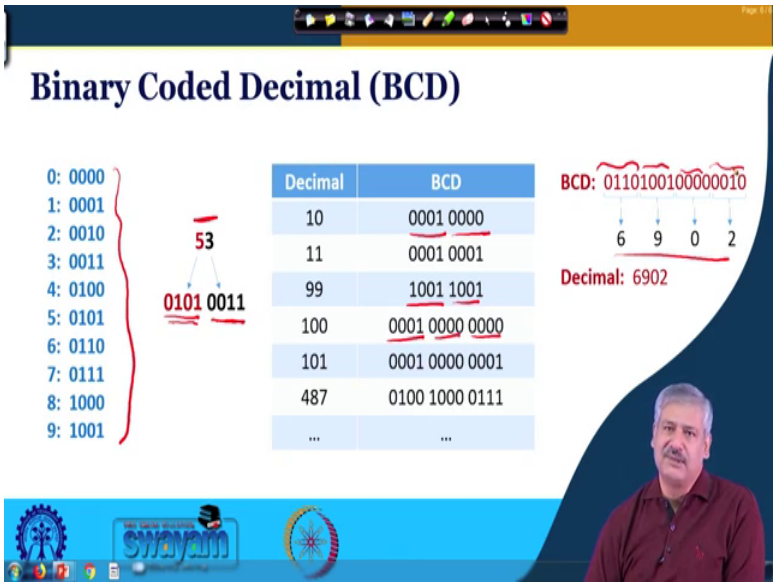
Now, how with then can we can detect if we have you know a adder you know fade with this numbers, then how we can detect with just can develop a simple logic circuit which

will take input from this adder the sum and carry bit that is getting generated right. If it is a 4 bit additions S naught 2 S 3 that will get generated and carry will also be generated right. And, intermediate carry if it is available C 2 C 2 C 1 C naught will also get generated. So, what we can see the overflow occurs when these are the 2 cases when the sin bit here, if the is 0 sin bit here is 0 and sin bit here which is S 3 is 1. So, this is a case A 3 prime B 3 prime and S 3.

So, this is the case when it occurs, you see in the other cases it will not occur right. When if S 3 is 0 you will see that this will not generate this output to be 1 and the other case when it, occurs the sin bit here is 1 1 and this sin bit that is getting generated S 3 is opposite right. So, these are the cases when A 3 B 3, these two are 1 and S 3 is 0. So, we are taking S 3 prime. So, we add them up then we can get it right. And, if we have access to inter intermediate carry C 2, C 3 is the final carry that is getting generated right. Then we can see that whenever there is overflow; wherever there is overflow, the C 3 is the final carry and C 2; the carry just before that C 3 and C 2. Just before that they are I mean if 1 is 0 another is 1.

So, if you take XOR of that you can get the overflow detected right. For other cases you can see this is 1 1 and similarly this is 0 0 you will see that there is no, such case that is generating the carry for C 2 and C 3. So, this is another way just having a XOR gate. So, if you have got access to C 3 as well as C 2 otherwise A 3 B 3, those are the input data and S 3 is getting generated you can put them together with a logic relation like this. And, you can get overflow detected is it clear right and if there is a overflow we have to take remedial action.

(Refer Slide Time: 09:15)

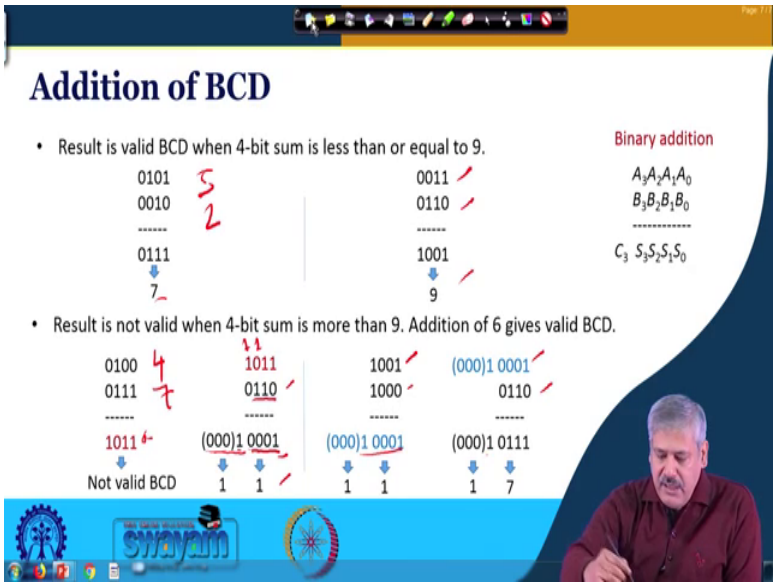


Now, we will look into BCD addition and subtraction. So, before that again we look at how BCD numbers are represented. So, BCD numbers are binary coded decimal, we have seen you know BCD decoding and other things before. So, we are already familiar that BCD numbers are from 0 to 9. So, this is the decimal number, decimal representation you have got digits from 0 9 right and corresponding binary codes are 0 0 0 0 to 1 0 0 1.

So, in BCD coding the number more than that is valid and if that is called 1 digit. If you have got a 2 digit decimal number say 53, how do you represent it in BCD. So, 5 we represent as 0 1 0 1 fine and 3 is represented as 0 0 1 1. So, BCD coded 2 digit decimal number 53 53 will look like a 4 digit binary number 0 1 0 1 0 0 1 1 0 1 0 1 0 0 1 1. If it is a binary coded decimal the corresponding decimal is 5 and 3. So, group of 4 you have take. So, if you we look at some example so, 10 so, 1 and 4 0's right 99 1 0 0 1 1 0 0 1 100 0 0 0 1 0 is 0 0 0 0 and 0 is 0 0.

So, this is your 12 digits will be required to represent 100. Similarly, if you are looking for you know 6902 a 4 digit decimal number getting the presented. So, 6 will get converted to 0 1 1 0, 9 is 1 0 0 1 0 is 4 0's and 2 is 0 0 1 0. So, this is the way you represent binary coded decimal and you are talking about addition and subtraction of these kind of numbers these kind of number representation. So, how do you do that?

(Refer Slide Time: 11:25)



So, first let us look at some of the theory that goes into the you know development of the circuit for BCD addition. First of all we see that the result of BCD addition is valid when the sum is less than or equal to 9 because, up to 9 you have got valid BCD representation. So, if you are adding 5 sorry, if you are adding 5 decimal which is represented in BCD as 0 1 0 1 with 2 which is represented as 0 0 1 0 then we will get 0 1 1 1. So, that is 7 which is absolutely no issue right. Similarly, 3 with 6 you get 9 absolutely no issue.

But, problem occurs when the number sum of this number is more than 9 ok. For example, over here so, 0 1 0 0 that is your 4 and this is your 7 ok. When you add it up 1 0 1 1 which is you know as a binary you know number 1 0 1 1 you know if you look at the weights. So, this is your 8 2 1 11 that is fine, but 11 is not a valid BCD, I mean decimal representation. There is no symbol for that and we are looking at symbols which is 0 to 9.

So, up to 1 0 0 1 you can have a number presented here a group of you know 4 digit. So, this is not a valid BCD ok. You look at another example say this is your 9 right and this is your 8, if you add them up you get 1 0 0 0 1 and there is a carry ok. So, this is again something which is goes the going out of the range the when you get added up it is 17, 17 is not what will get I mean within a decimal representation as a symbol ok. So, these are the cases when we are having issue, when the number is going more than I mean

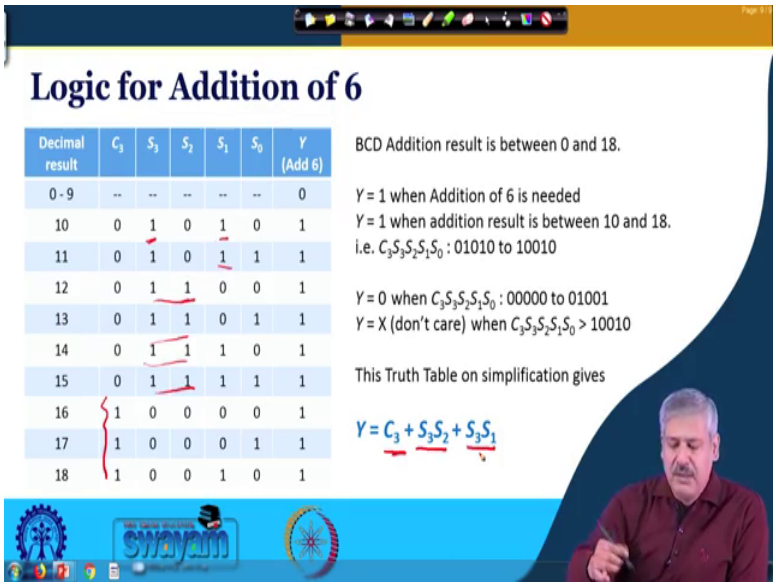
results becoming more than 9. And, we can see a solution can be obtained by a simple addition of 6 in each cases ok, let us see how it happens. So, this was 1 0 1 1.

So, these result when it is more than 9 you have to develop a logic circuit that we add 0 1 1 0. So, if we add it up what will happen? So, this is 1 this is 1 1 0 and that is a carry which is added with 1 is 0 there is a carry with 1 0 and this is 1 ok. So, this is now a proper BCD coded representation of the result. And, if there is 1 1 so, you put leading 0's which does not alter the value. So, this is 0 0 0 1 1 BCD number right. So, corresponding decimal is 1 0 0 0 1 is 1 and this is another number; so, 0 0 0 1 1 1.

So, if you take it to a display device through this 4 digit which converts to a 7 segmented display driver and all. So, the display device will show 1 and 1 which is 11 and which is the correct result is it clear right. So, similarly in these case this was 17, I mean in if you look at in corresponding binary weighted value right and if add it with 6 right. So, this is 0 1 1 1 that is 7 and this is 1. So, we can check for every such cases it will happen in that manner right. Whenever the number increase is more than 9 you add 6 you will get the corresponding a correct representation of in the BCD format ok.

So, this is so, we will keep in our mind and try to develop the appropriate logic function for this.

(Refer Slide Time: 15:45)



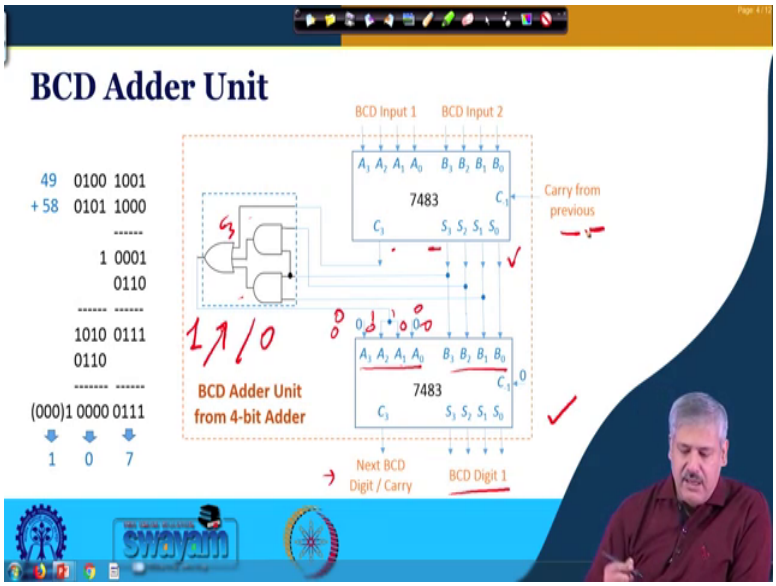
So, what is that logic because it will be part of the circuit right you have to develop the corresponding adder BCD adder. So, we see that for decimal result I mean when we are doing the representation, when it is remaining between 0 to 9 ok. So, the addition whether it is required or not addition of 6 is required or not if we take this as output Y ok; so, Y will be 0 that time the 0 means no addition is required; no addition of 6 is required ok. Now, instead of result being you know 0 to 9, if it is 10 11 12 and up to 18, it is possible 2 numbers getting added is 9 and 9 that is the maximum.

So, in each of these cases these output is 1 because, addition of 6 is required and what are the corresponding representation for at that time. These 10 11 12, if you have the adder for some bits are S naught to S 3 and the carry bit is C 3. These are the corresponding representation 0 1 0 1 0 that is your 10 and 1 0 0 1 0 that 0 18 ok. So, there can be you know with 5 such bits 5 such bits you can have 32 different possibilities. So, possibilities more than 18, 19, 20 etcetera that will never occur if the input digits are you know BCD ok, because maximum 9 and 9 can be given as input. So, we do not care. So, those values are do not care, I mean you do not really bother about what will happen if we in this case of if 19 is presented, is it fine right.

So, these 2 table we need to convert to a equation and we can see that this 2 table, if you can just simplify using standard process algebraic or kernel map or QML algorithm with them you can see that a relationship something like this emerges. So, if you just quickly have a look at so, C 3 means right. So, whenever C 3 is there right you need to have the Y flag the addition of 6 you know high, isn't it. Because, that is C 3 means the number is 16 or more S 3 S 2. So, this is your S 3 and S 2. So, S 3 and S 2 means, if both of them are there; so, this is the case these are the cases.

So, S 3 stands for 8 and S 2 stands for 4 so, 8 plus 4 12. So, 12 and more right these are the cases of course, at that time you are having a value which is high and S 3 S 1. So, S 3 occurs; that means, it is 8 and 1 S 1 is also there; that means, 2 right. So, you can see all those cases S S naught there or not there, the number is more than 10. So, these are the cases you can see which is perfectly feeds these equation and otherwise as I said you can do the minimization by standard process taking the mean terms and then minimising by Boolean algebra or through k m kernel map or QML algorithm and this is what you get. So, basically we have to it realize these from materialogic in hardware and then the job is done ok. But, the how the overall circuit will look like let us see.

(Refer Slide Time: 19:33)



So, this is how the overall circuit will look like for the adder BCD adder. So, here what you have got, you have the first adder 7483 the 4 bit adder generating is not is S S naught S naught to S 3 that what you see. And, this is your C 3 right and if these BCD adder takes you know carry from previous stage you will take it will take. So, that carry will be used here and this is the logic circuit to generate your whether the 6 will be added or not. So, this is taking C 3 this is your C 3 getting connected here, this is your C 3 right, then S 3 and S 2; so S 3 and S 2 and S S 3 and S 1 ok.

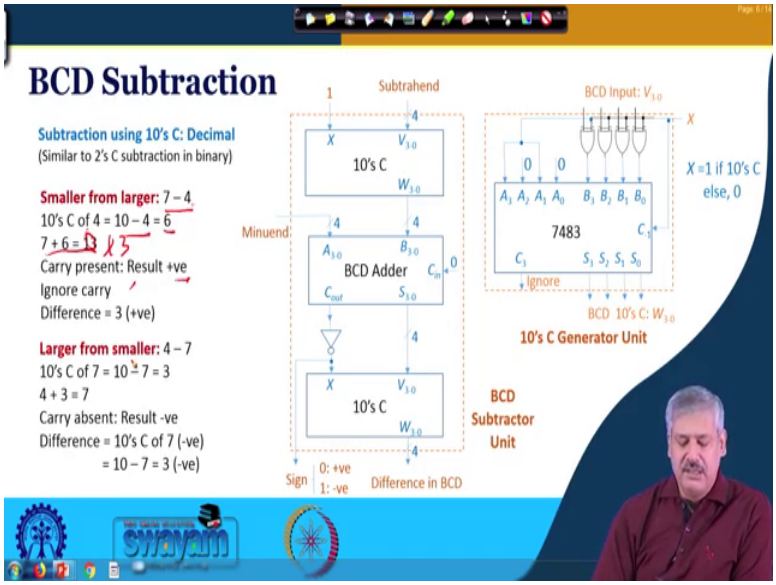
So, this AND gate S 3 S 1, this is S 3 S 2 and finally, it is odd. So, if it is 1 means 6 need to be added and if it is 0; that means, no addition is required; I mean no addition required means here in this particular please since, you have to put a adder; so, basically 0 is getting added ok. So, this number the number that was added before, it is now brought forward to the next adder right next 4 bit adder and here right the number is when this is Y is high. So, 0 1 1 0 A 3 is 0 A 2 is 1 A 1 is 1 and A 0 is 0 right. So, what does it mean at that time, then 0 1 1 0 that is 6 is getting added. And if this is 0 so, this will be 0 0 0 and 0 right. So, no addition I mean no number is actually getting added.

So, the final thing will be the BCD digit and if there is a carry right. So, that carry will be displayed with leading 0's we had shown otherwise, if there is another BCD number getting added like 53 example; I had given before then this carry will be taken their to bring next BCD adder we make. So, there it will be given as input is it clear. So, this is

how a 1 unit of BCD adder will act and we look at one example here. So, 49 is getting added with 58. So, 0 1 0 0 1 0 0 1 and 0 1 0 1 1 0 0 so these are the corresponding you know this is represented in BCD 4 and 9, 5 and 8 and when we add it up ok.

So, there is a carry here what is generated. So, the number is more than 9. So, 6 is getting added. So, this is the corresponding number that is 7 will be there and this carry is now getting added with the next numbers 4 and 5 ok; so, 4 and 5. So, though it is corresponding number and 1 is getting added 1 0 1 0 we get direct that again is not a direct BCD ok. So, we add 6 here. So, you get 0 and a carry is generated as and then you put the letting 0's. So, the result is none 0 7 ok. So, if you look at in decimal it is also in 1 0 7 you can add it up then see.

(Refer Slide Time: 22:57)



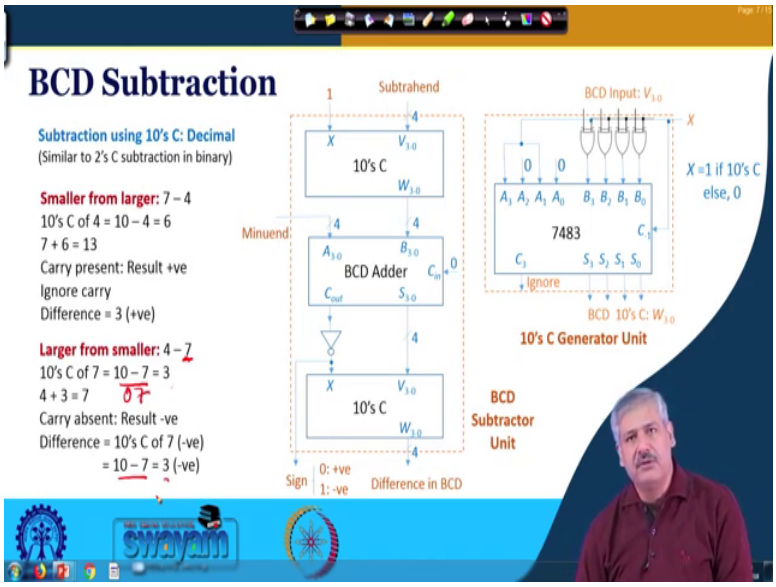
Now, BCD subtraction; so, to do BCD subtraction we shall look at what is known as subtraction by 10's compliment. So, we have done subtraction by 2's compliment in binary it is somewhat similar. So, what is it done here? So, if will considered you know here subtracting a smaller number from a larger number so two possibilities smaller from larger, larger from smaller ok. So, say the example is of the 7 minus 4. So, what we do first the sub trend the where your done 2's compliment, here we shall do 10's complement ok.

So, 10's compliment is 10 minus the number is we will give you the 10's complement of that and that is the negative number of you know in that sense. So, this will give you 6

right then you add it up 6 7 plus 6 it is 13 right. And, if carry is present the result will be positive you can see the similarity between 2's complement addition of a negative number ok. So, you ignored the carry and difference is without the carry you remove the carry sorry 13.

So, remove the carry so 3 so, the answer is 4 positive. So, 7 minus 4 is positive 3 ok. And, had it been 4 minus 7 how would how would that would have been the case.

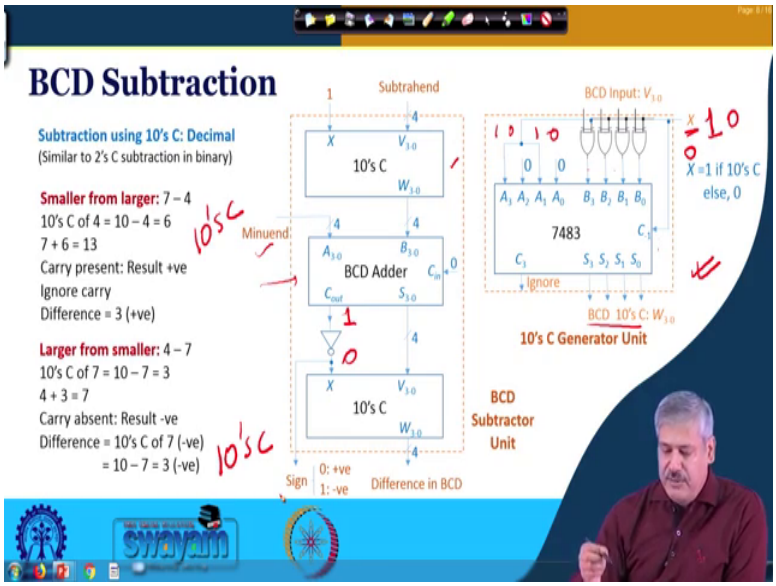
(Refer Slide Time: 24:25)



So, will take 10's compliment of 7 which is 10 minus 7 is equal to 3 4 plus 3 is 7 and when you see there is no carry. So, basically 7 is 0 7 right. So, carry is absent means result is negative, again you can see the similarity with 2's complement you know subtraction. So, now the difference the result already in that case was in 2's complement from if it is negative. So, we have to do another 2's complement to get the original I am the magnitude of the result in proper binary coded representation ok.

So, in this case you will be doing in this case when it is negative you if you want to know the magnitude so 10 minus 7 10's compliment of 7. So, 10 minus 7 is 3 right. So, the answer is negative and the value of it magnitude of it this 3; so, minus 3 ok. So, this is how you can perform 10's complement subtraction for decimal numbers and now we come to the representation of the decimal number in BCD and how the circuit will look like. So, to do that we see that we would require one 10's compliment generated here.

(Refer Slide Time: 25:43)



And, if the result is negative another 10's compliment generated here ok, if the result is possible is not required ok. But, when you at looking about the hardware it will be in place so, whether you use it or not that that is a separate thing. So, first we look at how to generate 10's complement ok. So, the circuit over here you can see can generate 10's compliment, how does it do. So, it is doing it by in this case this is your 1 0 1 0 when you are looking for 10's compliment ok. So, this will be 1 so, this is 1 0 1 0.

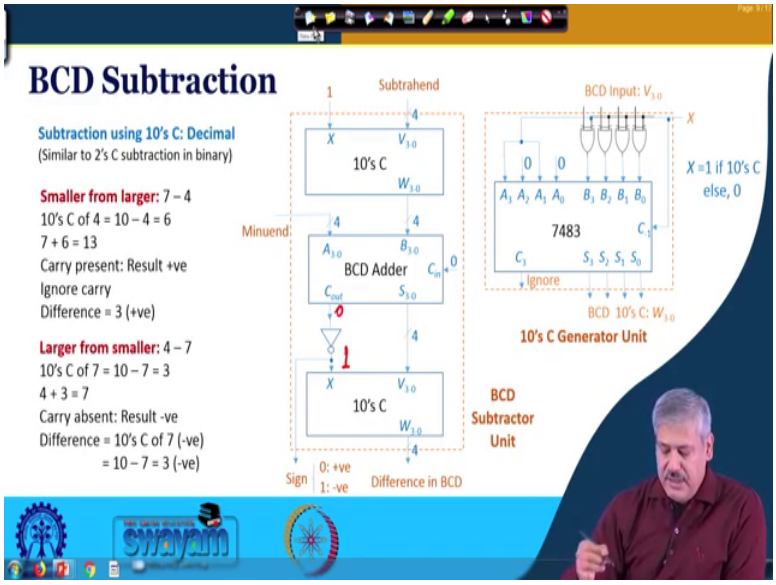
So, 10 and the number for which you are looking for 10's compliment that will be that you want to subtract ok. So, subtraction we know already how to do it using a 4 bit adder. So, this is a bank of XOR gate. So, if it is 1 the inversion thing will be coming here inverted thing right. So, that is 1's compliment will come and then this 1 means this is the carry will be getting added. So, that is giving it the 2's complement and together when you do it then you get 1 0 1 0. This is value will get the corresponding binary coded decimal part of it in the corresponding negative number of or the 10's complement of the original decimal number ok. And, if there is a carry you have to ignore it right, is it clear.

And, if X is equal to 0 what will happen the same number will get passed the same circuit can, in that case you can do the addition also it is for subtraction it will be recurring to have this particular value to be 1 ok. So, this is the 10's complement generator. Now, during subtraction how does it work it out ok? So, the subtrahend is put

here right and minuend the number from which it is getting generated; so, that will be put here. So, we are doing BCD addition BCD addition you have already seen before right.

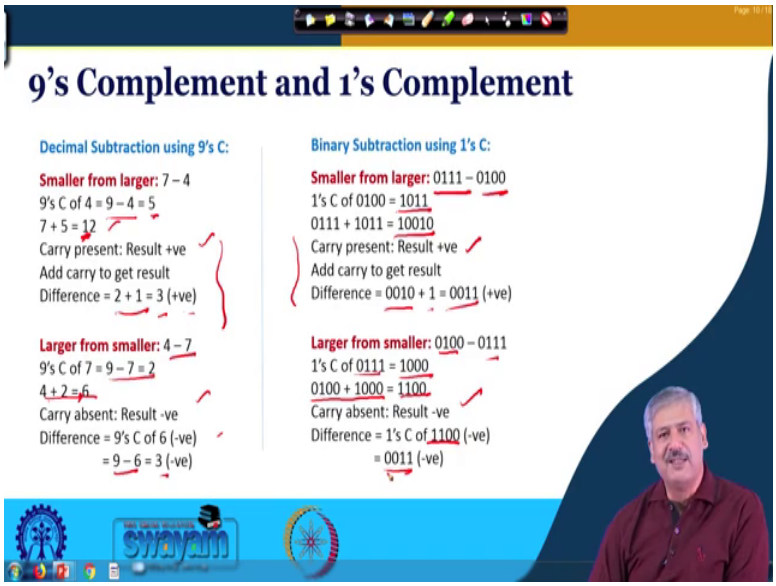
And, if BCD addition generates carry then there is a result is positive and we do not need to do anything, I mean you have to just take the result as it is right. So, if it generates carry so, then this output will be 0. So, these output 0 means here this is 0 right. So, basically you are not do anything you are that number is getting passed here. And, and here the sign you can see that to be positive is it fine, and what happens when the number is when these number is this carry out is 0 that means the number is negative right.

(Refer Slide Time: 28:53)



So, at that time this will be 1 and we need to take 10's compliment of this. So, this is what we when you do then you get the 10's complement of it, is it clear? So, this is 1 BCD subtracted unit.

(Refer Slide Time: 29:19)



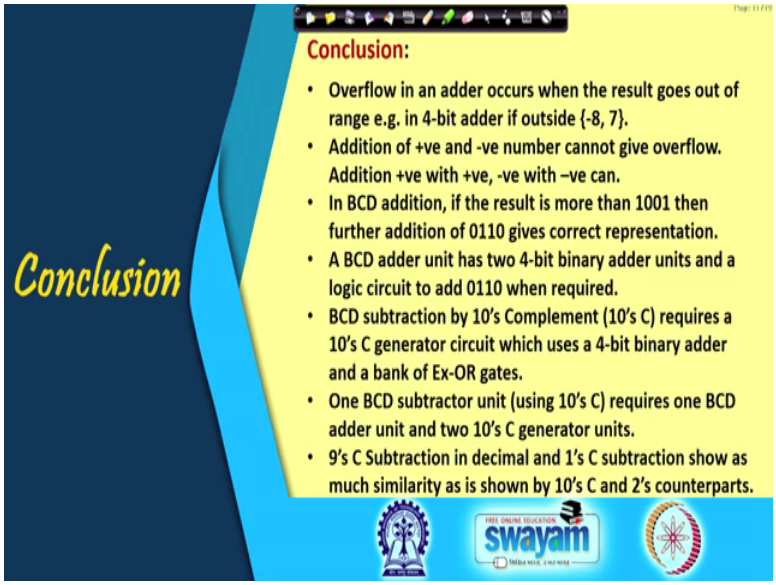
And, to end we shall have a quick you know understanding of what is 9's complement and 1's complement, I mean in decimal system and binary system and how it is useful in subtraction. So, in a 9's compliment what we do actually we subtract the number from 9 not from 10 ok. So, for the same example of 7 minus 4 9's complement of 4 is 5 9 minus 4 5; so, 7 plus 5 is 12. So, it carries present the result is positive and at that time we have to do, what we have to do this carry it is also often called sometimes called in around carry.

So, this carry would need to be added to get the result, if carry is present carry need to be added to get the result. So, 2 plus 1 3 that is positive and because of the presence of the carry it is positive and the value is 2 plus the carry that is 3 ok. And, the other case when 4 is subtracted, 7 is subtracted from 4. So, 9's compliment of 7 is 9 minus 7 it is 2 ok.

So, 4 plus 2 is 6 and carry is absent means result is negative understood. So, here the a carry was present here this 6 there is no carry it is 0 I mean; that means, it is 0. Then the result is in 9's complement you have perform another 9's compliment to get the magnitude of the negative number that is given the final result. So, 9's complement of this 6 is 3; so, the answer is minus 3. So now, if you look at 1's compliment subtraction for binary numbers so, this is your 7 and this is your 4. So, 1's complement we have seen it before basically it is just inversion of it ok.

So, 1's complement of this 0 1 0 0 is 1 0 1 1 you add them up 1 0 0 1 0 carry is present so, result is positive. And, what you have to do in 2's complement it was result was like that in this case this carry need to get added, the way we have done here for 9's compliment 1's complement the carry need to get add get added. So, 0 0 1 0 added with 1. So, this is 3 and the result is positive and larger from smaller. So, this is 4 minus 7 case. So, 0 1 1 1 0 triple 1 1's complement is 1 triple 0; so if you add it up 0 1 0 0 with 1 triple 0 you get 1 1 0 0. So, the carry is absent. So, result is negative and to get the actual difference you have to take 1's complement of that. So, you have invert these 1 1 0 all the bits so, 0 0 1 1 so, that is 3. So, that is minus 3 that is your answer.

(Refer Slide Time: 32:35)



So, with this we just look at the summarizing points of this particular class. Overflow in an adder occurs when the result goes out of range for 4 bit cases it is minus 8 2 7 4. You know 8 bit cases and other cases you have to have corresponding understanding of the overflow and the logic relation that we will get will be accordingly take input from the bits which is closer to the which is designated as sin bit or carry of the previous one and the current one.

I mean which is there in the pacific with the sin bits ok. And, in BCD addition if the result is more than 1 0 0 1 then further edition of 6 that is 0 1 1 0 is required and BCD adder will required to 4 bit binary adder and logic circuit to decide when this 6 is getting

added. And, BCD subtraction will required, 10’s complement generator circuit and normal BCD adder.

So, together two 10’s complement generator circuit and 1 BCD subtractor you can do BCD subtraction ok. And, 9’s complement subtraction will and 1’s complement subtraction in decimal and binary respectively they have got similarity. And, we have seen examples and how to do that and you can convert that to circuit with you had known for 2’s complement and 10’s complement.

Thank you.