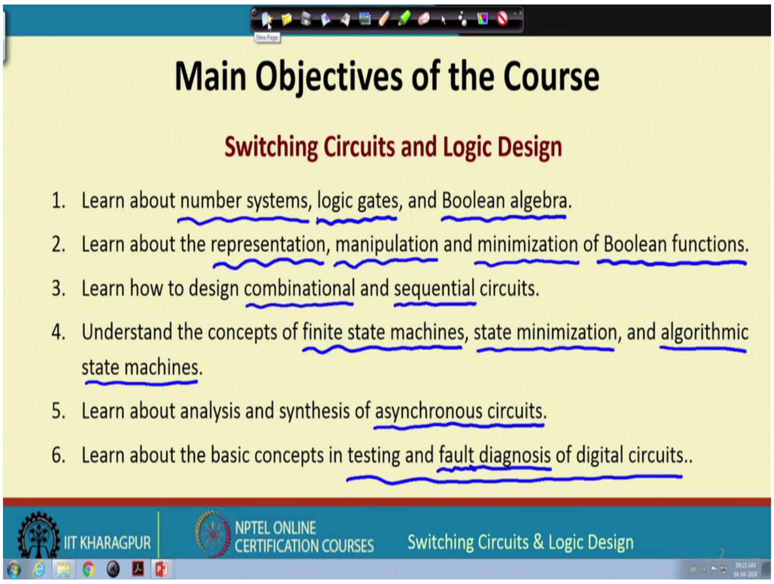
Switching Circuits and Logic Design
Prof. Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 01
Introduction

So, I welcome you all to this NPTEL MOOC course on Switching Circuits and Logic Design. Now, as you may be knowing that this course is primarily targeted to the undergraduate students in their 2nd and 3rd years primarily in the disciplines of computer science and engineering, information technology, electronics engineering and of course, also electrical engineering. Before I start I would like to summarize the topics or the coverage of this course which may help you in understanding the path that we are trying to follow.

So, in our first lecture here introduction I shall be introducing you to this course and a few basic concepts shall be discussed.

(Refer Slide Time: 01:17)



Firstly as I had said let us briefly look into the main objectives of the course and what are the things that we are expected to cover here. You see of when we talk about switching circuits and logic design the basic thing that we need to understand is something called number system; because whatever you do whatever design we talk about we shall be working with some kind of numbers.

There are various ways in which you can represent numbers as we shall be discussing over the next few lectures. So, the first thing that would be talking about at the number systems ok; number systems are basically the way numbers are represented in a computer system or in any digital circuit in general. So, we shall see that implement these number systems we need something called logic gates.

So, we shall be talking about different kinds of logic gates; how they work and how they can be implemented. And finally, we should be looking at some kind of formalisms called Boolean algebra where we shall be looking into the theory behind the design of digital circuit we can see. There is some kind of an algebra or set of mathematical rules using which formalize we can design; we can optimize various kinds of functions that you want implement and we can realize them using the gates that we are talking about ok. Then Boolean algebra talks about Boolean functions, various functions that we want to implement let us say we want ye implement an adder or subtractor or comparator various kinds of this kinds of functionalities may be there.

So, when we talking about Boolean function there are several things how to represents the functions, there are various ways we shall be talking about how to manipulate this functions in various ways. And of course, when we are talking about you realizing this functions using gates one thing is important called minimization, where we are basically concerned about how to reduce or minimize the number of gates that are required to realize a particular given functionality fine.

And talking about circuits broadly speaking we can classify them into either combinational, and sequential we shall be talking about this. Combinational circuits are those who are whenever you apply some input you immediately get some output, but sequential circuits are those where not only the inputs, but it also memorizes the previous history in some sense and then it will generate some output.

Let me take a very simple example suppose, I have implemented the circuit that counts the number of persons that are entering a room through a door, there is some kind of a sensor in a circuit. Now, whenever there is a person entering the room the light inside the room will automatically glow ok. Now, as long as minimum 1 person is there in the room the light should remain glowing.

So, let us say 5 persons have entered now if 1 person goes out that should not switch off the light. The system should remember that there were 5 persons were entered and only when all 5 have left the room only then the light has to be switched off ok. This is an example of a sequential circuit where you need to memorize or remember the previous states of the system right ok.

Now, these sequential circuits are also sometimes represented as something called finite state machine and when you talk about such finite state machines there are various ways to minimize them. And there is a concept called algorithmic state machine or ASM which can be used to model or build relatively larger systems. We shall also be talking briefly about such ASM modelling which are called ASM charts, how to model larger systems using this formulism ok.

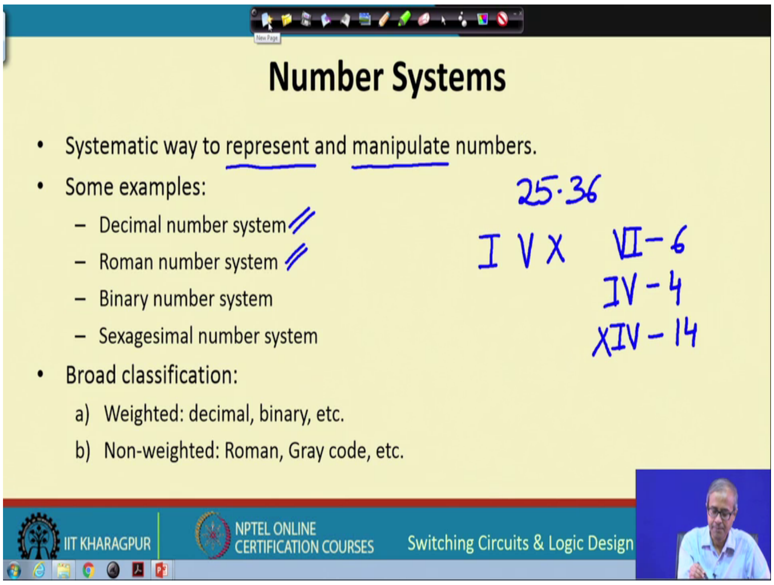
And of course, there is something called asynchronous circuits which are relatively much more difficult to handle and analyze. Well in a normal sequential circuits when you talk about there is a concept of a clock or a periodic signal which synchronizes the activity of the systems. So, whenever there is a clock only then the system will move from one state to the next. But in case asynchronous circuit there is no clock, depending on the inputs the way they changed, the way the inputs arrive the system will automatically adapt itself and change its states.

So, the sequential circuit whenever it is synchronous it is much easier to model because I know the events of blocks when we need to change. But for asynchronous circuits changes may happen anytime depending on the delays of the gates, inputs arriving various kinds of events can be there ok. So, it is much more difficult analysis we shall see this.

And lastly whenever we design something there is always the possibility of some faults appearing in the circuits that can be some defects in manufacturing during operation, there can be some short circuit or some wires might come out. There could be lot of faults that may happen in an operational circuit or a design. Now, lastly you shall be talking about how we can test such circuits and fault diagnosis means how to locate where the fault has occurred. So, we shall be talking about these things, alright. So, let us move forward now.

So, here we shall I mean we have given very rough bird's eye view about what our coverage during the next few weeks is expected to be like. So, we now you have fair idea about the topics that we shall be covering; Some of these are part of your basic undergraduate course curriculum, but few things I shall be discussing which will be at a slightly advanced level which may not be part of your course or syllabus, but it is always good to know about them alright.

(Refer Slide Time: 09:08)



Let us start with the basic concept of number systems. Number systems form the heart of any digital system; before I talk about number system let me very briefly tell you about these circuits or the systems that we are talking about. So, we shall be coming becoming to this later again. Circuits can be broadly classified into two types, one we can call digital other you can call analog.

For digital circuits the inputs as well as the outputs they are discrete in nature, discreet means they cannot assume any arbitrary values. They can contain only some discrete values and these discrete values we typically represent in terms of some numbers like you can say 0, 1, 2, 3 these are the discrete values. But for an analog circuit the inputs can be continuous, let us say we are sensing the temperature of the environment temperature can be anything it can be 25.1 degree Celsius, 25.2, 25.25, 25.225 and so on.

So, there is no discreteness evolved here the values are continuous and those continuous values are fed as input, similarly the outputs that is generated they can also be that kind of continuous of analog. So, when you talk about digital systems most important thing to understand are the number systems. Number system is basically the way to represent numbers and some rules to manipulate numbers.

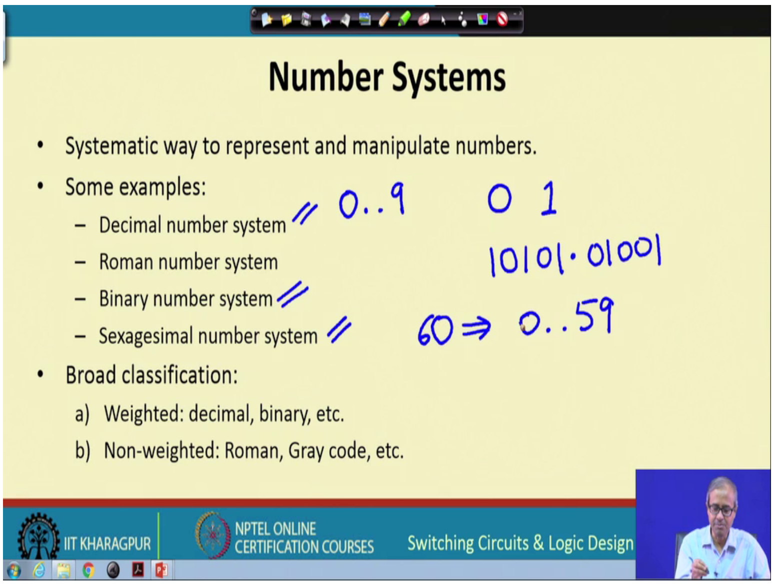
Now, let us talk about some examples decimal number system. Decimal number system is the number system which we are all familiar with like we write numbers like this 25.36, this is an example of a decimal number system

So, in a decimal number system we have the digits 0 to 9, there are some positional weights. So, when you write a number so, we can find out what is the value. Roman number system you are ensure you are familiar with this, here the concept is slightly different.

There are some distinct symbols which represent distinct values like for example, this I represents the number 1 V represents 5 X represents 10 and so on. There are some rules also if I write V followed by I it means 6, but if I write I before 5 V this means 4. So, depending on whether a smaller digit say I smaller right I is 1 is V is 5.

So, if this smaller one appears after the larger one the two are added, but if the smaller one appears before then the smaller one is subtracted from the larger one. So, the rules fairly complicated in that sense like if I write XIV this means 14. So, the rules to find out the values are not as straight forward as the decimal number system, now ok.

(Refer Slide Time: 13:03)

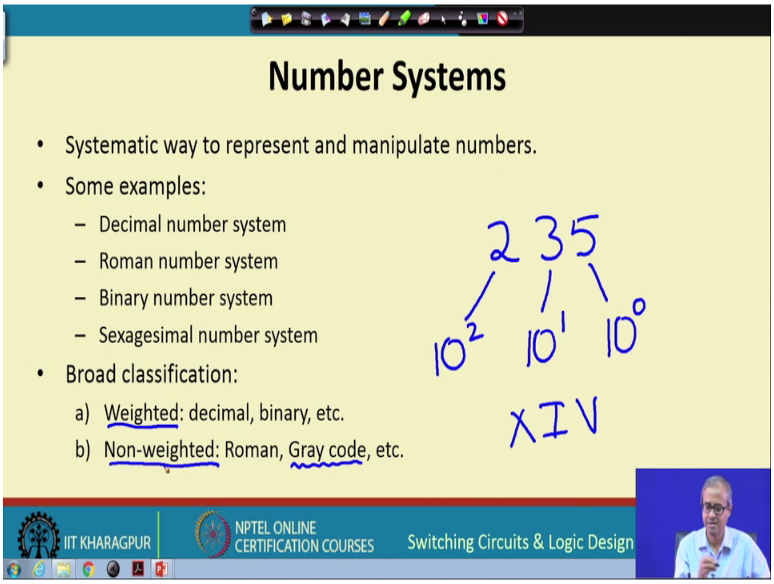


Next let us talk about the binary number system, binary like decimal we shall mostly be discussing about the binary number system. Here you talk about only two digits 0 and 1 and numbers are represented as combinations of 0 or 1 like for example, I can have a number like this point 01001 let us say. This is an example of a binary number. So, we shall be discussing this is in much more detail.

Sexagesimal number system which is one which we all use knowingly or unknowingly you imagine our clock seconds or minutes. So, how do you count? Seconds or minutes we count from 0 we go up to 59 after 59 it again comes back to 0. Similarly minute; for hours it is 12 or 24. Sexagesimal number system means 60 that means, the way we count seconds and minutes, we count from 0 up to 59 and then back to 0.

You think of a decimal number system here we count from 0 up to 9 and then back to 0 there are 10 digits it is called decimal, binary there are two digits 0 and 1. Sexagesimal there are 60 such things 0 up to 59, after 59 again back to 0 ok.

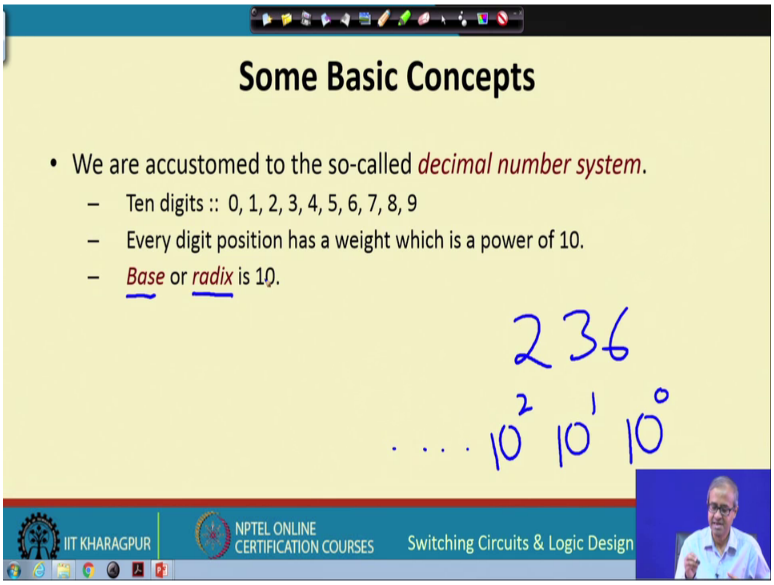
(Refer Slide Time: 14:51)



And these number systems, there is another way to classify based on whether they are weighted or non-weighted. The weighted number systems are like decimal and binary for example, when you write a decimal number 2 3 5 it means that each of the digit position has a weight associated; 5 has a weight 10 to the power 0, 3 has a weight 10 to the power 1, 2 has a weight 10 to the power 2.

So, if we multiply these digits by the corresponding weights and add them up we get the value 235, but you think of the roman number system when I write XIV. So, I cannot identify the weights like this which I multiply and add them of I will get the value. So, these are non weighted course. There are other course also which will see later gray code which is also non weighted code right ok.

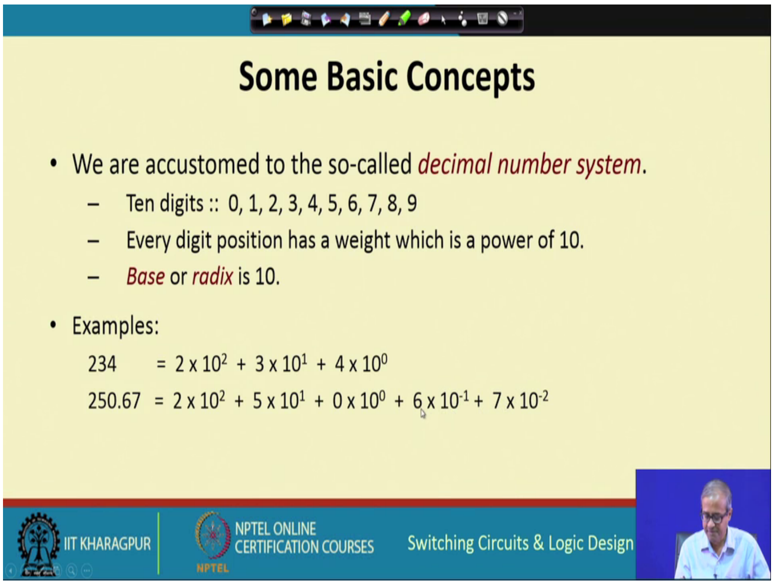
(Refer Slide Time: 15:58)



So, let us move on so, as I had said we are most familiar with the decimal number system where there are 10 digits 0 to 9 and as I had said that every digit position in a number, when I write 236 has a weight and this weights are powers of 10, 10 to the power 0, 10 to the power 1, 10 to the power 2 and so on right.

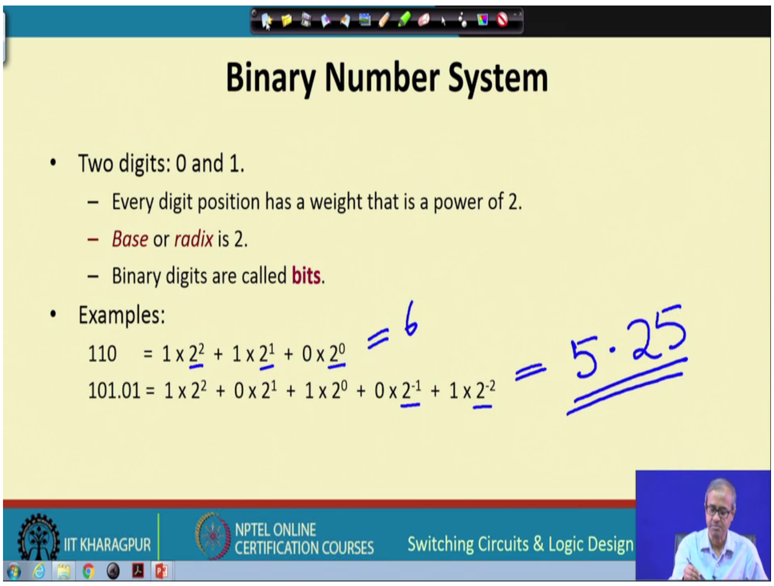
Now, this number 10 is called the base or the radix of the number system, for the decimal number system the base of the radix is 10 fine.

(Refer Slide Time: 16:50)



So, let us take some examples here let us take this the number 230 as I had said; each of this positions have a weight 10 to the power 0, 10 to the power 1 and 10 to the power 2. You multiply the digits by the weights, add them up you get the value. Now, if there is a decimal point the concept is similar, but the only difference is that after the decimal point, the powers become negative 6 into 10 to the power minus 1 sorry then minus 2 and so on.

So, this weights will become negative, but the concept is same you multiply every digit 2 into 100, 5 into 10, 0 into 1, 6 into point 1, 7 into point 01 can you add them up you get the value 250.67.

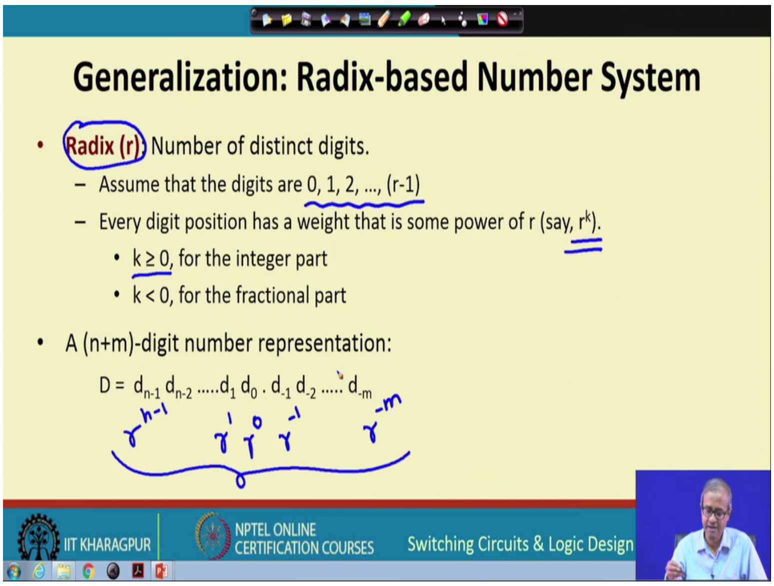


Coming to the binary number system here we talk about two digits 0 and 1 because there are two digit the concept is similar. So, here we talk about a weight that is a power of 2. So, in decimal it was power of 10. So, here the base or radix is 2 and binary digits are sometimes in short called bits. So, a bit represent a binary digit that means, a 0 or a 1, 0 or a 1 is called a bit alright.

So, let us take a examples again suppose, a binary number if I write as 110 following the same rule like decimal we multiply each digit position by a weight which is now a power of 2, 2 to the power 0, 2 to the power 1, 2 to the power 2 which means, 2 to the power 0 means what 2 to the power 0 is 1. This is 1, 2 to the power 1 is 2, 2 to the power 2 is 4. So, if you add 1 into 0 plus 2 into 1 plus 4 into 1 the value becomes equal to 6.

Similarly, for a number with fractional point rule is same 2 to the power 0, 1 and 2 and after the fractional point it will be 2 to the power minus 1, 2 to the power minus 2. So, in this case the value will be you can check 5 point after the point 0 into this, 1 into this so, it is 0.5 and 0.25. It will be 0.25 this will be the value of this number right. So, in binary when you write down the number the value can be calculated like this, following this same rule as in the decimal number system.

(Refer Slide Time: 19:58)

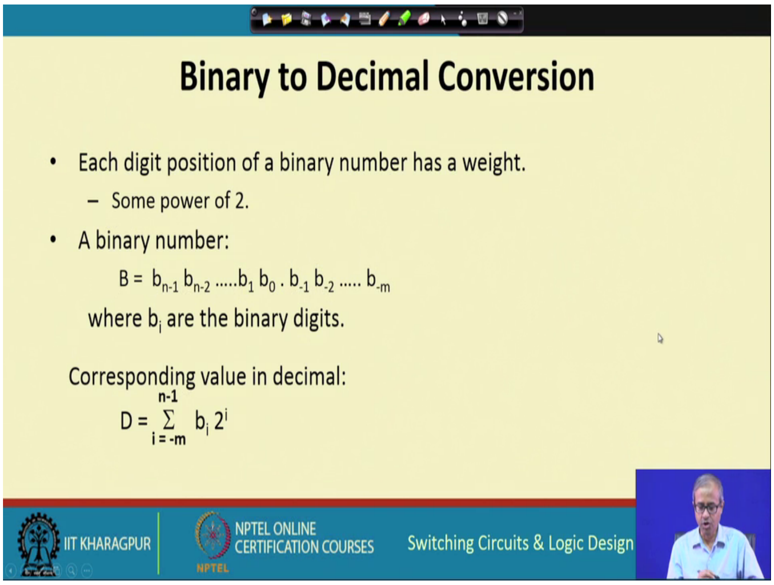


So, let us now talk in general terms if we consider a general radix based system let us call radix r in general for decimal r equal to 10, for binary r equal to 2. Radix denotes the number of distinct digits which let us say its starts with 0 and it goes up to r minus 1.

Every digit position has a weight which is some power of r for the integer part the weights will be this k will be greater than or equal to 0.

Like if this is the fractional part d 0 will be having a weight r to the power 0, d 1 will be a have a weight r to the power 1 and so on; d n minus 1 will having r to the power n minus 1. Similarly, for fractional part k will be negative d to the d minus 1 will have a weight r to the power minus 1 and so on; d minus m will have r to the power minus m. So, the rule is same for a n plus m digit number where there are n digits in the integer part m digits in the fractional part you multiply digits by the corresponding weights add them up you get the value.

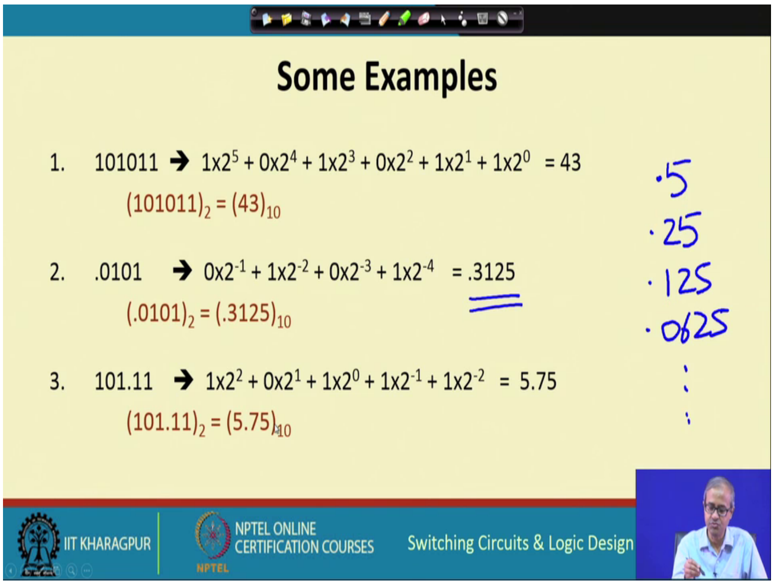
(Refer Slide Time: 21:30)



So, when you talk about binary to decimal number conversion the rule as I had said simple I already mentioned the rule.

Given a binary number so, when you want to convert to decimal multiply each digit position by the corresponding weight which is the power of 2 which can go from minus m up to plus n minus 1 add them up you get the decimal number, ok. This I have already mentioned.

(Refer Slide Time: 22:01)

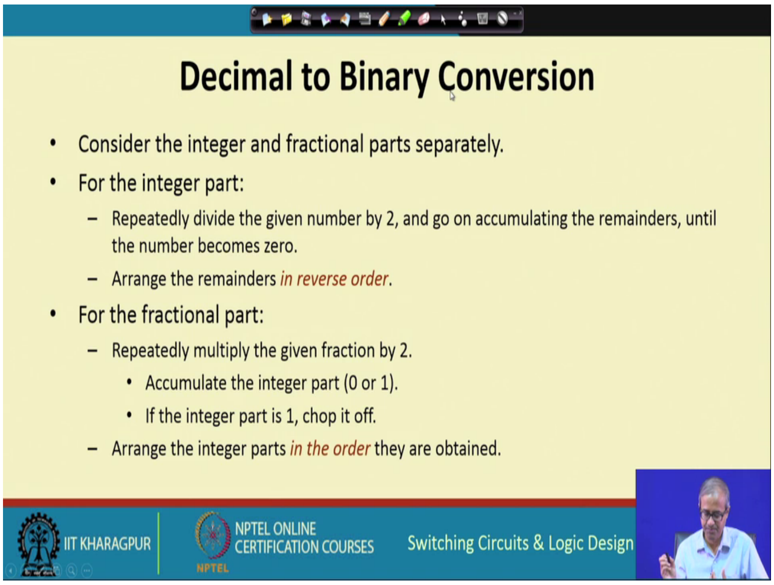


Now, some examples are here 101011 just compute the weights 2 the power 0 1 2 3 4 5 multiply the corresponding digits, add them up you get 43. Now, we write it like this, this number base or radix 2 we use it as a suffix; that means, it is the binary number and 43 10 means it is a decimal number. Take another example a purely fractional number.

So, the weights will be minus 1 minus 2 minus 3 and minus 4 if you add them up the weights are how much 2 the power minus 1 is 0.5 0.25. So, the weight are like this here 0.5 0.25 0.125 0.0625 and so on. So, you multiply the corresponding weights to these digits add them up to get the value right. Take a number with both integer and fraction parts. So, some of the weights will be positive, some of the weights will be negative and you can compute the value.

So, this is the way you can convert a binary number into decimal very simple.

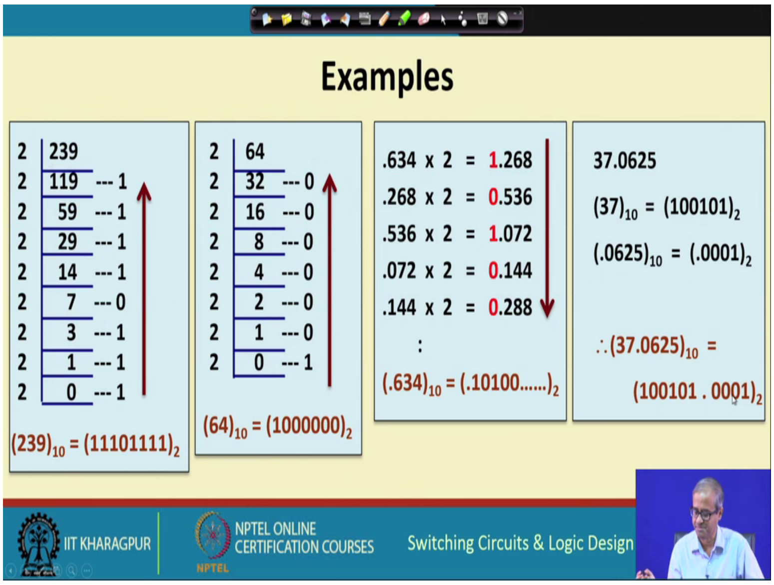
(Refer Slide Time: 23:31)



But when you want convert decimal to binary we need to consider the integer and fractional parts separately. The rules are like this for the integer part I shall show an example, we repeatedly divide the number of 2 and go on accumulating the remainders. And we consider all remainders in the reverse order that will be your binary equivalent.

For the for a decimal number the integer part you have to repeatedly divide by 2 and for the fractional part you will have to repeatedly multiply by 2. So, I will take an example to illustrate and after every multiplication you will have to take out the integer part remember it and the integer parts if you take them in the in the order they are generated that is your binary equivalent ok.

(Refer Slide Time: 24:31)



Let us take some examples.Take the number 239 we repeatedly divide it by 2, you divide by 2 119 with the remainder of 1 you divide again 59 with a remainder of 1 29 remainder of 1 and so on; you go on repeating until this becomes 0.

So, once you get 0 the reminders that you have accumulated you take it in the reverse order 1 1 1 0 1 1 1 1. This will be your binary equivalent right, take another example 64 you repeatedly divide by 2 32 remainders are all 0's 0 0 0 0 at the last stage remainder will be 1, take in the reverse order 1 followed by six 0's this is 64.

Now, considering the fractional part I said you have to multiply by 2 take an example 0.634 you multiply by 2. So, it comes to 1.268 you take out the integer part remove that part 0.268 you take it and again multiply by 2 the next is 0.536 again multiply by 2 it is 1, 0.072 again multiply by 2 it is 0, 0.144 again multiply by 2 it is 0.

So, this integer parts that have been generated after this multiplication you take them in the same order, not in the reverse order like here 1 0 1 0 0 this will be your binary representation. So, it depends how long you want to go the more number of digits you generate, the more accurate your representation will be ok. And if a number has both integer and fraction parts like 37.0625 let us say you do it separate, the integer part you follow this rule repeated division. Let us say this is the binary number, the fractional part you follow repeated multiplication you get this and they will combine the two this is your final answer.

So, with this we come to the end of this first lecture where we essentially talked about the various weighted number systems like decimal, binary and radix r in general and specifically we talked about how to convert decimal number into binary and a binary number to decimal. So, we shall be continuing with our discussion in the next lecture.

Thank you.