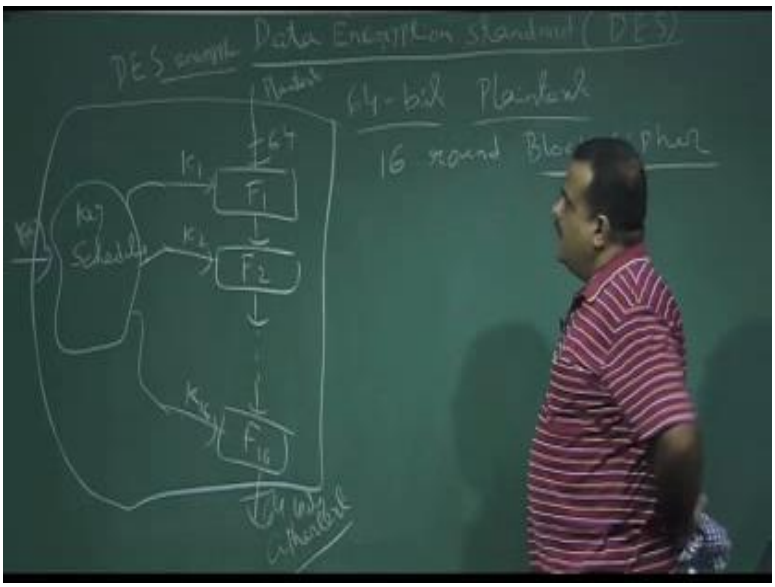
Internetwork Security
Prof. Sourav Mukhopadhyay
Department of Mathematics
Indian Institute of Technology, Kharagpur

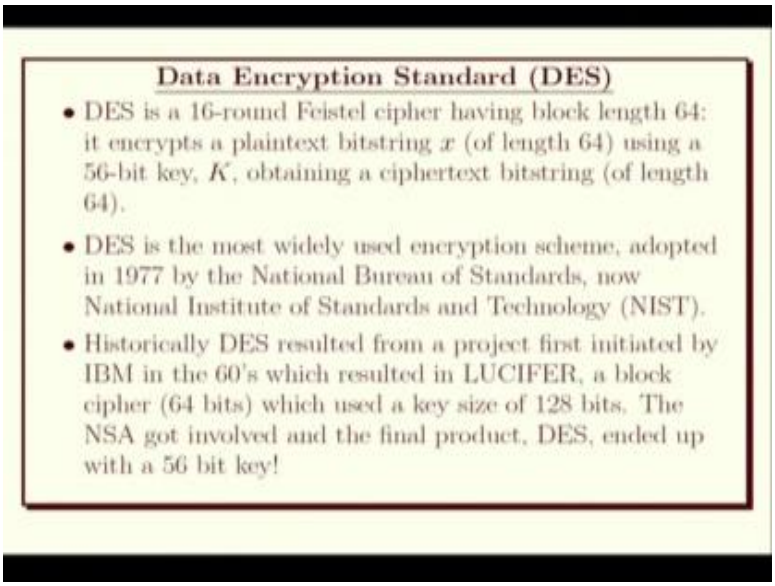
Lecture - 06
Data Encryption Standard (DES)

(Refer Slide Time: 00:20)



So, we will talk about another block cipher which is data encryption standard. So, this is basically a 64 bit block cipher. So, plaintext and ciphertext size are 64 bit and this is a 16 on block cipher, so r is 16.

(Refer Slide Time: 00:55)

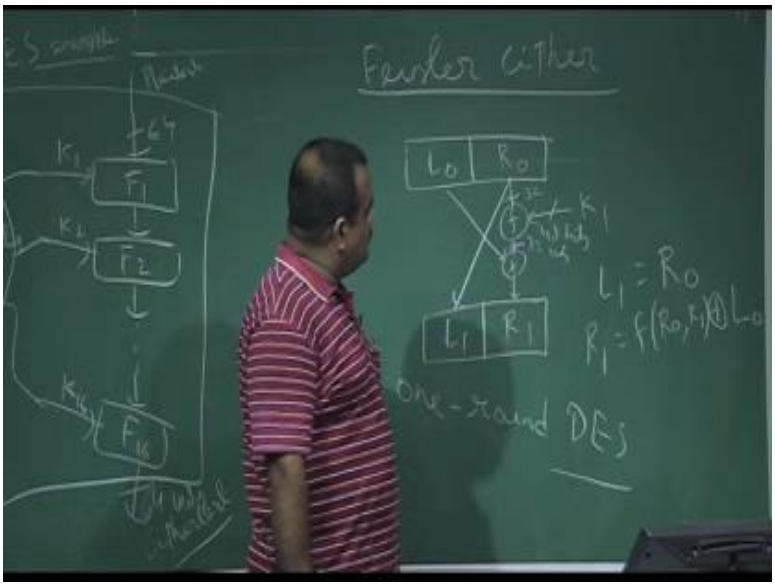


So, the history of this block cipher is basically, can you please go to the slide. So, it is a 16 round Feistel cipher, we will come to that what is Feistel cipher and the plaintext and ciphertext is 64 bit and DES is most widely used encryption scheme adapted in 1977s by NIST and they made this as a standard and this is historically DES is from the cipher which is called LUCIFER which is designed by IBM in 60s and this was they are modified this slightly and then it become a DES

So, basically it is a 16 round block cipher, so we have a plane text which is 64 bit and we have 16 rounds. So, F 1, F 2 then F 16 and for each round as we know we need another input each of which are called round key. So, you have 16 round keys and this round key is coming from key scheduling algorithm, which is taking the key; the secret key shared between Alice and bob and giving the round keys and we are getting this is the ciphertext. So, this is a typical 16 round block cipher and this is also 64 bit plaintext and ciphertext as 64 bit.

So, this is DES encryption and we have to talk about the round function, this is the general structure of the DES encryption, we have 16 round and for each round we have two input: one is the first round input is the plaintext and last round output is the ciphertext and for each round, we need to have the round keys case and these are coming from the key scheduling algorithm which we need to talk what is the key scheduling algorithm for DES and DES round function basically what is called Feistel cipher.

(Refer Slide Time: 03:41)

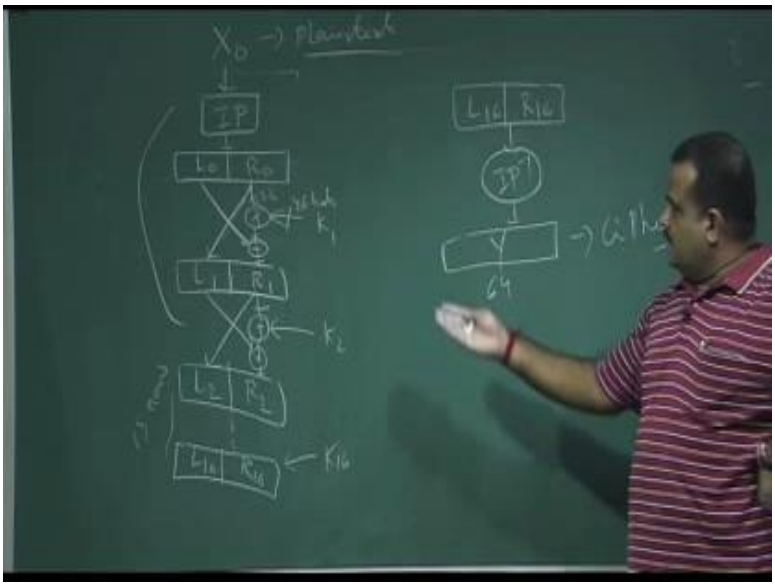


Feistel structure; Feistel cipher, so it is a 64 bit plaintext, we denote this L 0; R 0 this is a 1 round Feistel structure and it will output L 1; R 1. So, this R 0 is directly copied here and for this part, we have to apply function F which is taking input as r 0 and the round key you can secure 1 for first round key and it is beta is XORing with this and giving us the R 1.

So, basically you have this L 1 is the R 0 and R 1 is basically F of R 0 comma k 1 XOR with L 0. So this is the input, this is the output, so this is a typical one round Feistel cipher. So, DES is basically each round of DES is basically this function; this Feistel function it is just dividing the 64 bit plaintext into 2 part 32 bit, 32 bit and this 32 bit is directly copied here and this 32 bit we are taking the input of a function F which is having another input which is the round key which is 48 bit and this is 32 bit. So, this F has 2 input 32 bit and 48 bit and it is giving us output 32 bit again and this 32 bit output here XORing with the left part of the plaintext and we are getting the bit of sectioning and then we are getting the right part of this output are on, so this is a one round DES or one round Feistel cipher.

So, the DES rounds are basically this Feistel and this round keys is basically 42 bit. Now how to get this round, we will talk about the key scheduling algorithm from the secret key how we are getting the round keys.

(Refer Slide Time: 06:49)



So, let us just write the full brief description of the DES, so each round is a Feistel structure and so basically we have the plaintext X; which we denote by X 0 this is the plaintext. So, it is a 64 bit, so what we are doing we are breaking into this; into two part. So, first this is 64 bits, so we are applying a permutation on it which we denoted by IP.

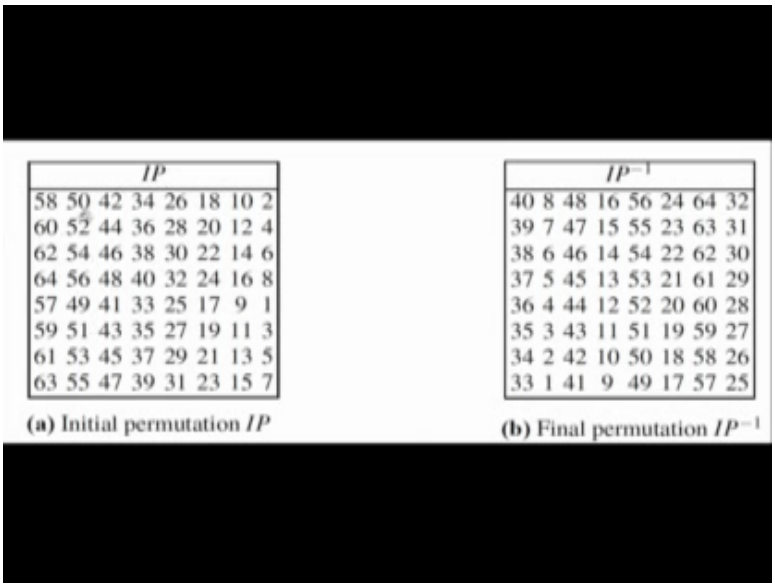
So, this is a permutation which is taking 64 bit input and it is again giving us a 64 bit output this is a permutation we will talk about will see what is this permutation is. So, then this 64 bit, we are now dividing into two part 32 bit, 32 bit this is a l 0, r 0 and then we are applying the Feistel cipher. So, this part is directly copied here this is L 1 on R 1 and then this part we are applying a function F which is taking input k 1, the round function; this is 48 bits and this is 32 bit and this we are XORing with beta is XORing with this and we are getting this L 0, R 0.

Now, again will this is the first round function of the DES, so in the first round we have a one extra (Refer Time: 08:14) which is called permutation operation, we are just sapling the bits, you just positioning the position of the bits random happening. So, what is this IP will come will come in a moment. So, this is first round F 1 then after that again we have to apply the second round. So, second round is basically, so again it is a Feistel cipher, so again it will. So, this is L 2, R 2, so this will be copied here. So, this will apply the same F function and this is the k 2 second round function and these will XOR with this, this will this, but this way we will continue.

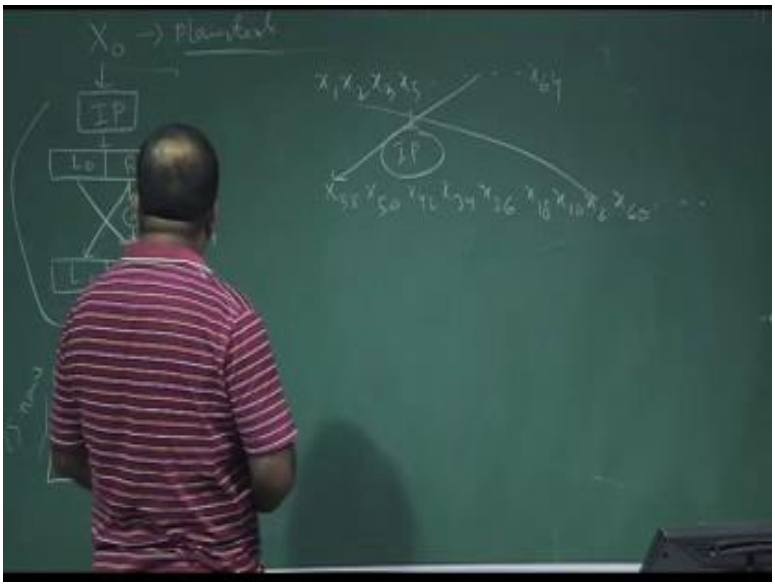
So, again we have another Feistel cipher, so this way we will continue dot dot dot finally, we reach to L 16 R 16. So, after reaching to L 16; R 16, so L 16 R 16, so these are many these are 15 more Feistel cipher; 15 rounds this is 1 round first round then we have 15 round of this. After that what we do we will apply the inverse permutation of IP inverse, we have a IP then we will apply the IP inverse and then will get the ciphertext this is 64 bit, this is y this is ciphertext.

So, for here we have k 16, so for each; so we have k 1, k 2, k 16 round function each round keys and each round this is a 48 bit, so now will see what is this IP.

(Refer Slide Time: 10:29)



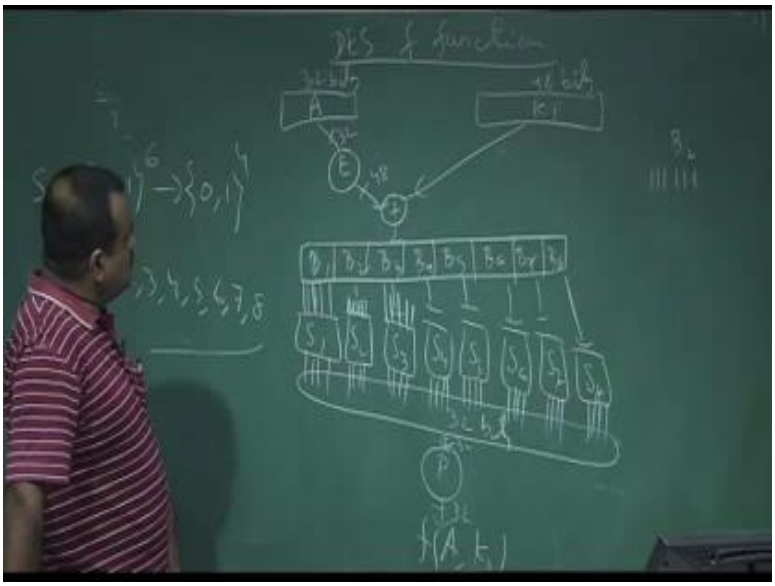
(Refer Slide Time: 10:43)



So, can you please go to the slide? So this is the IP. So, IP is basically, so we have this plaintext. So, plaintexts are basically x 1, x 2, x 3, x 4, like x 64, now this on this we want to apply this I P; IP means we are sapling the bits. So, how we are sapling we are sapling by this, so this will show x 58 will come here then x 50 then x 42 then x 34 then x 26 then x 18, x 10, x 2 then x 60 like this. So, it is basically, so x 56 is coming here x 2 is going there. So, sapling of the bits can you please go to the slide, so this is the way we are sapling. So, x 56 will come here then x 50, x 58 then x 50 like this x 2 then x 60 like this. So, we are just sapling the bits and this is the inverse permutation of this permutation and these we are going to apply in the after this 16 round of the DES.

So, this is just IP inverse, so inverse of this permutation. So now we will talk about this F function because we have used this F function here. So, this is the typical DES encryption, you have to talk about decryption also. So, let us just talk about the F function because we have used F function here, so F has two input.

(Refer Slide Time: 12:30)



So, DES function, so it has two input one is 32 bit; we denote this by say a and another one is the key; say k 1 for the first round, this is 48 bit and this is 32 bit.

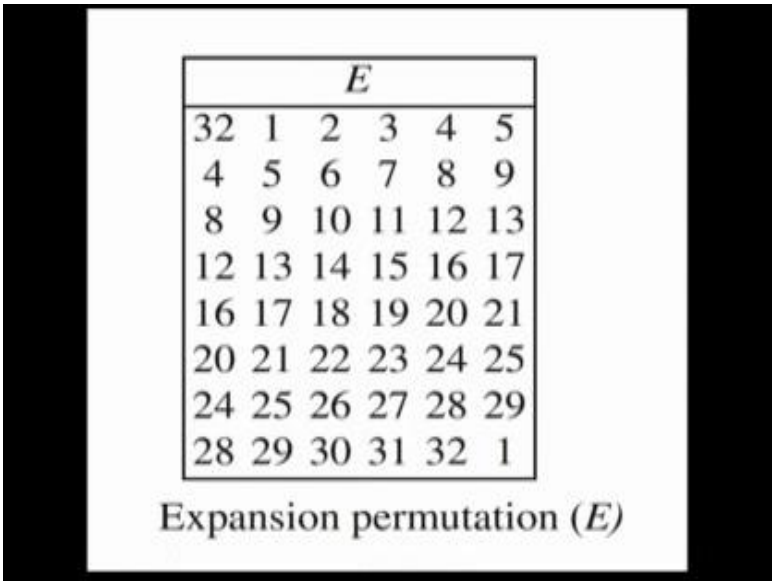
So, what we are going to do, we want to XOR this beta is XOR, but these are two different size. So, we cannot do the beta is XOR, unless we do something unless we expand this or unless we reduce this, so better to expand this. So, we will apply a expansion function on this to make it this is 32 bit to make it 48 bit and then this is also 48 bit, now we can easily bit wise XOR these 2 and will get a 48 bit output.

Now, this 48 bit output will divide into 6 bit blocks. So, how many blocks will be there, so this is 48 bits, so will be divide into 6 bit blocks. So, this is say B 1, B 2, B 3, B 4, B 5, B 6, B 7, B 8, so B I is 6 bit. So, we just divide into 6 bit blocks because we want to apply S-box which is taking input 6 bit and giving output 4 bit.

Now, suppose we have some S-box say s 1; which is taking input 6 bit 1, 2, 3, 4, 5, 6 and giving output 4 bit. We have another S-box, s 2 which is again taking input this B 2; 6 bit giving 4 bit like this 6 bit S 3, S 4, S 5, S 6, S 7, S 8. So, this S I are basically the substitution function which are S-box basically, so S I is basically taking a 6 bit input and giving us the 4 bit output. So, I is equal 1, 2, 3, 4, 5, 6, 7, 8 we have such 8 S-boxes, so how they will look like; we will come to know. So, each of this is a 6 bit we are applying this S-box and we are getting 4 bit output.

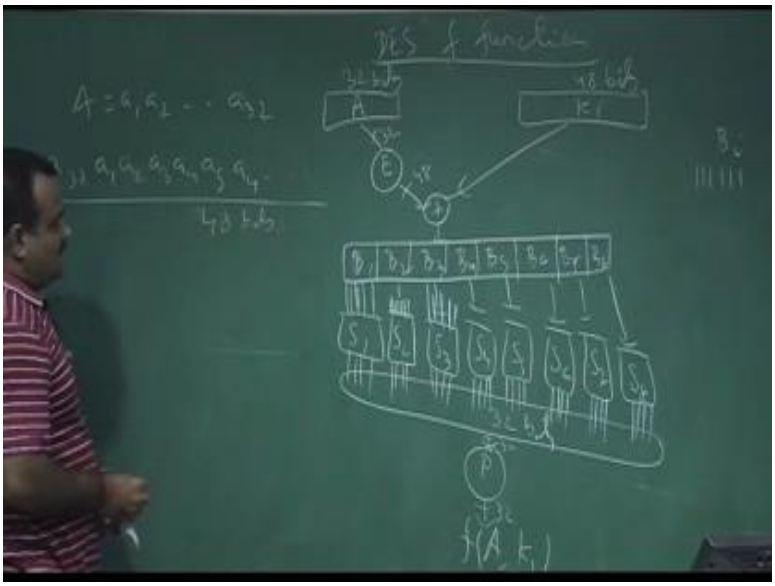
So, 4 bit output, so then we what we do then this is total is 32 bit then we apply a permutation on this 32 bit which is denoted by P again sapling the bits and then this is the output of f A comma k 1, so this is the f function; so this is again 32 bit, so this is 32 bit. So, it is just e is the random I mean permutation which is taking 32 bit giving us a 32 bit, so this is typical F function of this DES Feistel cipher.

(Refer Slide Time: 16:51)



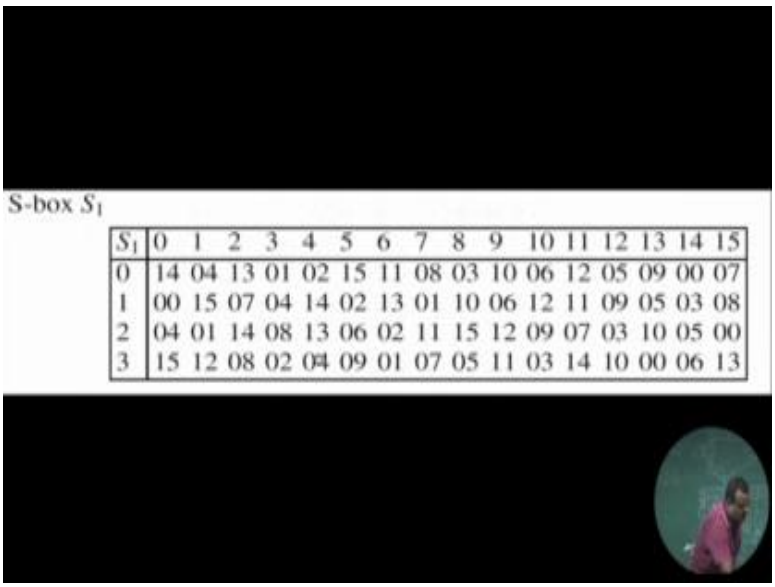
So, now we have to know what are this expansion function and the S-box. So, this is the expansion function, so we have a 32 bit number. So, from this 32 bit we want to make it 48 bits, so some number has to be repeated. So, say for example, 4 is repeated; so this is how many rows 1, 2, 3, 4, 5, 6, 7, 8; how many columns 1, 2, 3, 4, 5, 6. So, this is a 48 bit number, so we just apply this expansion function.

(Refer Slide Time: 17:33)



So, we are applying this expansion function over here, if our A say A 1, A 2, A 32, so by that expansion function it is telling this will be A 32 then A 1 then A 2, A 3, A 4, A 5 then A 4 again like this. So, this will be again 48 bits and this is given by this matrix this is a 8 by 6 matrix, so this is again 48 bits. So, basically we are repeating some of the bits 4 has repeated over here 8 has repeated like this 9 has repeated 13 has repeated. So, many bit because we have only 32 bit, we want to make it 48 bit, so we have to repeat the bits.

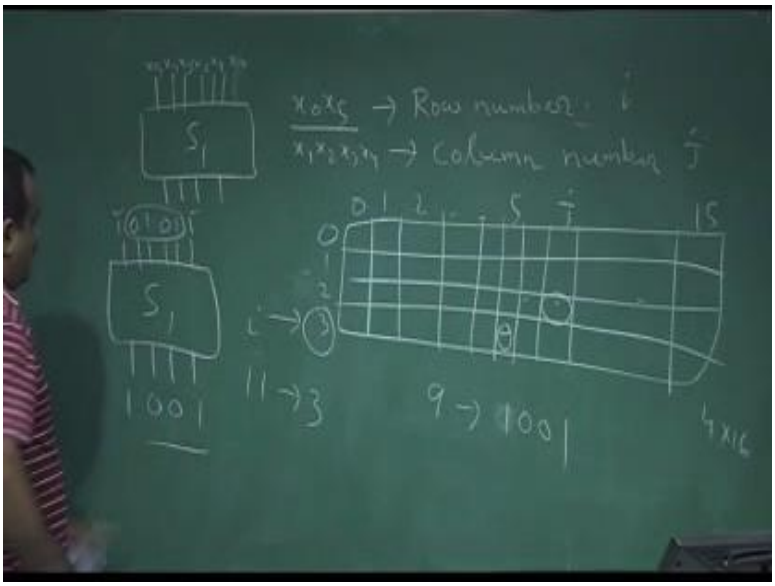
(Refer Slide Time: 18:24)



S-box S_1

S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13

(Refer Slide Time: 18:47)



So, this is the expansion function now we have we need to talk about the S-box. So, S-box it is taking, so this is a typical S-box S 1. So, basically each of this S-box is taking 6 bit input sorry and 4 bit output. So, if you denote this by x 0, x 1, x 2, x 3, x 4, x 5, so this is says S 1 we are talking about; so now this x 0 and x 5. So, suppose we have a table of size, so we have a matrix. So, this is 0, 1, 2, 3; so 4 by 16 matrix. So, this is 0, 1, 2 up to 15 if you start from 0. Now x and each of these entries is in hexadecimal form because output is 4 bit, so 4 bit can be representing x form. So, each of this entries are in hexadecimal form.

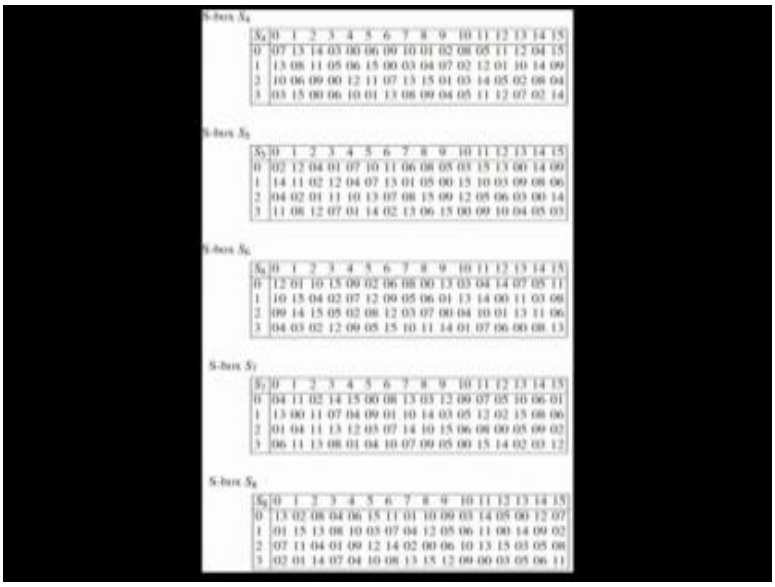
So, now suppose we want to get the output for this x 0 to x 5. So, x 0 x 5 will give us the row number and this x 1, x 2, x 3, x 4 this will give us the column number of this matrix. Suppose this is I and this is say j, so will go to that particular, so this is a j and this is a I will go to that particular position and we will get the value over there that is in hexadecimal form and that will be the output.

Now, for example, for S 1 suppose we want to get the output of this say 1 0, 1 0, 1 1 say we want to get what is the output of this what is the output of S 1, if the input is this. So, these 2 bit will denote the row number. So, 1 1 means 3, so this row, so this is our third row and these bits; this is basically 0, 1; 1 0. So, this is basically 4 and then plus 1 5, so this is basically fifth. So, we have to go to the; so this is fifth say for example, we have to go to this position and you have to get the value, so can you go to the slide please.

So, this is the S-box; it is giving in this form where 0, 1 this. So, we are looking for third row and the fifth column. So, this 0 9 is the S-box we are looking for, the output we are looking for.

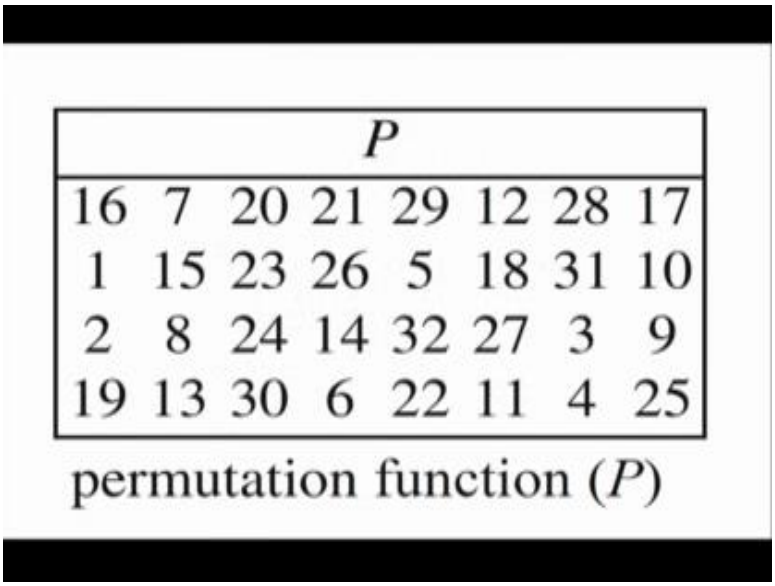
So, 9 is the output, so 9 means or what 9 is basically 0 sorry 8, 1, 0, 0, 1. So, basically 1 0 0 1 is the output of this S-box; of this value. So, this is how we have to read this S-box from the table, so there are 8 S-box.

(Refer Slide Time: 22:52)

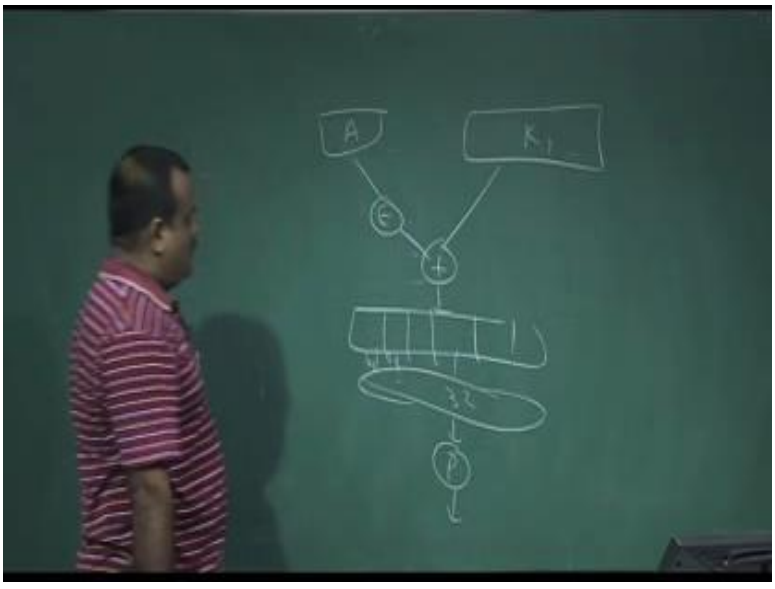


Can you please go to the slide? So there are 8 S-box this is the first S-box S 1, S 2 then we have the remaining S-box is like this. So these are given in the table, so designer just there is no explanation how they are getting this S-box, but they are just giving this S-box by this table or by this matrix, so this is 1 of the drawback of this DES.

(Refer Slide Time: 23:18)

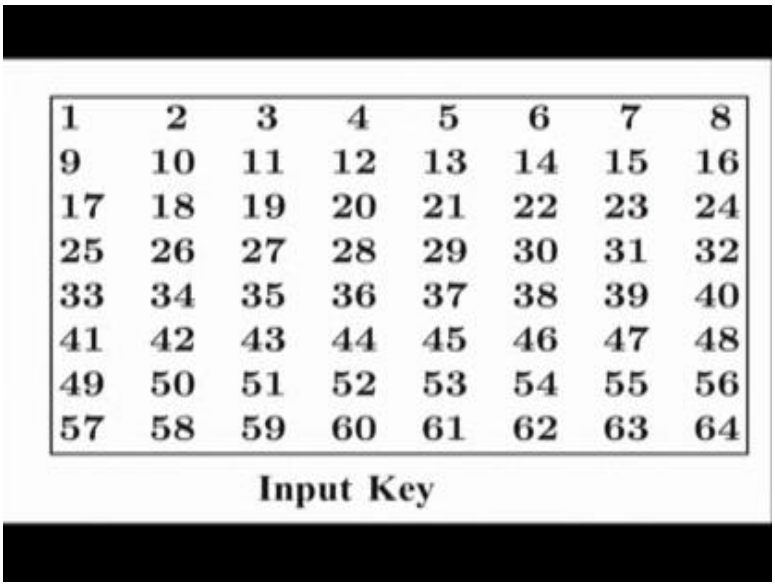


(Refer Slide Time: 22:36)



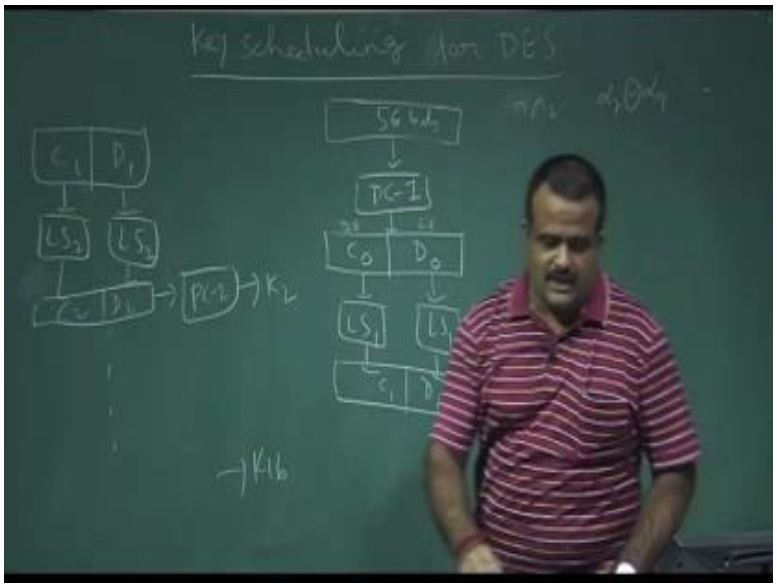
Now, come to the last permutation function in that f. So, if you remember, so this is this is A. So, we are still in the F function. So, we are breaking it into we are applying the S-box then this is S-box is S 1 is to then this is 32 bit and then we are applying a permutation P. So, this permutation is given in this table, so this is again a 32 bit number will be again 32 bit number. So, this is given by this, so we have 16, so S 1 is going to here. So, S 16 is coming in the front and then S 7 is coming in the next. So, this is just a sapling of the bits like this, so this is our P permutation.

(Refer Slide Time: 24:47)



So, this is the F function of the DES one now DES which is the Feistel cipher; now we talked about the key scheduling algorithm, so how we can get the round keys from this secret key k.

(Refer Slide Time: 25:02)



So, that is called key scheduling for DES, so for key scheduling. So, our key is actually the DES key size is this is plaintext size, this is the ciphertext size and this is the key size. So, key size is typically effective key size is 56 bit, but usually keys are given in 64 bits. So, if you look at this slide keys are given in 64 bit k 1, k 2, k 3, k 4, k 5, k 6, k 7, k

8, k 9 like this, but these are the bits we used for the parity checking to detect the any error in the communication or not.

So, these are not including the keys, so effective key is other than this parity check bit. So, these bits are basically effective key bit and this is 56 bit. So, if you just, so this is our effective key which is 56 bit then these are the bits is usually come for the parity checking. So, this we ignore when we take the key, so we have key is 56 bit. So, we take this 56 bit key; secret key and we divide into 2 parts then we apply a permutation which is called P C 1. So, 56 bit means this is 28 bit 28 bit, so total is 56 bit.

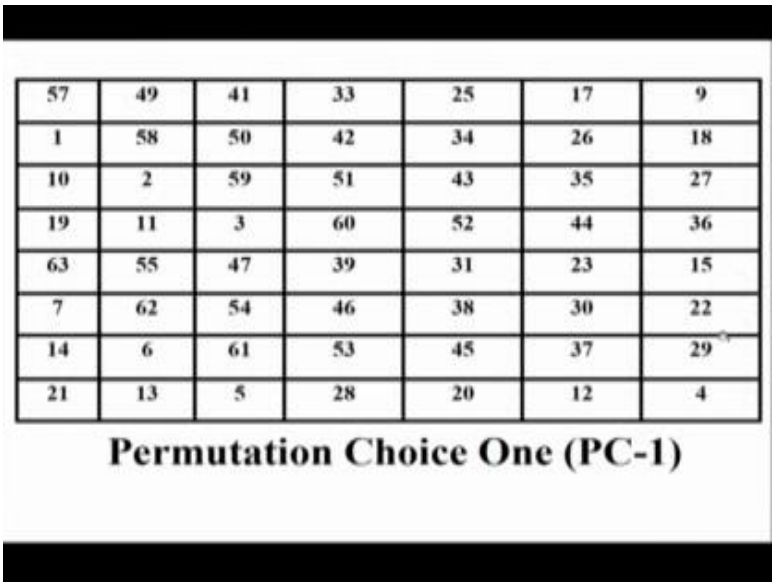
So, we apply permutation on this, so we do not partition first, this is a 56 bit number 56 bit key effective key. So, we apply a permutation just a sapling and it will be again a 56 bit output, we will come to that what is the PC 1. Now we divide into 2 part, this is C 0, D 0 and each is 30 by sorry 20. So, 56 bit 28 bit each of this is 28 bit.

Now, will apply circular shift; left circular shift on separately on both this part. This is for the first round, you want to get first round key then we store it into what is called C 1 D 1 and then on this C 1, D 1 we apply another permutation P C 2 and this will give us the k 1 and to get k 2 what we do. So, we have C 1, D 1, so again we apply the circular shift, left circular shift.

So, this is basically left circular shift and it will it is either one time or two times depending on which round key we are looking for. So, it is a circular shift, left circular shift, but either 1 time or 2 time. Now we will come to that, so now, if we just after this we will store it into what is called C 2, D 2 and we apply this same function P C 2 and we get k t2.

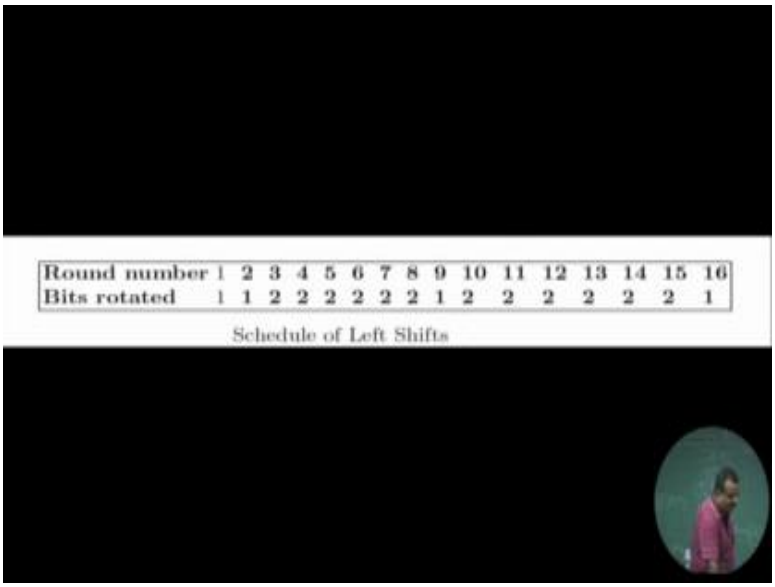
So, like this we continue and finally, we get k 16 we need to have 16 round keys, so now, we need to know what is the this PC 1 and P C 2. So, PC 1 is basically taking effective key 56 bit also this is the keys - keys alpha 1, alpha 2, alpha 7, alpha 8 is not there because alpha 8 is just (Refer Time: 29:34) alpha 9 like this.

(Refer Slide Time: 29:39)



So, let us go to the - what is the P C 1, so PC 1 is given in this table. So, it is just a random sapling of this 56 bit number, but we are removing the parity check bit; those are basically not the effective key.

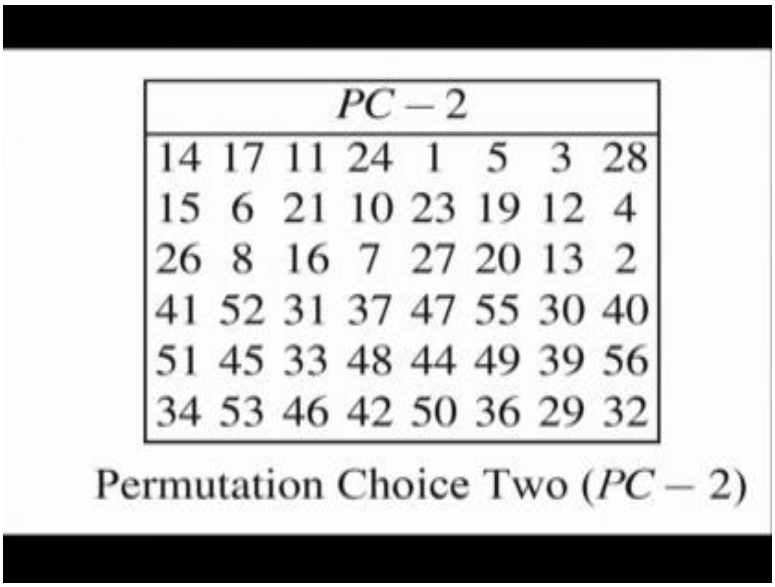
(Refer Slide Time: 30:02)



So, this is the PC 1 we are having and then in the P C two and this is the round; this is the left shift we are doing. Here we are doing the left circular shift for each of this, now this is for the first round. So, first round will do one time, for the second round also we will do one time, but for the third round, fourth round, fifth round this round will do

twice; circular shifts. So, these are the number of times we will do circular shift, we are in which round depending on that we will do this many circle; either 1 times or 2 times depending on which round keys we are looking for.

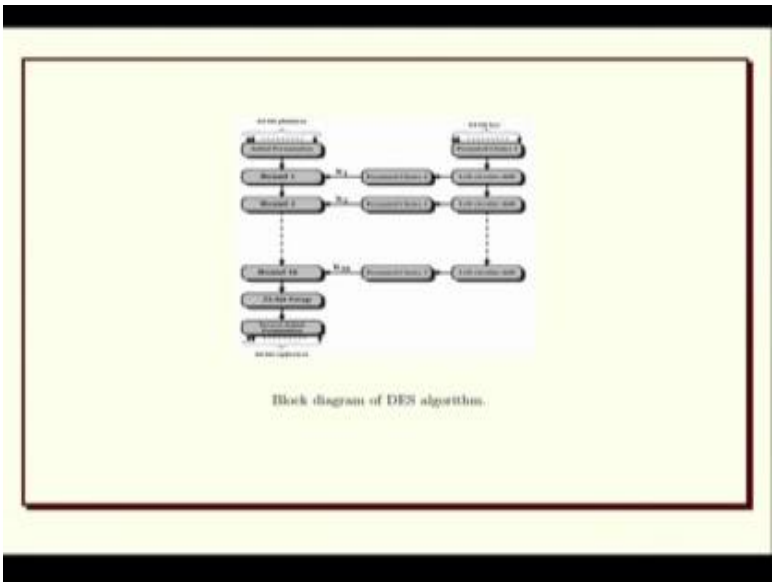
(Refer Slide Time: 30:02)



And this is basically typically P C 2; P C 2 is basically taking, so this is a input is basically again 56 bit, for the output is 48 bit.

So; that means, we need to remove some of the bits. So, this is the way how we are removing the some of the bits; please go to the slide. So, this is P C 2, so this is basically not taking all the bits. So, this is 1, 2, 3, 4, 5, 6, 1, 2, 3, 4, 5, 6, 7, 8, so this is a 6 quart cross 8 matrix, so there are 48 number. So, this is the way how we get this permutation from this 56 bit to 48 bit. So, this is basically alpha I mean the 14 bit, this is 17 bit like this. So, 28 bit then 15 bits, so some of the bits are must be missing here because we want to from 56 bit, we want to make it to 48 bit.

(Refer Slide Time: 31:52)



So, this is the overall diagram of this DES algorithm, this is the encryption algorithm. So, for encryption, so we are applying the first permutation then we are having the first round function; these are all round functions are basically Feistel cipher and then for each round function, we need to have this round key and this round keys are basically coming from the key scheduling algorithm, which we have discussed and then finally, after the 16 round we are applying the inverse permutation of this and we are getting the ciphertext, so this is the plaintext is the ciphertext, so this is the DES encryption. Now in the next class we will talk about DES decryption; how we can decrypt the DES.

Thank you.