Hardware Security
Prof. Debdeep Mukhopadhyay
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture – 20
Hardware for Elliptic Curve Cryptography – II

So, welcome to this class on Hardware Security. So, we shall be continuing with our discussions on Elliptic Curve Cryptography.

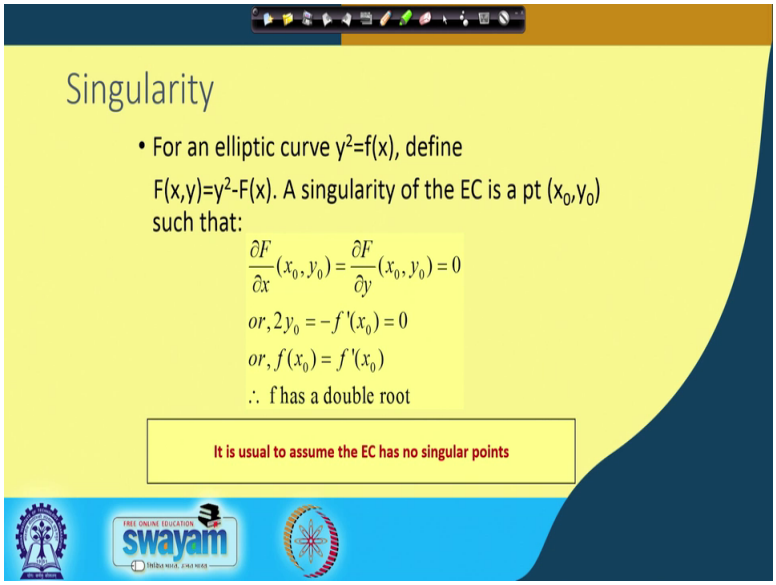
(Refer Slide Time: 00:23)

So, today we shall be starting with definition which we did not covered in the last class which was on what is called as singularity of elliptic curve. I say that the elliptic curves are typically nonsingular. And then we shall be going into what is called as scalar multiplications which is essentially the corner store a elliptic curve cryptography. And we will try to see various approaches of obtaining this elliptic curve operations.

So, we shall be trying to look into some techniques which are called as say Montgomery techniques you know like some very interesting techniques of how you can operate with only x coordinates you do not need the y coordinates for doing your computations ok. And we shall be subsequently looking to a transformation which is called as projective coordinates the objective is to reduce the complex inversions which are often required in this kind of operations. And we shall be taking a quick look into what is called as mix coordinate systems. And finally, we shall be trying look into some penalization

techniques and how we can paralyze the Montgomery ladder which is which can be utilized for efficient architecture.

(Refer Slide Time: 01:23)



So, let me quickly define what is called as singularity of an elliptic curve. So, for an elliptic curve say y square equal to f x define. So, you know like you can write an function f x comma Y as y square minus f x ok. So, singularity of the elliptic curve is at a point X 0, Y 0. If its partial derivative with respect to x and partial derivative with respect to y both vanishes ok; that means, del F del x with x 0, y 0 at the point x 0, y 0 is equal to del F del y at the point x 0 y 0 both are equal to 0 ok.

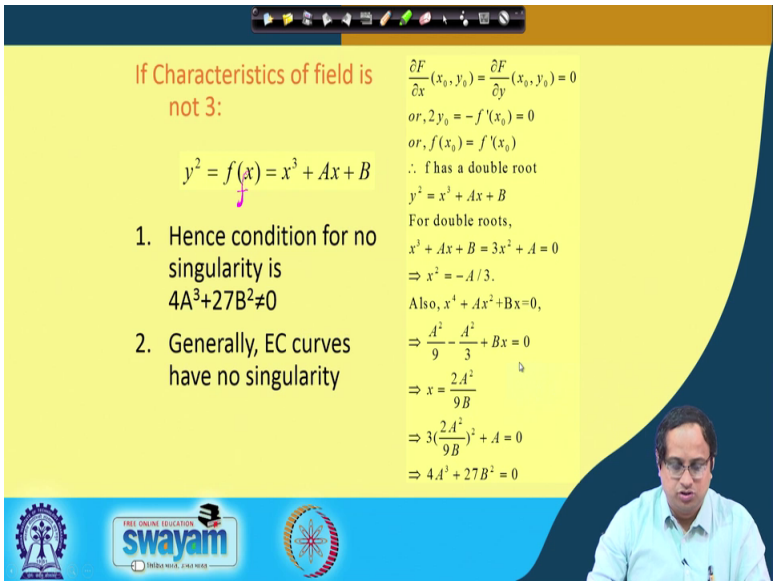
So, if you take this equation therefore, if you derivative with respect to say x ok. Then you get minus you know like you get you get minus F dash x 0. So, you get minus F dash x 0 and likewise right if you take 2 so this will f x comma y equal to y square minus small f x ok. So, please correct this is y square minus small f x ok.

So, so; that means, right what I mean to say is that this is y square minus f x small f x. So, then if I derivate this with respect to y then I get 2 y ok. And if I derive this with respect to x then; that means, equal to minus f dash x ok. And if I plug in the point x 0 comma y 0 then; that means, 2 y 0 is equal to minus f dash x 0 ok.

So, now you can easily verify therefore, that; that means, that 2 y 0 is equal to 0 which means y 0 is equal to 0; that means, f x 0 is equal to 0. And likewise F dash x 0 is equal

to 0; that means, the curve has a double root at the point x ok. So, it has got a double root; that means, it also itself satisfies the curve and it also satisfy this differential equation.So, it is usually it is usual to assume that elliptic curve has no singular points and therefore, we considered nonsingular curves in our discussion.

(Refer Slide Time: 03:19)



So, therefore, right let us try to kind of you know like similarly kind of derive you know like the criteria which is essentially what we have for such kind of let me clear this ok..So, therefore right if I for example, you know like if I take my equation as y square equal to x cube plus a x plus b. And I want to essentially you know like derive the criteria for which this will be a nonsingular curve.

So, this is often related to what is called as a discriminant of the curve ok. And the discriminant in this case is 4 A cube plus 27 B square. If this is not equal to 0, then this is implies that there is no singularity in the curve you can easily derive it. So, as I said that it means that there are no double roots.

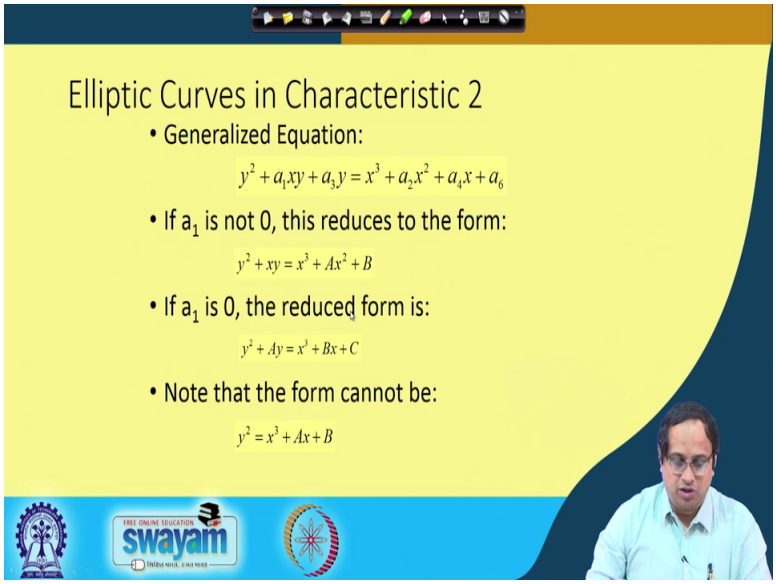
So, therefore if I take the curve like x cube plus A x plus B equal to you know like and if I you know like since it has got no double roots; that means, for double root it would mean that this is equal to 0. And likewise the derivative with respect to x which is 3 x square plus A that is also equal to 0. So, if 3 x square plus A is equal to 0 which means x square is nothing, but minus A by 3.

So, now if you take this equation x cube plus A x plus B equal to 0. And if I multiply x on both sides then I get x to the power of 4 plus A x square plus B x equal to 0 x to the power of 4 therefore, you can write as x square squared; that means, minus A by 3 whole square which is A square by 9 and x square you can substitute by minus A by 3. So, you get minus A square by 3 plus B x equal to 0. So, therefore, you obtain that x is equal to 2 A square by 9 B ok. So, you can just do a few arrangements and you will get x equal to 2 A square by 9 B.

So, now if you take this equation 2 A square by 9 B and you know that since you have got this equation that is you know like 3 x square is equal to minus A; that means, 3 x square plus A is equal to 0. So, therefore, you will get 3 into 2 A square by 9 B whole square plus A is equal to 0 and from there you can get this criteria which is 4 A cube plus 27 B square is equal to 0. That means, for singularity you need to have this discriminant is equal to 0, for non singularity this means that they should not be equal to 0 ok.

So, typically we consider elliptic curves which has got no singularity in them ok. Because separately right that makes the discrete log problem more hard ok. And that is essentially what we required for our public key cryptography ok.

(Refer Slide Time: 05:45)



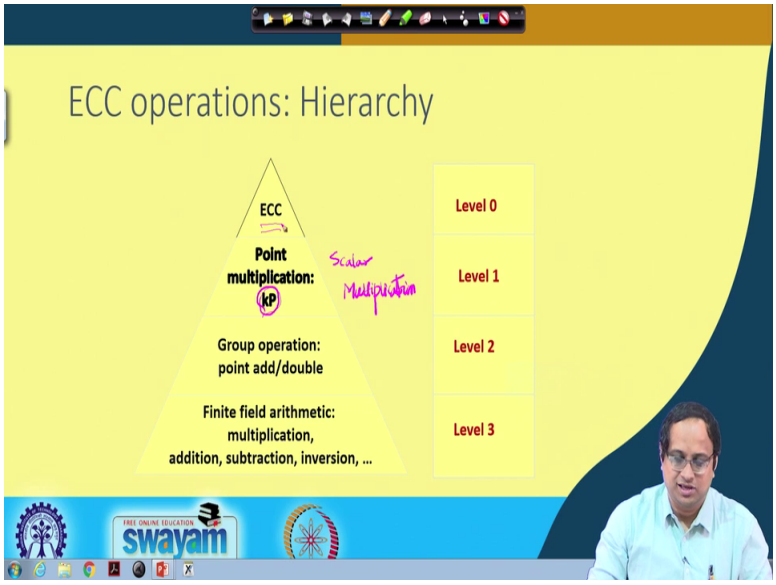
So, now right I mean there will be more focusing on elliptic curves in characteristic 2. Again we can start with our original Weierstrass equation and you can see that with few

simplifications like for example, if a 1 which is this coefficient is not 0, then this reduces to the form y squared plus x y equal to x cube plus A x square plus B.

Again I am not going to the derivations, but you can take it and you can also try it yourself. And likewise if a 1 is 0 then this reduce form is y square plus a y equal to x cube plus B x plus C. So, you can see that there are small differences in the form of the equations, but nevertheless all these forms are much more simple compared to the original Weierstrass equations and therefore, depending upon the cases these are often used for doing computations.

Likewise right and also note that it is very important note that this equation cannot be of this form like y square equal to x cube plus A x plus B. Because we remember that when we derive this we assume that the characteristic was not 2 ok. And therefore, right we get curves of different nature compared to this ok, but nevertheless they are also you know like still having the same you know like same nature; that means, it is quadratic still with y and cubic still with respect to x ok.

(Refer Slide Time: 07:01)



So, now we were all set to understand how the elliptic curve operations are done. So, the elliptic curve essentially you can imagine is kind of a pyramid.So, you have got finite field operations at the bottom where you are doing your multiplications additions and field inversions, on that you define your group on which group operations are you know

the chord and tangent rule that we saw. So, we saw how to add 2 points P and Q if the point P and Q are same then I add P with p.

So, in that case we draw a tangent at the point and that essentially is often called as a doubling operation because you are adding P with p. So, you are getting 2 P that is the doubling operation ok. So, you have you can define your addition you can define your doubling and now with addition and doubling you are all set to compute what is called as the scalar multiplication ok.

So, in scalar multiplication very simply put you essentially do this operation.So, you basically calculate so this is your k P ok. So, here this is often called as the scalar multiplication. So, scalar multiplication often implies that you are adding you know like you are adding say the point P k times ok.

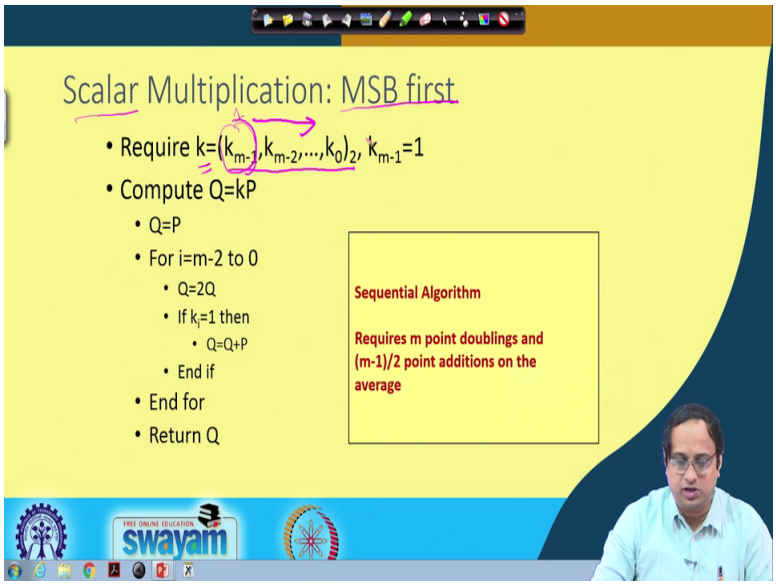
So, if I adding to the point P k times that implies that you are you can also you know like of course, that would that can need to a very in efficient operation. So, there are techniques of how you can make it more efficient ok; that means, how you can make the scalar multiplication more efficient, That means, you can make them work in you know like reasonable amount of time and also consuming reasonable amount of resources ok.

So, that is essentially you know like the sort of main theme of today's discussion how you can you know like do the scalar multiplication in an in an efficient manner. And subsequently what we will see are how you can make a corresponding hardware architecture for realizing the scalar multiplication operation ok. And on this right your elliptic curve cryptography is kind of dependent.

So, we will not be really looking into this ECC part where there are several versions you have got elgamal cryptosystems, you can apply elliptic curve for key exchanges which are often called as a Diffie Hellman key exchange ok, but all of them right fundamental uses this scalar multiplication.

So, the scalar whenever we talk about the elliptic curve processor it means that there is a hardware unit which essentially able to do this scalar multiplication ok. So, therefore, or the entire focus is in making this scalar multiplication more efficient. So let us try to you know like see therefore, how we can obtain the scalar multiplication in a more efficient manner.

(Refer Slide Time: 09:39)



So, there are some techniques of doing this ok. The easiest thing to see is that what is called as you know like double and add algorithm, but the double and add algorithm operates from the MSB. So, what essentially is done here is we take you know like what we do is that we take the scalar this is the scalar is k.

So, this k is called as the is called as a scalar ok. So, this is a scalar which is essentially referred in the scalar multiplication and we write that in a binary format. So, I means all of these values k 0, k 1 till k minus 1 are nothing, but either 0 or 1 in this case.

So, now I want to calculate Q equal to k P ok. So, I want to basically add P k times, but what will what. So, now, what I will try to do is that I will try to process it from the MSB. So, this is your maximum significant bit, so we try to operate it from this side ok.

So, note that we will assume that in this expansion they that the previous term that is k m is implicitly 1 ok. So, we try to process it from m minus 1.So, therefore right what I so, so you can. So, so basically right I mean what we will first look is in a scalar multiplication algorithm where we expand a scalar.

So, the scalar is in this case you know like this term that is the variable k and we essentially are so this is your scalar k. That means, this k is essentially often refer to as a scalar and here we kind of expand is in this binary format.

So; that means, k m minus 1 till k 0; that means, we are used m bits to represent k and all of these very a term like k 0, k 1 till k minus 1 there, either 0 or 1. So, it basically the binary expansion of k. So, note that k minus 1 is 1; that means; this term is essentially 1 ok. So, this term is implicitly assumed to be 1 in this case ok.

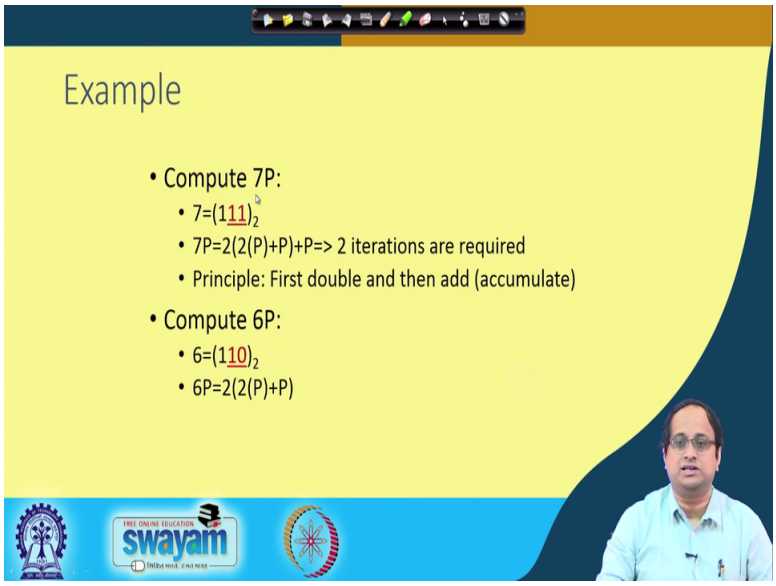
So, now when we are processing so when we say MSB first; that means, maximum significant bit first. It implies that we have processing from the left ok. So, this is also called as the left to right scalar multiplication. So, the algorithm is very simple. So, what we want to do is basically add P k times, but [explicit/explicitly] without explicitly doing.

So, so what we do now is that we initialize Q to P and then from m minus 2. Note that I know that k minus 1 is 1. So, I start processing from m minus 2 and I start to do always a doubling operation and I do a conditional addition depending upon the fact that whether this k i bit is 1 or not ok. That means, if k i is equal to 1, then I do a conditional addition that is Q equal to Q plus P and finally, I return Q as the result ok.

So, you can easily verify that these answer will be correct ok; that means, you can you will get correct result. But before I going to that you can observe that this is a sequential algorithm because we are you know like processing one after the other. And it will typically require m point m point doubling operations because you always doing a doubling operation. But if you assume that on the average half of the half of these terms are 1 and half of them are 0 then you are doing half number of additions ok. Then you are not doing m number of additions, but probably you are doing m minus 1 by 2 number of point additions so on an average ok.

Sometimes you may do all I mean if it is all 1 then you have to add always. And likewise if it is all 0 then you have probably add even less, but on an average right you should be a kind of half number of additions that you have to do. So, therefore, right I means this algorithm is correct you can easily verify this.

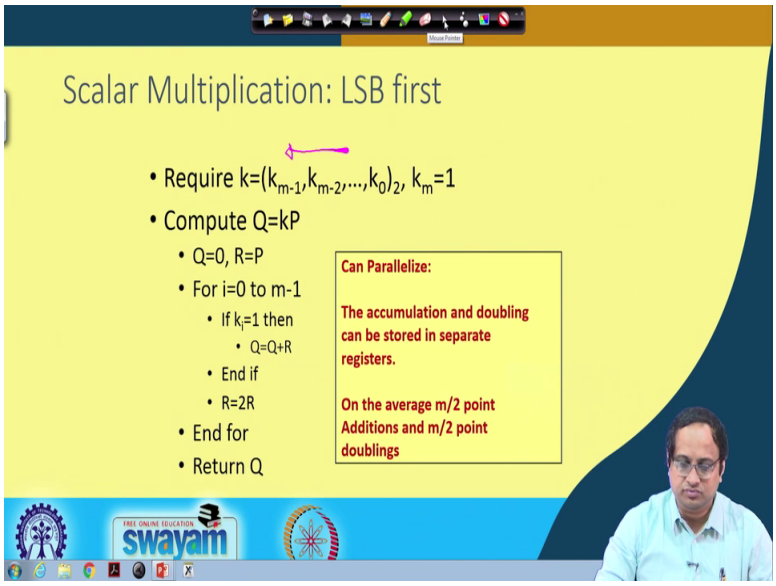
(Refer Slide Time: 13:27)



So, suppose I want to calculate 7P, so 7 can be written as 111. So, note that I if I leave out the first one and I start processing from this 1 ok; that means, like I will do you know like 2 into P plus P ok, that is because of it is 1. And likewise you see the second bit is also 1 I again do 2 into that plus P ok.

So, I can see that this is 2 P plus P which is 3 P into 2 is 6 P plus P is 7 P ok. So, you note that 2 iterations are required and I first do a double and then I do an add. So, that is the basic principle of this technique? If I likewise want to calculate 6P, so 6 P is 110. So, here because of this 1 I will do a 2 into P plus P, but since this is a 0 I now only do a doubling, but I do not do an addition ok. So, therefore, I get 2 P plus P is 3 P into 2 is 6 P ok. So, you can also write likewise this is very simple you can do it very pretty simply ah, but there are you know like some variations of this also which will very soon see.

(Refer Slide Time: 14:27)



For example you can also do it from the right this is called as a right to left scalar multiplication. So, if I start processing from the right; that means, I start processing from this side ok. And essentially I am starting to process from the LSB in that case. So, in this kind of algorithms or in this variations right we basically have got 2 registers we have got a register Q and a register R and now I initialize Q to 0 and R to P ok.

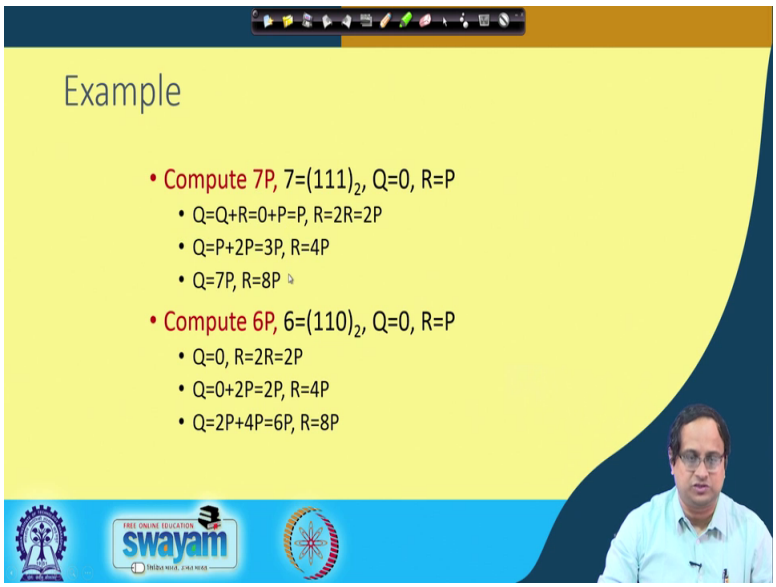
So, now what I do is that if k i is 1 then I do a Q plus R; that means, I add these two registers and I store the value in Q else I only do a doubling in R ok. I only do a doubling operation in R; I mean rather I do a doubling operation always ok. So, this is exactly similar as the previous one; that means, I am always doing a doubling, but the doubling is done in the register R ok. But the addition is being done in the register Q and, but the addition is done in a conditional way; that means, if your secrete bit is 1 only then you are doing the addition operation ok.

So, there are certain small points here for example, you can parallelize it ok; that means, since the accumulation; that means, addition and doubling are done in separate registers you can be pretty much parallelize you know like if you have got resources of course. You can parallelize the accumulation and the doubling operation the addition and the doubling can be parallelize which means like since we will see as will see that addition is more complex than doubling operation the latency will be that of course, that of the addition operation ok.

So, therefore, right on average you can imagine that there will be m by 2 point additions and m by 2 point doublings ok. So, the note that here also you are actually doing m doubling operations ok, but then why do I write m by 2 point doublings, I am writing because of the assumption that they are parallelized ok. So, if they are parallelized right then the whenever I am doing addition then of course, the cost is that of an addition operation, but on the other cases it is only the doubling operation which is coming into play ok.

So, with that assumption right you can essentially and the important point is that if you pay this resource; that means, 2 registers few more computations in parallel then there is a scope of parallelism in the algorithm although it's a sequential algorithm. The nice thing is that there is a scope of parallelism in the algorithm ok. So, now right I mean we can essentially you know like look at the correctness of this equation.

(Refer Slide Time: 16:53)



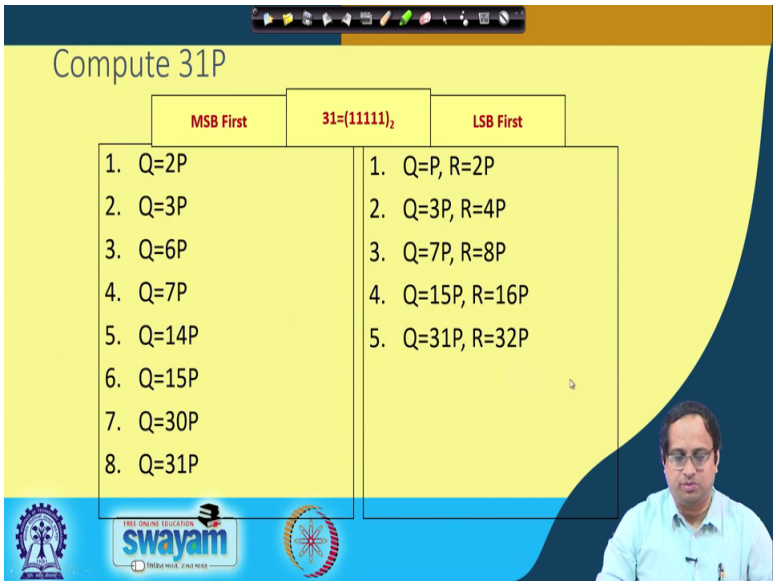
Again let us take 7 P and 6 P as an example. So, again I am processing now from the right side. So, I initialize Q to 0 and R to P note that that is 1. So, I add Q with Q plus r. So, I get 0 plus P which is P and I double R. So, I get 2 R here; that means 2 P again the next bit is 1 ok. So, next bit is 1 which means I add in Q; that means, I add P with 2 P. So, I get 3 P and since it is 1 I also I mean I do a doubling operation always.

So, I calculate from 2 P, I calculate 4 P the next bit is again1. So, note that here I have to go for all the 3 bits actually ok. So, therefore, here it is 1, so I add 3 P with 4 P I get 7 P

and I always do a doubling. So, I get 4 P into 8 P, but this is not useful here because my result is already computed in 7 P ok. This is required if I have got further more things or inputs to process ok.

Likewise considered 110 case, so in this case right I get a 0. So, I do not do anything in Q there is no addition being done. Of course, I have to do a doubling because doubling is always doubling because these are all always double algorithms. So, I double R I get 2 P next time I have got a 1 here. So, I again add with Q with P 2 with 2 P here. So, I get 2 P in Q I double R I get 4 P and in the next iteration I add 2 P with 4 p. So, I get 6 P, so 6 P is my result ok.

(Refer Slide Time: 18:17)

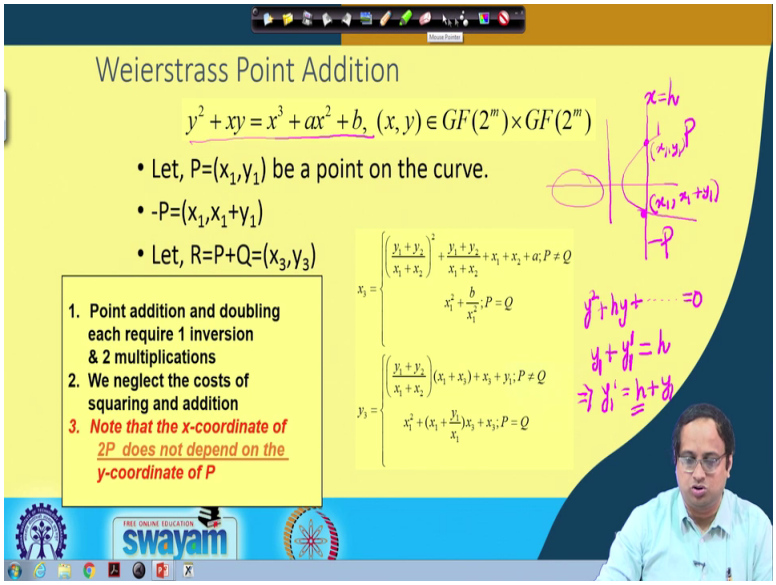


So, right have been so now if we compare say 31 P which is slightly bigger example and compare these 2 algorithms MSB first and LSB first. So, you can see here there are more steps which are required ok. So, you are doing Q equal to 2 P then. So, I am again, so I am doing MSB first I am processing from this 1 ok. So, I am getting a Q equal to 2 P, then I am doing a Q equal to 3 P, then I am doing 6 P, 7P, 14 P 15P, 30 P and 31 P. But when you are doing LSB first you see that you can do the additional doubling in parallel.

So, if you can do them in parallel then you can do it in lesser number of clock cycles or lesser number of steps. But both of them will give you the correct result and therefore, your choice right essentially can be influenced depending upon in the context ok. There

can be more implications also about this, but at least this is at this is what we see at this point ok.

(Refer Slide Time: 19:13)



So, now we want to basically you know like consider in particular curves or characteristic 2. So, the suppose I assume that I have got this equation y square plus x y equal to x cube plus a x square plus b. So, this is my curve equation and I want to essentially you know like do my arithmetic where x comma y. That means, a point on the curve belongs to the G F 2 power of m cross GF 2 power of m field ok

So, now let P equal to x 1 comma y 1 be a point on the curve. The first thing to observe here is a minus P slightly different from what we have seen previously ok. So, minus P is x 1 comma x 1 plus y 1 ok. Why does it happen? So, you can easily see this the y wide work. So, so if I take you know like the point for example, I take a this equation say y square plus x y equal to ax square plus b and consider. So, the idea will be the same; that means, I have got my elliptic curve in this way. So, imagine that this is my elliptic curve and I will basically draw a straight line like this ok.

So, this straight line is say x equal to h ok. So, now, if this is my point you know like x 1 comma y 1. I consider this point as a negation of p so if this is P then if this is point is P then this point is minus P ok. So that means, I will take this line x equal to h and I will intersect in this curve ok.

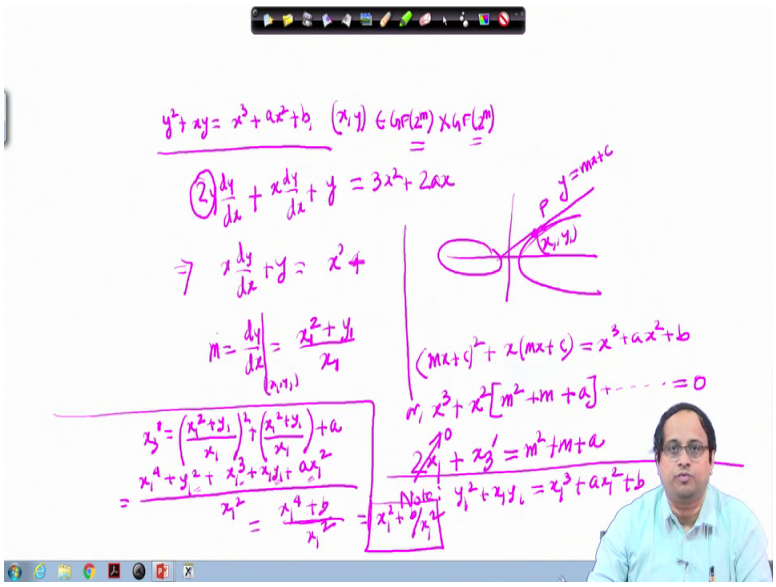
So, if I do that then I will get y square plus x will be h. So, I get h y remember that is a characteristic 2 field. So, I can bring this in this side, but that is I do not need it. So, finally, I get this is equal to 0 ok. 21

So, now there are you know like ah. So, we can observe that there are two equations here. So, one of them is say y 1 ok, the other one is say y 1 dash ok. So, y 1 dash is what we want to compute. So, this if I add will be the negation of h, but since it is characteristic 2 this will be equal to h ok. So, this implies that y 1 dash is equal to h plus y 1 ok, so h here is x 1.

So, if it is x 1 then I get x 1 plus y 1 of course, x coordinate remains the same. So that means, the minus P in this case is x 1 comma x 1 plus y 1 ok, this is the first thing to observe. So, likewise there are certain other things also which are slightly different compared to what we have seen, but essentially the ideas or the principle is still the same.

So, you can essentially derive the addition equations and you can derive the doubling equations ok. It is interesting to see how we can you know like draw the doubling equation. So, let me try to you know like concentrate on that ah. So, let me try to kind of derive this equation. So, suppose I have got you know like my equation has y square.

(Refer Slide Time: 22:25)



So, this is my equation right y square plus x y equal to x cube plus ax square plus b where x comma y all are elements in GF 2 power of m cross GF 2 power of m ok. So, now, I want to obtain the doubling equation because that is a particular importance to me ok.

So, here what I do is I derivate this as previously, but remember that these are characteristic 2 fields ok. So, therefore, write when I write 2 y d y by d x right then this 2 is actually equal to 0 in this field ok. So, but I let me continue that so if I write 2 y d y by d x plus I derive this right. So, therefore, I write derivate this so I write here x and d y by d x plus y ok. This is equal to 3 x square so first I am just writing in a very naive way 2 x ok.

So, now if I put in the fact that I have not characteristic 2 so I will do modulo 2 of the coefficients. So, with that right this is equivalent to saying that this x d y by d x plus y is equal to x square plus 0. So, therefore, I would the just x square so; that means, d y by d x is equal to x square plus y by x ok. So, if I now do this at the point x 1 comma y 1 so; that means, this x 1 square plus y 1 divided by x 1 ok.

So, now if you take again apply the same chord and tangent rule; that means, we have got your elliptic curve and you want to calculate the doubling say at the point P; that means, you draw your tangent at this point ok. So, if you want to do this then I mean that you have got your equation as y equal to m x plus c where m is obtained in this fashion, so this is your m right m of the slope of the curve or slope of the tangent ok.

So, this is this is the tangent. So, therefore, right if I do this therefore, if I mean if I plug in this into this equation then I get m x plus c whole square plus x into m x plus c equal to x cube plus a x square plus b ok.

So, if I arrange this term then; that means, like this would imply that I bring x cube here and if I obtain the coefficients of x square then I will have m square plus m plus a plus other things we have I do not require here and this is equal to 0. So, now you see that I have got the point say x 1 so suppose x is x 1 comma y 1. So, now, you have got 2 x 1 because x 1 plus x 1 plus the point which want here.

So, suppose this is your x 3 dash and this is equal to m squared plus m plus a again note that this is equal to 0 here ok. So, therefore, this will vanish. And therefore, right you

have got I can I can take this space to write that x 3 dash right x 3 dash will be equal to x 1 square plus y 1 divided by x 1 whole square plus x 1 square plus y 1 divided by x 1 plus a this is your x 3 dash ok.

So, this right essentially turns out that if you do a little bit of simplification you will get the denominator as x 1 square and the numerator is x 1 to the power of 4 plus y 1 square plus x 1. So, I have to multiply this way I will x 1 cube plus x 1 y 1 plus a x 1 square right.

So now you can observe that since x 1 comma y 1 is a point on the curve ok. So, you can note here that this is note that x 1 comma y 1 is a point on the curve. And therefore, I will have y 1 square plus x 1 y 1 is equal to x 1 cube plus a x 1 square plus b ok. So that means, y 1 square plus x 1 cube plus x 1 y 1 plus a x 1 square this you can write as nothing, but x 1 to the power of 4 plus b by x 1 square ok.

So, now you see that there is no y in this equation this doubling right essentially can be is nothing, but x 1 square plus b by x 1 square ok. And therefore, you have completely eliminated y from this equation ok. So, this is the first thing to observe that this is a nice property where you only have x involved ok.

So, now with this setup right we can essentially try to you know like get back and understand what is the implication of this. So, the first implication is that you do not require any y for doing the doubling ok. But of course, write for x I mean for the addition where P and Q are not the same you still need y ok.

And now what we will see is a method which is due to what is called or what is called as a Montgomery ladder which says that you can still do the addition operation by a trick which is called as a Montgomery ladder trick, where you do not still need to compute or store the y coordinate ok. Before going to that you can observe here that for point addition and doubling both require 1 inversion and 2 multiplications ok.

So, for example, here you see that I need to calculate 1 by x 1 that is 1 inversion and I need to obtain 2 to perform 2 multiplications. Why then? Because I have to multiply by y 1 that is 1 multiplication and also I need to multiply with x 3 that is my second multiplication.

Note that I am neglecting operations due to squaring and multiplications with constants because for them I do not need a dedicated multiplier ok. When I am doing multiplication with a constant I can optimize it when I am doing a squaring operation, also we have seen it is much more simple. And in fact, there are some basis representation like the normal basis where a squaring is nothing, but a rotation so you can essentially do it quite simply ok.

But likewise for the addition operation you have to do 1 by x 1 plus x 2 which is 1 inversion. Again you have to do a multiplication because you have to multiply with y 1 plus y 2 and you have to also multiply with x 1 plus x 3. So, here it seems that both of them like addition and doubling both requires 1 inversion and 2 multiplication operations that is a cost ok.

So, what we will see in the next class is you know like how you can optimize this and how you can a perform the part from the Montgomery's ladder trick. So, that you can do the addition still without storing the y chords in the next class so.

Thank you for your attention.