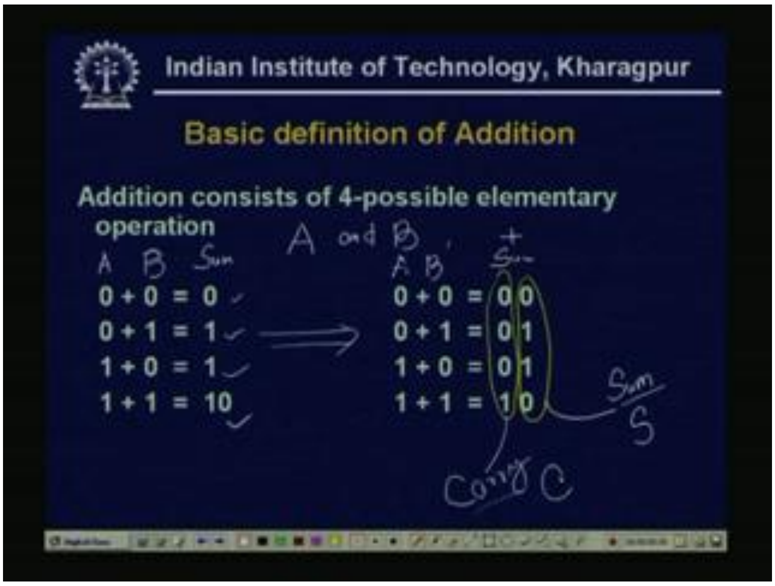
Digital System Design
Prof. D Roychoudhury
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture -13
Design of Adder Circuits

In the last class, we have read that, how we can design the combination circuits, what is the conventional design procedure and we have seen a comparator design. Now, today we will see how the common circuits such as some very popular circuits and very important circuits that different type of adders can be designed, because these adders are nothing but the combinational one.

(Refer Slide Time: 01:27)



Indian Institute of Technology, Kharagpur

Basic definition of Addition

Addition consists of 4-possible elementary operation

0 + 0 = 0
0 + 1 = 1
1 + 0 = 1
1 + 1 = 10

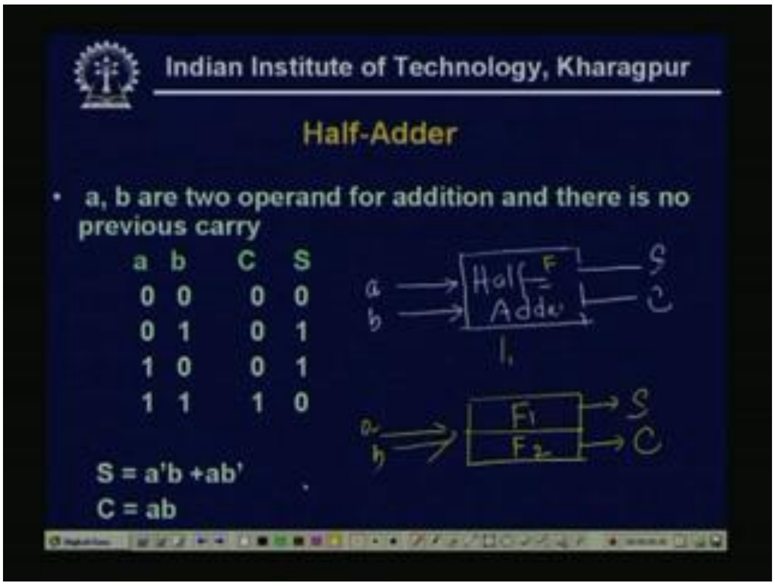
0 + 0 = 0 0
0 + 1 = 0 1
1 + 0 = 0 1
1 + 1 = 1 0

So, as it is name implies the addition consists of the adder, does the addition operator and this consists of four possible elementary operation and these operations, say there are two bits say this is. So, addition we are assuming there are two operands A and B, we are taking the addition, so I am giving the addition notation as plus, now this is my A value and B value.

Say, if it is 0 plus 0, then my sum means the output is 0, if it is 0 plus 1, the sum is 1. If it is 1 plus 0, it is 1; that means, these are binary addition and 1 plus 1 means 1 0, sum is 1 0. Now, if we can we just express this sum by two bits, because that 1 plus 1 is 1 0, the remaining 3, we can represent by single bit, so now if you represent, see this is my two bits, sum 0 0 0 1 0 1 1 0.

So, what will happened, say now say there are two bits, this is 1 LS, BS, I can tell this 4 and this is 1. So, if I add A and B, then if my sum is realized by only 1 bit, then these LSB the lowest, this bit; that means, at 0 1 1 0. This, I can define as my sum bit, normally note denoted as S and see these 0 0 0 1, these we are call that carry bits, normally noted by C.

(Refer Slide Time: 04:32)



Indian Institute of Technology, Kharagpur

Half-Adder

- a, b are two operand for addition and there is no previous carry

a	b	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

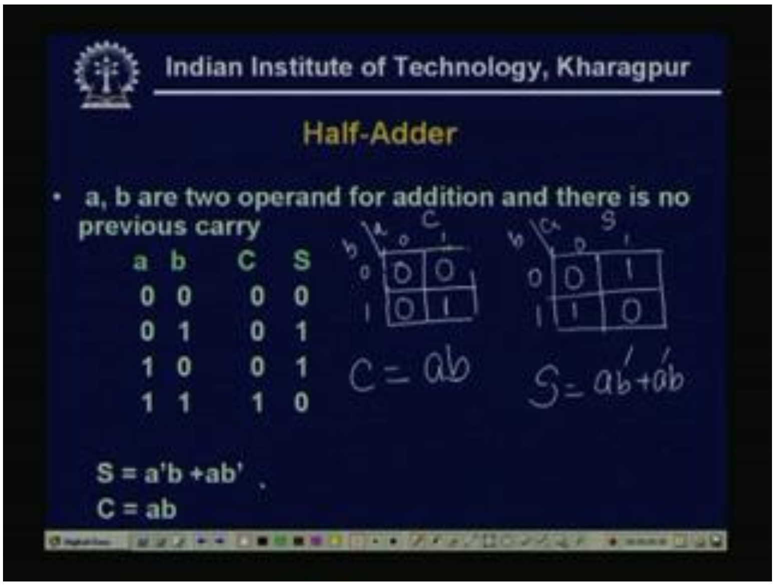
S = a'b +ab'

C = ab

So, given these problem of addition, now how we can write my adder, say if it is, I am considering 1 half adder and the half adder. We define as if there are two operands for addition and there is no previous carry; that means, only two values are added and the operator already operation, we have defined. So, here as if we are now identifying that, there are two operands, a b there are the inputs.

There are two outputs, 1 is my sum and 1 is my carry and this is my half adder I am telling, because I am not considering the previous carry bits, so now we can write in this way. So, again the way in the last class we have defined, so if the half adder is a 1 function, this F, again I can partition this thing by 2, because there are two outputs. So, I can define this as a F 1, F 2 and this a F 1 gives you S, sum F 2 gives C and as if both are fit by a b.

(Refer Slide Time: 06:27)



Indian Institute of Technology, Kharagpur

Half-Adder

- a, b are two operand for addition and there is no previous carry

a	b	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

C

b \ a	0	1
0	0	0
1	0	1

$C = ab$

S

b \ a	0	1
0	0	1
1	1	0

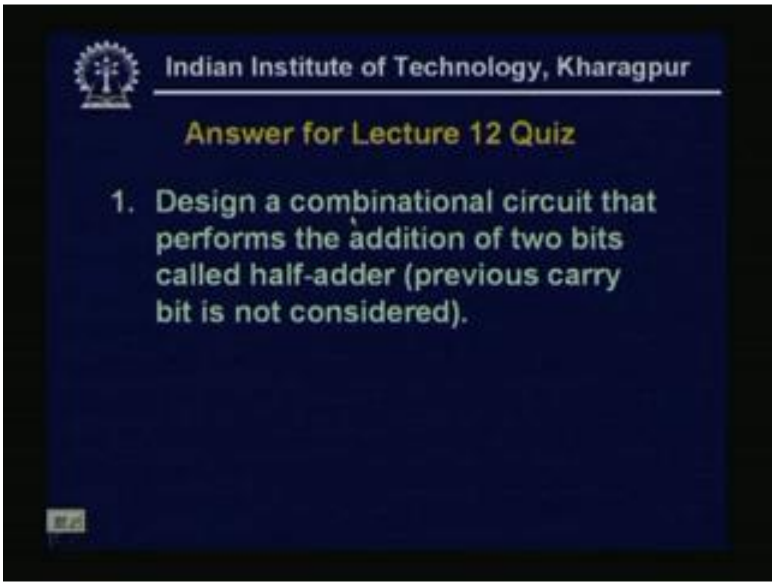
$S = ab' + a'b$

$S = a'b + ab'$

$C = ab$

So, now if we draw the maps, if we draw the maps, then what will happen say for the carry, so there are two inputs. So, there are two inputs say a b, say this is for my carry, then this is 0 1 0 1. So, for 0 0 carry is 0, 0 1 carry is 0, 1 0 carry is 0, 1 1 carry is 1, so only that C is equal to a b, it realizes. Similarly, if we draw the map for my S, what will happen, again I am taking a b this is for my S 0 1, 0 1, see for 0 0 sum is 0, 0 1 sum is 1, 1 0 sum is 1, 1 1 sum is 0. So, for this what will be there that a b dash, so my S equal to a b dash plus the for this, for this one the minterm is a dash b. So, that we have written that S is equal to a dash b plus a b dash and C is equal to a b.

(Refer Slide Time: 08:35)



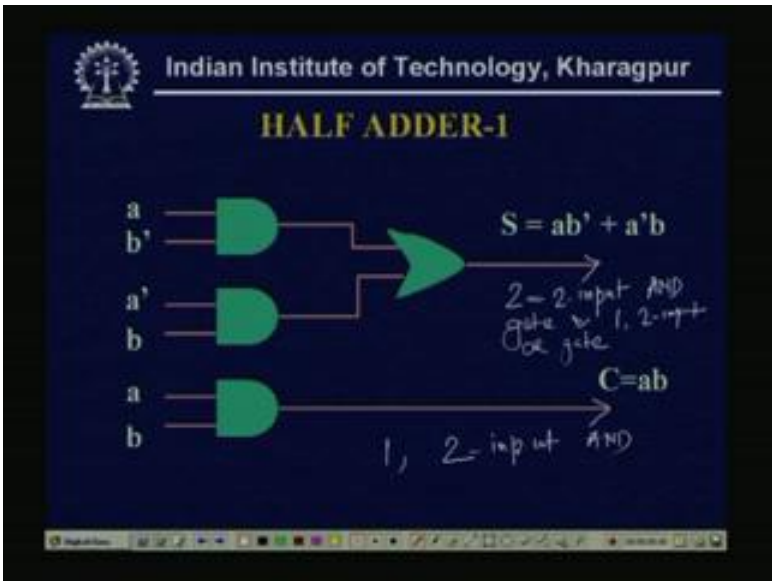
Indian Institute of Technology, Kharagpur

Answer for Lecture 12 Quiz

1. Design a combinational circuit that performs the addition of two bits called half-adder (previous carry bit is not considered).

So, this is my half adder circuit and remember that was our last quiz question. That design a combination circuit, that performs the addition of two bits called half adder and previous carry bit is not consider, mainly there are two inputs.

(Refer Slide Time: 08:50)



Indian Institute of Technology, Kharagpur

HALF ADDER-1

$S = ab' + a'b$

2 – 2-input AND gate & 1, 2-input OR gate

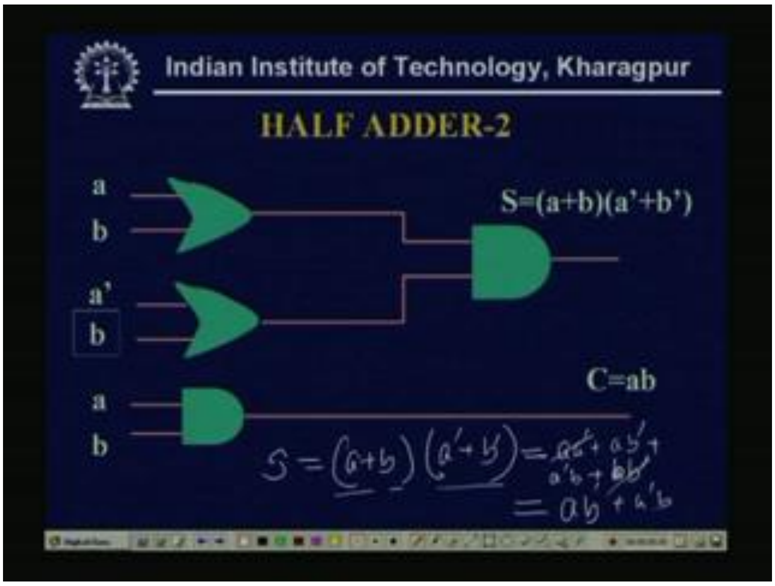
$C=ab$

1, 2-input AND

So, if we realize the realization is very simple because a b dash plus a dash b and now again assuming, that my complimented as well as uncomplicated variables are literals are available. Then, this is a 2, two input AND gate and 1, two input OR gate, so my S is that to a b dash and a dash b and there are OR, answer this is my S and carry a simple C is equal to a b.

So, in the just to generate the sum, there are 2, two input AND gate 2 two number of two input AND gate and 1, two input OR gate and in the carry part only 1, two input AND gate, that is my circuit. Now, we see that, this is very simple circuit, that where we have done that a b dash plus a dash b.

(Refer Slide Time: 10:30)



Indian Institute of Technology, Kharagpur

HALF ADDER-2

S=(a+b)(a'+b')

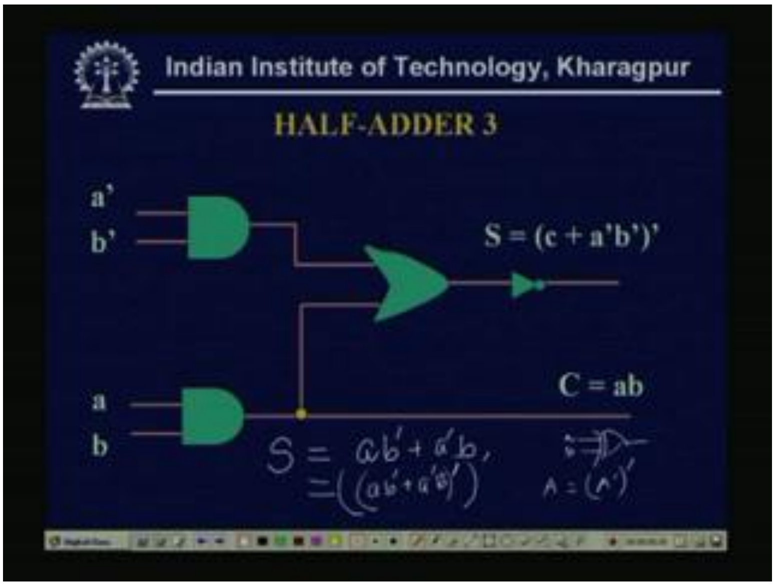
C=ab

$S = (a+b)(a'+b') = \cancel{aa'} + ab' + a'b + \cancel{bb'}$

$= ab' + a'b$

Now, this a b dash plus a dash b, that we can write in different way, see that I have written my S was a b dash plus a dash b. Now, see I have written a plus b into a dash plus b dash, say if I apply the distributive law, then this is a dash plus a b dash plus a dash b plus b b dash, now a dash and b b dash, that will be 0. So, this becomes a b dash plus a dash b, so that mean a b dash plus a dash b, that I can write as a plus b into a dash plus b dash. Means again, this is 1 AND gate 2 OR gate, which is the reverse of the previous 1, In the previous circuit, it was 2, two input AND gate 1, two input OR gate.

(Refer Slide Time: 12:20)



Indian Institute of Technology, Kharagpur

HALF-ADDER 3

S = (c + a'b')'

C = ab

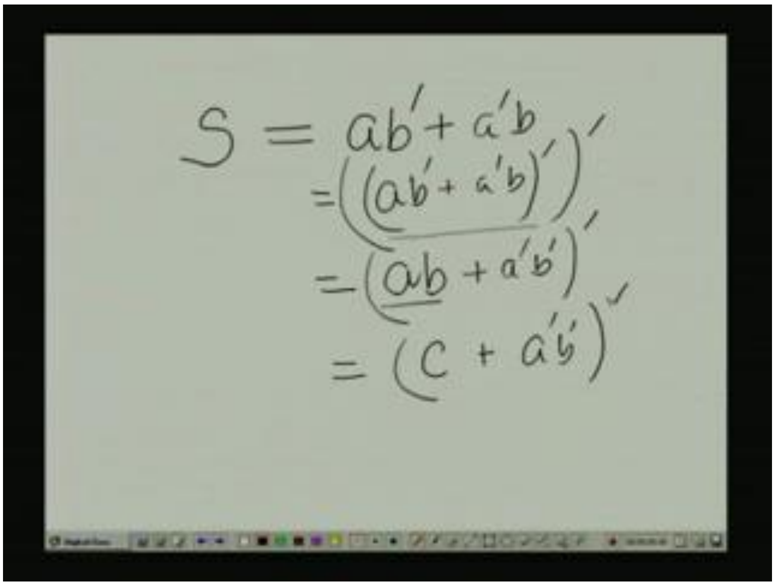
$S = ab' + a'b,$

$= ((ab' + a'b)')'$

$A = (A')'$

Now, I am realizing 2, two input OR gate and 1, 1 input AND gate, see that a plus b and this is a dash plus b dash 2, two inputs OR gate and then 1 into two input OR gate and carry remains the same, so the total number of gates are same. So, this is a different type of realization of the half adder, now we see that, my carry is a b, my carry C is a b and my sum was my sum was a b dash plus a dash b. And that, I can write because see a b dash plus a dash b is nothing but two input EXOR a b. So, that I can write, that a b dash plus a dash b dash, dash means, we know that a equal to means that, this is the twice I have taken, that compliment value.

(Refer Slide Time: 14:02)

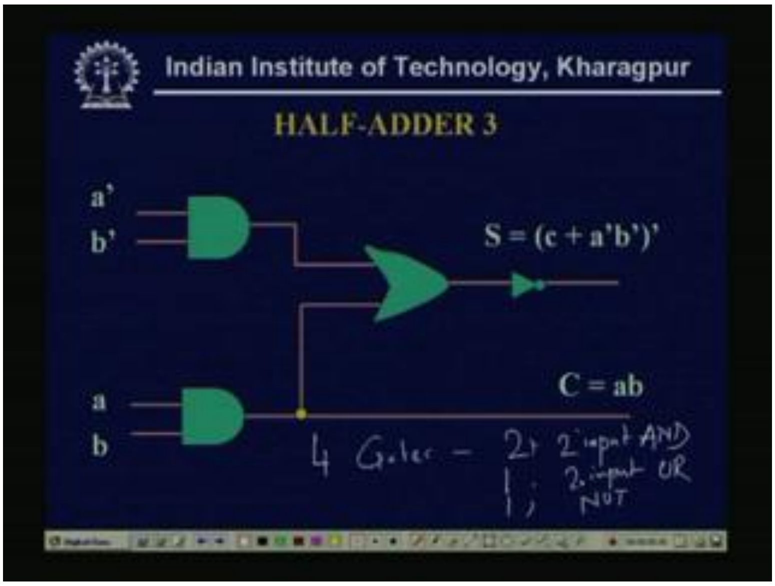


$$S = ab' + a'b$$
$$= ((ab' + a'b)')'$$
$$= (\underline{ab} + a'b')'$$
$$= (C + a'b')'$$

So, what will happene, that means my S sum, again if we write this is a b dash plus a dash b and this is I can write a b dash plus a dash b of complement. If I compute this, this is nothing but XNOR and we know two inputs XNOR is a b plus a b plus a dash b dash that compliment of that. Now, see this a b means, this is my C the carry and this is 1 a dash b dash and compliment of that.

Now, this C is the same that thing as been realized, so this is the carry as it is that a b carry is there. Now, carry I am taking as the one of the input and the other input is coming from a two input AND gate a dash b dash. So, again it is fed to the OR gate, so they see this is C plus a dash b dash and now it is complemented, so one inverter is put.

(Refer Slide Time: 15:24)



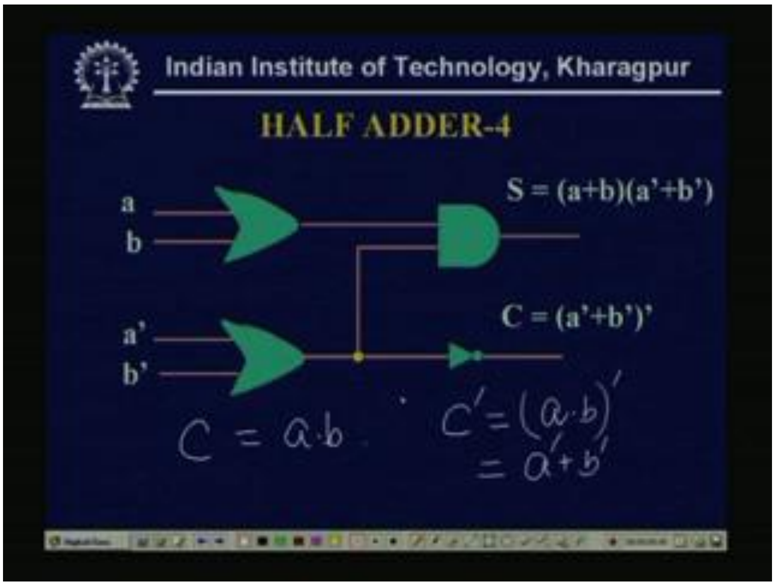
Indian Institute of Technology, Kharagpur

HALF-ADDER 3

Now, see here only 4 gates are needed, if we do this type of realization, then my half adder circuit takes only 4 gates, that 2, two input AND gates 1, two input OR and 1, NOT. So, totally inputs are only 4, initially it was 17 inputs, always we are assuming in the all the circuits, we are assuming that the variables are available both in complemented and uncomplimented form.

So, again this type of realization, that number of gates are very much reduced and this are only 4 basic gates, whereas in the previous design, see that the gates, again they are 4 gates, but the different type of gates. Here, we have used that one inverter, we have used and AND gates 1 OR gate or 1 AND gate, that we are we have reduced.

(Refer Slide Time: 16:58)



Indian Institute of Technology, Kharagpur

HALF ADDER-4

$S = (a+b)(a'+b')$

$C = (a'+b')'$

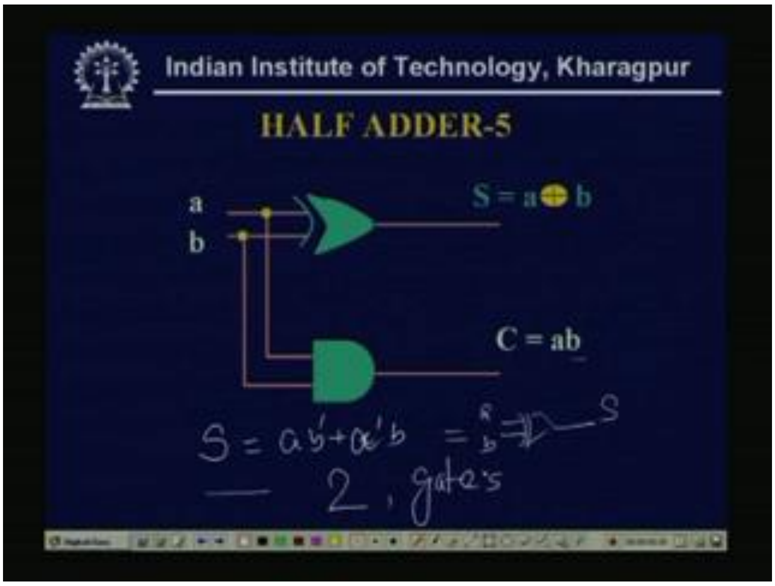
$C = a \cdot b$

$C' = (a \cdot b)'$
$= a' + b'$

Now, this is again another realization, the sum is same, that a plus b into a dash b dash, already we have discussed this some part. Now, carry is a dash plus b dash, how this is simple that carry was a b, so if we apply the De Morgan's law, then C dash is a dot b. So, this C dash is a dot b dash and it will be a dash plus b dash, so thus that thing we have done.

That C is equal to a dash plus b dash, see here what happen that only 2 OR gate, 1 AND gate 1 inverter. That means the difference mainly AND, OR and NOT these basic gates, we are utilizing and only the numbers of OR or number of AND are varying.

(Refer Slide Time: 18:09)


Indian Institute of Technology, Kharagpur

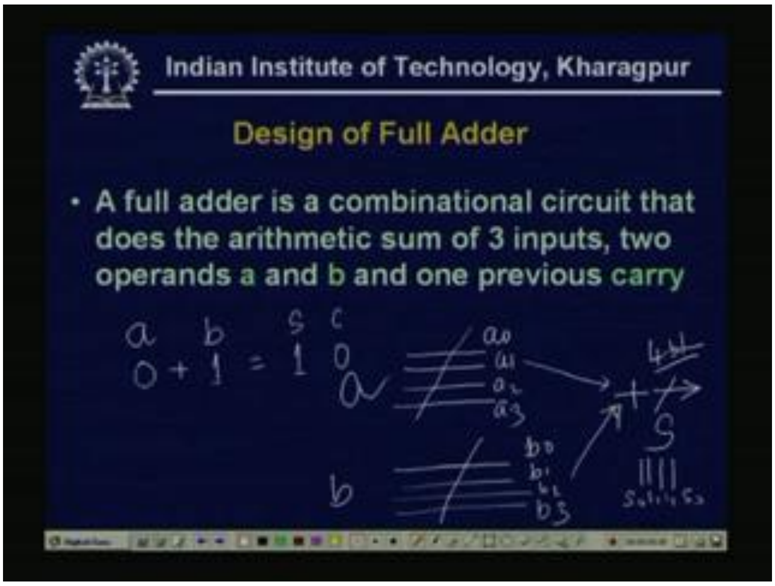
HALF ADDER-5

Now, this is another such real addition possible see what my expression was for that sum that is nothing but a b dash plus a dash b, which is and two inputs EXOR, that a b and this is my sum. So, that is realized by a simple EXOR gate and a b, that same input I am taking then C is equal to a b. Now, what is the advantage of that thing or disadvantage, see first thing is here there is no need, that that complemented variable should also exist, only we have taken the un complemented variables or the literals.

And they are carry simple a b and 1 EXOR function, so the only there are two gates, but just notice that here that this EXOR is not a basic gate, that this is a actually that AND, OR, sum AND, OR realization is needed for this EXOR. But, now for this MSI circuits, that actually that we can realize the circuits by EXOR, XNOR or multiplexer or different type of many other, that ROMs that is also nowadays available.

So, if mine MSI circuits, MSI means that medium scale integrated circuits, that where instead of the basic gates, that AND, OR, NOT that EXOR gates EXOR, XOR, NOR also available. Then, we can easily utilize or we can easily use this type of circuit for half adder and see there are only two gates are needed for that, so this minimum thing, that only two gates are necessary.

(Refer Slide Time: 20:27)



Indian Institute of Technology, Kharagpur

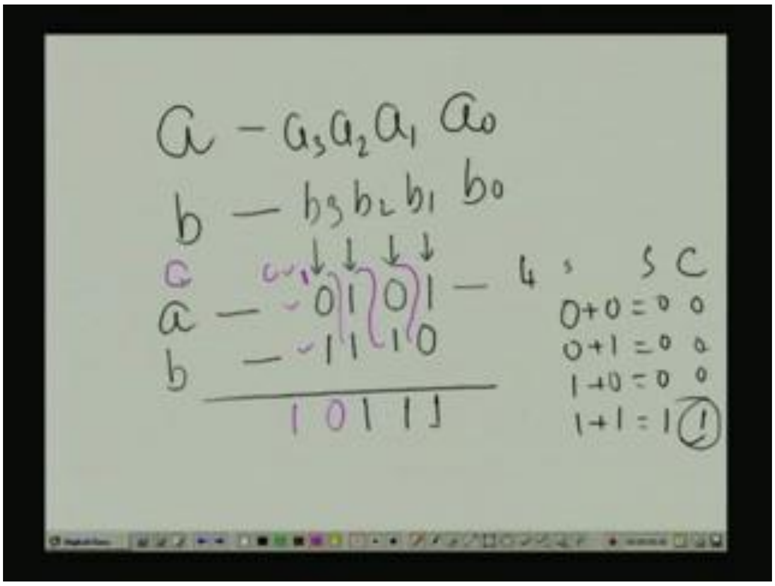
Design of Full Adder

- A full adder is a combinational circuit that does the arithmetic sum of 3 inputs, two operands a and b and one previous carry

Now, we see design of a full adder, now a full adder is a combination circuit, that does the arithmetic sum of three inputs, two operands a and b and one previous carry. Now, what do we mean by this, see earlier in the half adder, we have only taken two operands a b, this is 0, a is 0, say b is 1 and it is adding 1, sum is 1 and carry is 0, now this is simple two input or you can take single adder.

Now, what we have a b, I am adding, again I am adding a, but this a as 4 bits, if this is a 0, a 1, a 2, a 3. Similarly, these are b sum giving b 0, b 1, b 2, b 3 and as if I am adding, if I am adding this, so obviously, it will generate a 4 bit sum that means, S 0, S 1, S 2, S 3, so this type of circuit. Now, we are defining in this case, actually the half adder will not be sufficient.

(Refer Slide Time: 22:55)

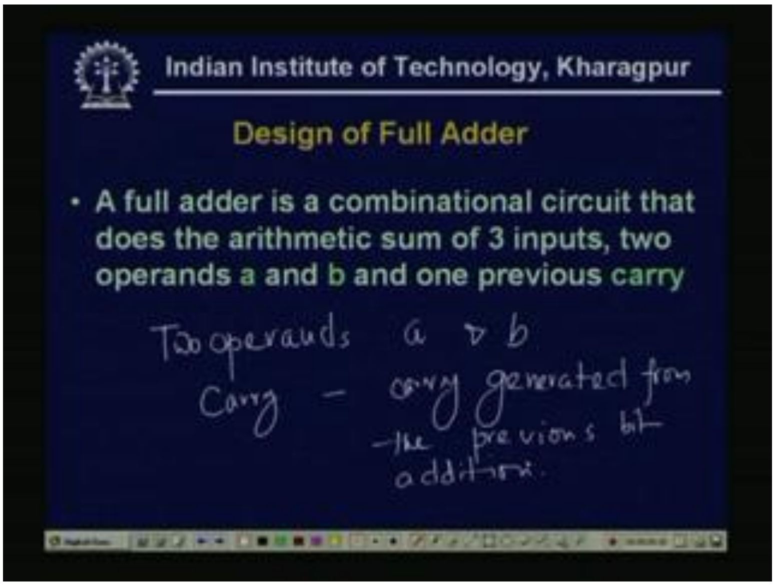


Because, if we take one example now, see my a is 4 bit, a 0, a 1, a 2, a 3 actually we should write in different order LSB first, a 0, a 1, a 2, a 3. Similarly, b is b 0, b 1, b 2, b 3, say I am taking now some values, say a is 0 1 by b is 1 1, 1 0. Now I am adding, now see I can treat or I can take this, if four single input adder, because these are 4 or single in 1 input adder or two input adders.

That means here, one single input of a 0 and b 0, I am adding, so this is 0 plus 1 1, now, already we have defined, that if it is 0 plus 1, 0 plus 0 is 0, 0 plus 1 is 0, 1 plus 0 is 0, 1 plus 1 is 1, carry 1, here all the carries are 0. So, see now here there is no carry, so it is not important. Now, 0 plus again is 1, 0 plus 1 carry 0, now 1 plus 1, this say the sum is 1 and there is 1 carry also, so I have to this carry will go here. So, when I will be adding, now actually there are three inputs, say this is my b value, a value and this is my carry value.

So, there are actually instead of two inputs, there are three inputs, so this is again 1 plus 0, 1 and 1 plus 1, this is 0 again carry 1, so this carry 1 will come here. Now, here the carries are 0, but actually that where that I have to consider, now all the carries will go here. So, instead of two inputs, now for the full adder, I have to consider three inputs, so that is why the half no full adder is defining as a two operands and the previous carry. So, C we are defined, this is the previous carry, means the carry generated from the previous bits.

(Refer Slide Time: 26:25)



Indian Institute of Technology, Kharagpur

Design of Full Adder

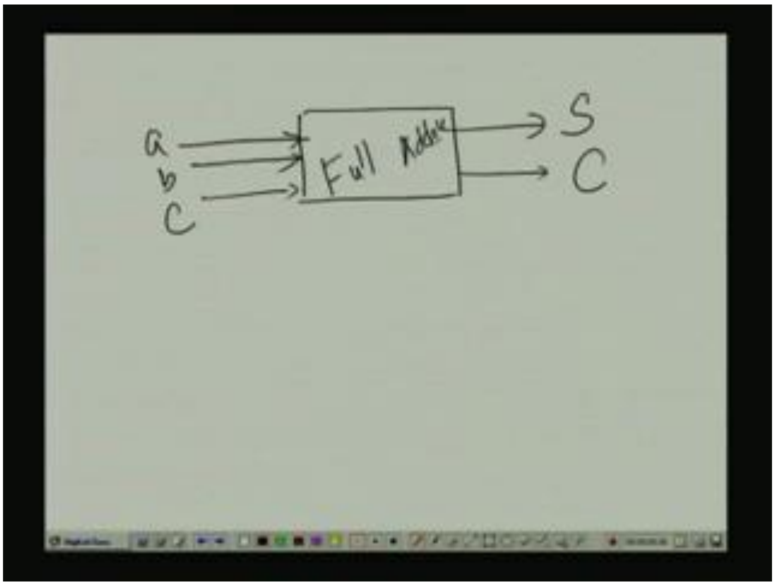
- A full adder is a combinational circuit that does the arithmetic sum of 3 inputs, two operands a and b and one previous carry

Two operands a & b

Carry - carry generated from the previous bit addition.

So, this we can tell that the two operands, a and b and the carry is actually the carry generated from the previous bit operation, bit addition. Notice here, that in the half adder we have not considered that thing. So, half adder means mainly the single bit addition and here actually the multi bit addition, so now we see that, how we can design the adder.

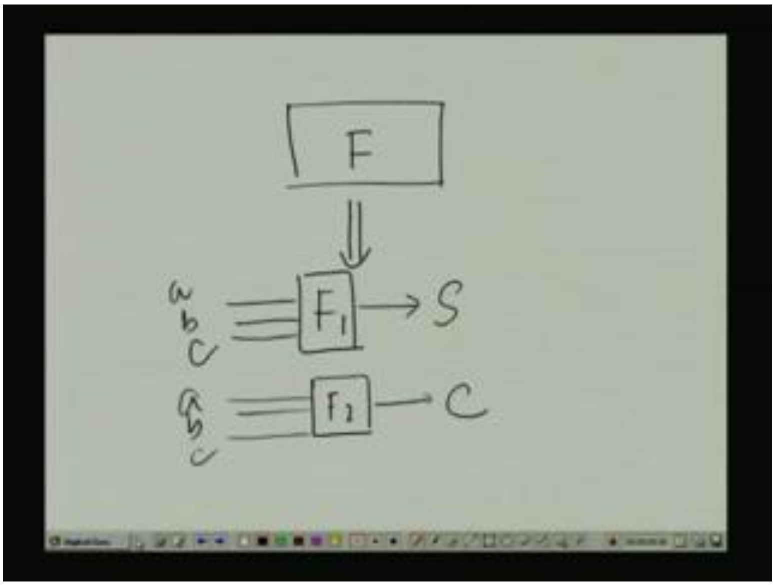
(Refer Slide Time: 27:29)



So, once it is the system is identified, again we can tell that my system is now, this is a black box, there are two operands a b and this is my previous carry, this is my sum and this is my carry, means the mixed carry or the carry to the mixed bit.

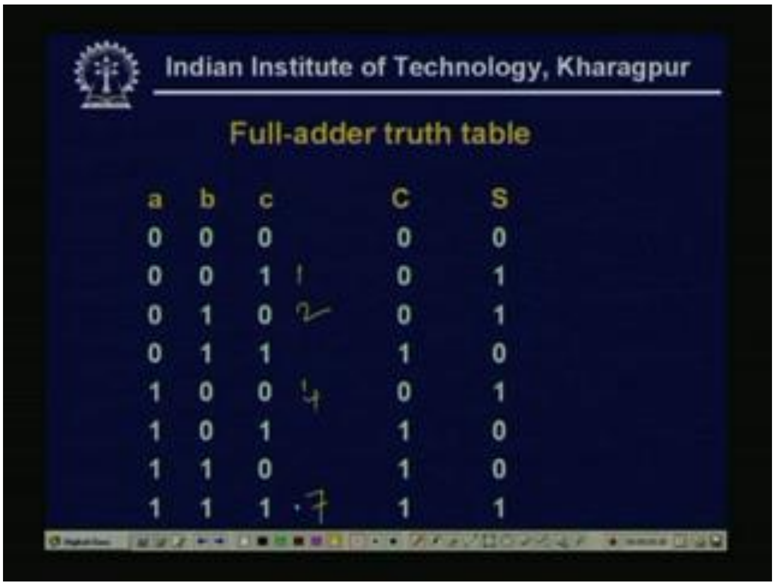
So, now we identify the output, input relationship, see that these two are the operands a b and this is the carry. So, 0 0, if the carry is 0, then again 0, 3 0 actually we are adding and sum is 0 carry is 0, 0 0 1 means sum is 0, carry is 1, carry is 0, sum is 1, 0 1 0 sum is 1, carry is 0, similarly 0 1 1 means, now 1 plus 1, so sum is 0, carry is 1. So, in this way for all possible combinations of three input, means the 8 combinations will be there, that we have found out, what will be carry and the sum.

(Refer Slide Time: 29:07)



Now, again here they actually the F; that has been partitioned into F 1 sum and F 2 carry a b c are the inputs. So, F 1 means that a b c are the three inputs and the carries the output and a b c are the three inputs and sum is the output, that is F 2.

(Refer Slide Time: 27:19)



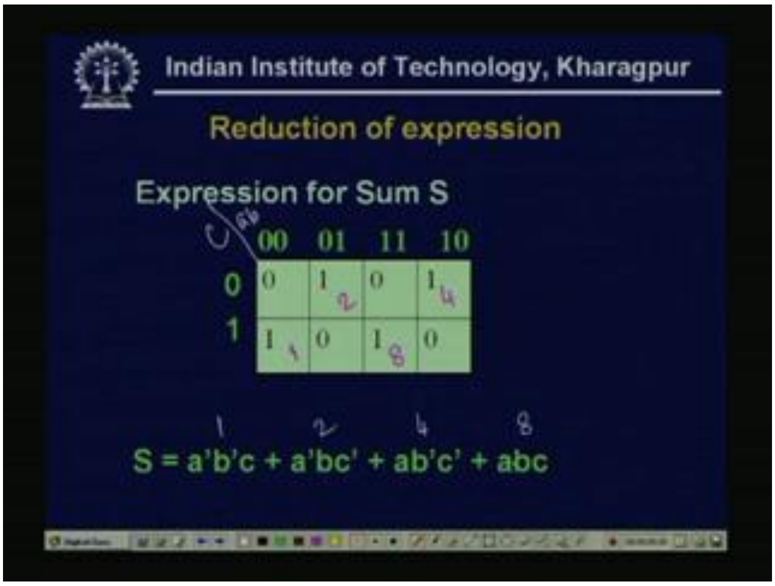
Indian Institute of Technology, Kharagpur

Full-adder truth table

a	b	c		C	S
0	0	0		0	0
0	0	1	1	0	1
0	1	0	2	0	1
0	1	1		1	0
1	0	0	4	0	1
1	0	1		1	0
1	1	0		1	0
1	1	1	7	1	1

So, again now we draw the Karnaugh map, so first we draw the expression for sum S. Now, again that if we see the sum that where it will be 1, it will be 1 for the decimal value.

(Refer Slide Time: 29:07)



Indian Institute of Technology, Kharagpur

Reduction of expression

Expression for Sum S

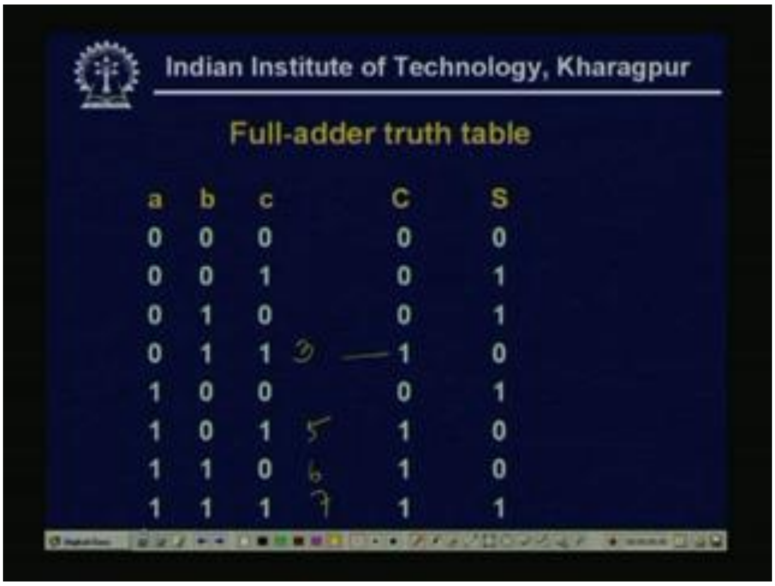
c \ ab	00	01	11	10
0	0	1 2	0	1 4
1	1 1	0	1 8	0

$$S = \overset{1}{a'b'c} + \overset{2}{a'bc'} + \overset{4}{ab'c'} + \overset{8}{abc}$$

That 1, then 2, 4 and 7, so for 1, 2, 4, 7, see for 1, if I take in this way, so it give this is my a and this is b, then actually this should be a b c, there are three inputs, so this should be a b, this should be c. So, these will be the 1, this is the 2 0, 1 0, then this is 4, this is 8, so for this 4, the output will be 1; that means, 1 sum bit will be 1 and the remaining the sum bit will be zeros.

So, if we now take the minterms for this 1 value; that means, for where 1 is there, that minterm will exist in the expression. Then, it will be 0 1 0 means, that a dash first we take a dash b dash c, means this 1, for 1 this is for my minterm a dash b dash c, then a dash b c dash, this is for the minterm 2. Then, a b dash c dash, this is for minterm 4, a b c for minterm 8, so these are this is the expression for the sum. And here, we see there are no couples possible, so there is no reduction from this, now this is the expression for sum.

(Refer Slide Time: 33:14)



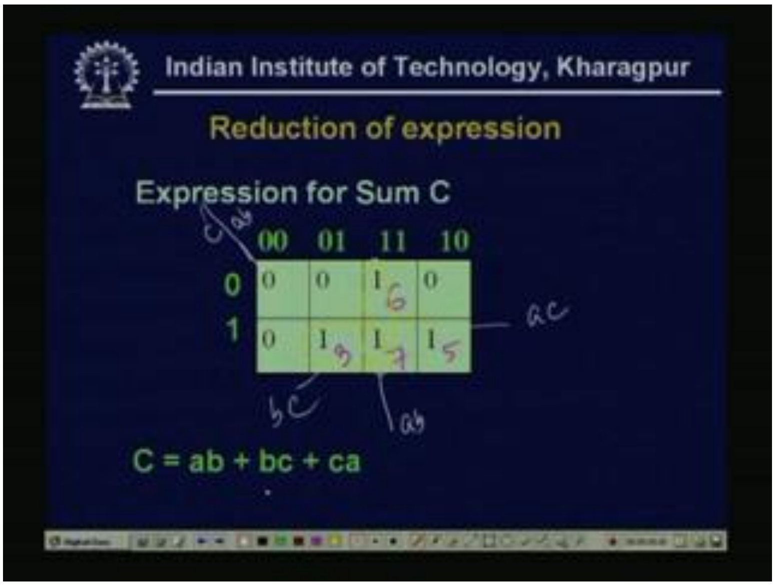
Indian Institute of Technology, Kharagpur

Full-adder truth table

a	b	c		C	S
0	0	0		0	0
0	0	1		0	1
0	1	0		0	1
0	1	1	3 —	1	0
1	0	0		0	1
1	0	1	5	1	0
1	1	0	6	1	0
1	1	1	7	1	1

Now, what will the carry, again if we see that from the truth table, that for carry for which value that carry will be say this is for 3 carry will be 1, then 5, 6, 7.

(Refer Slide Time: 33:36)



Indian Institute of Technology, Kharagpur

Reduction of expression

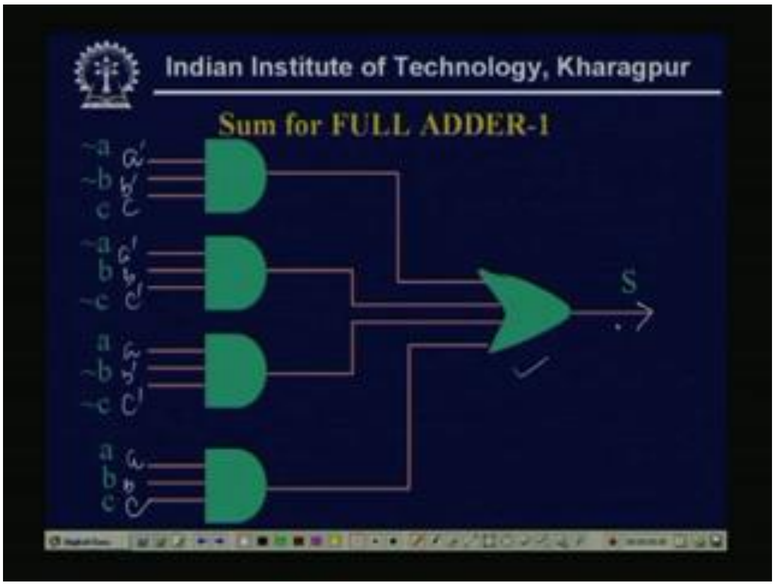
Expression for Sum C

c \ ab	00	01	11	10
0	0	0	1	0
1	0	1	1	1

C = ab + bc + ca

So, C 3, 5,6,7 again if I put a b c, so this is 3, then 1 1 0 means 6,7,5 so this 4, 3, 5,6,7 the output will be 1 and remaining cases, it will be 0, then what will be the mean terms, the mean terms will be say for here, actually there is 1 couple possible. So, this will be a b, again this will be ac and this will be b c, so C is; that means, for this column, it is a b for this couple, it is a c, for this couple this is b c. So, a b plus b c plus c a, so these are carry bit, now for the carry bit, now we have to see that, what will be the design.

(Refer Slide Time: 34:57)

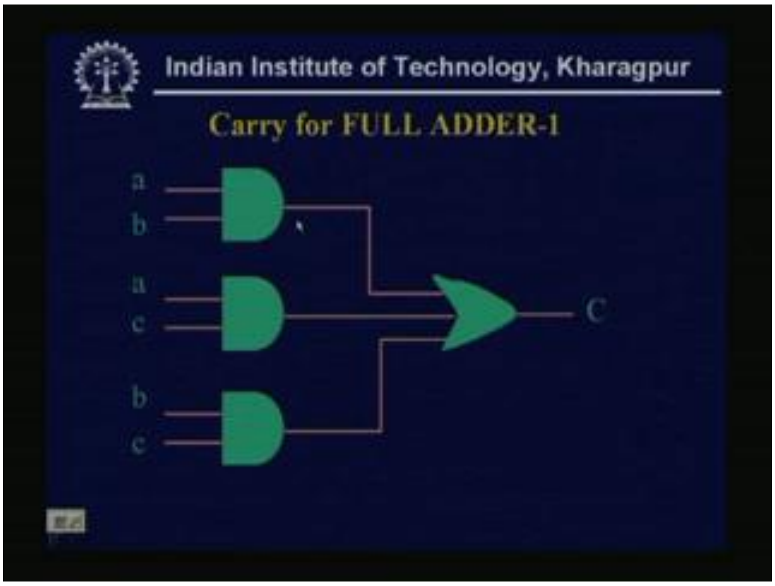


Indian Institute of Technology, Kharagpur

Sum for FULL ADDER-1

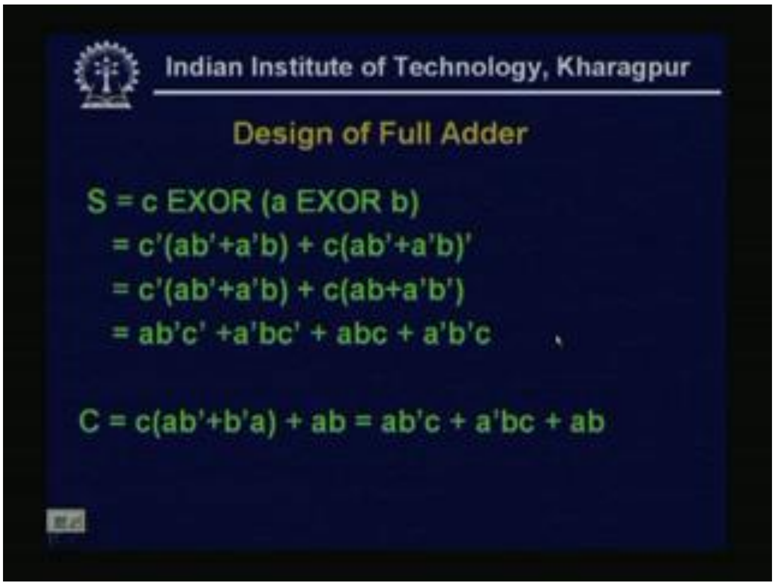
Because, now see the sum was how many 1, 2, 3, 4, 4, 3 input AND gate and 1, 4 input OR gate. Now, see... So, these are the 4 AND gate three inputs AND gate and it is inputs are actually a dash b dash c, a dash b is c dash, a b dash c dash and a b c. That means both complimented and uncomplicated variables are available and 1, 4 input OR gate, so this gives a sum.

(Refer Slide Time: 36:04)



Now, the carry will be simple, that a b plus b c plus c a, so the carry that 3, two input AND gate and 1 three input OR gate, that will give the carry. Now, there can be other realization, see here there was no reduction possible from the Karnaugh map.

(Refer Slide Time: 36:37)



Indian Institute of Technology, Kharagpur

Design of Full Adder

S = c EXOR (a EXOR b)

$= c'(ab'+a'b) + c(ab'+a'b)'$

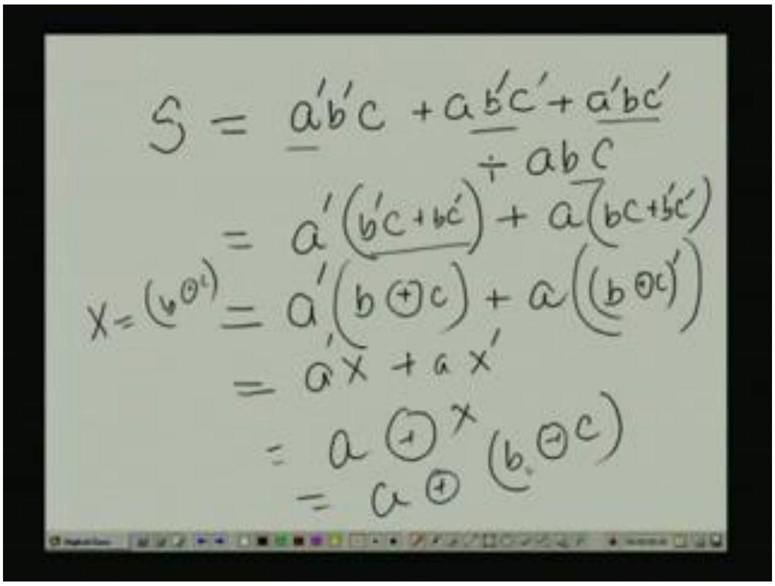
$= c'(ab'+a'b) + c(ab+a'b')$

$= ab'c' + a'bc' + abc + a'b'c$

$C = c(ab'+b'a) + ab = ab'c + a'bc + ab$

Now, what is actually that sum circuit, see my sum was a dash b dash c, a dash b c dash, a b dash c dash plus a b c.

(Refer Slide Time: 37:07)



$$S = a'b'c + ab'c' + a'bc' + abc$$

$$= a'(b'c + bc') + a(bc + b'c')$$

$$X = (b\oplus c) \quad = a'(b \oplus c) + a((b \oplus c)')$$

$$= a'X + aX'$$

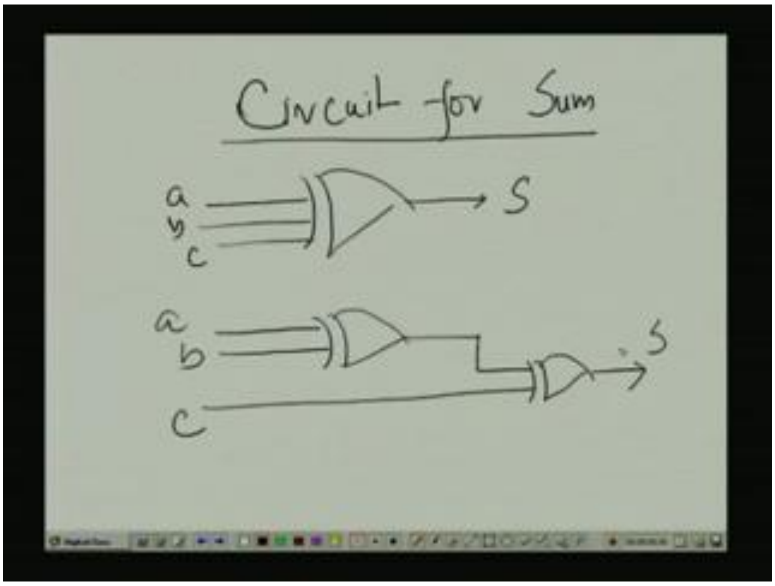
$$= a \oplus \overset{(b \oplus c)}{X}$$

$$= a \oplus (b$$

So, if we take that some expression, see if we write the sum expression a dash b dash c, a b dash c dash, a dash b c dash plus a b c. Now, say from the first and the third term, say I am taking a dash common, then similarly this is b dash c plus b c dash and from the second and 4th, if I take a common, then this is b c plus b dash c dash. Now, already we have seen from the previous one example from the last class, that this is actually b EXOR c.

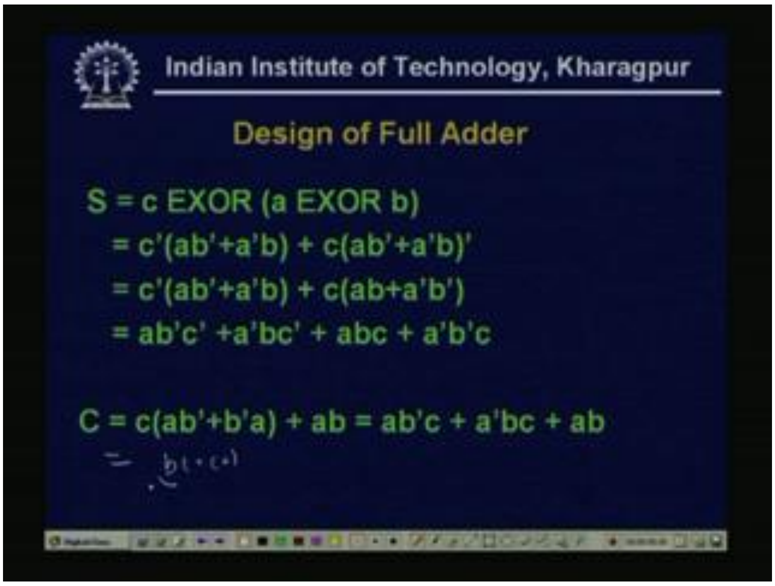
And this example is actually XNOR; that means, b EXOR c complement of that thing, now see b EXOR c, I replace as X b EXOR c, we replace by X then this is nothing but a dash X plus a X dash, which is again a EXOR X and that will be a EXOR b EXOR c. So, that means, my sum can be a simple a three input EXOR circuit or that three inputs EXOR, we can realize by 2 two input XOR.

(Refer Slide Time: 39:07)



So, what will be circuit that means my sum, circuit for my sum will be say simple I take a three input EXOR a b and the carry c. This is my sum or it can be 2, two input, that a b and c, then S, so this can be a sum circuit.

(Refer Slide Time: 40:09)



Indian Institute of Technology, Kharagpur

Design of Full Adder

S = c EXOR (a EXOR b)

= c'(ab'+a'b) + c(ab'+a'b)'

= c'(ab'+a'b) + c(ab+a'b')

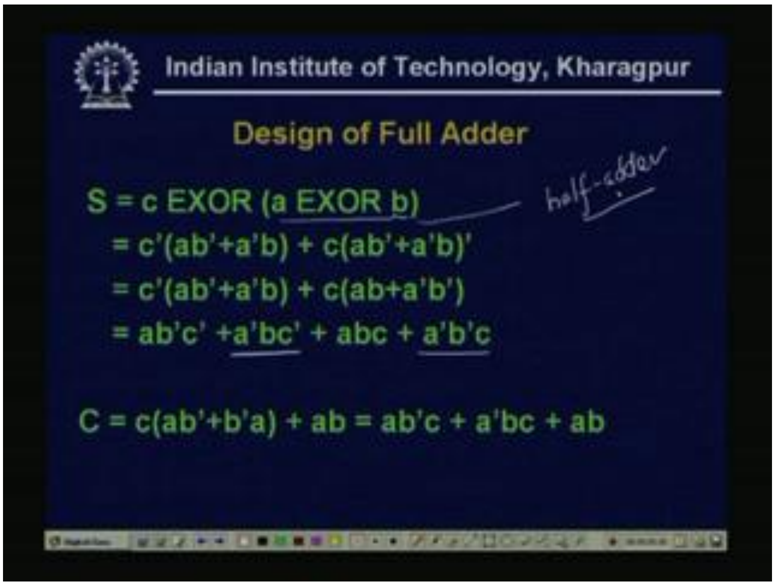
= ab'c' +a'bc' + abc + a'b'c

C = c(ab'+b'a) + ab = ab'c + a'bc + ab

= b(+ c+)

Now, see that c EXOR a EXOR b, already we have written that thing, also that we can write, already we know that, we got this thing. And then, that C we can write as carry C that is a b plus b c plus c a, see here how we can write actually this is a b and this is b c plus ca. So, that also we can write in this way, that that c a, b c, a b dash plus b dash a and that is also a b dash c plus a dash b c plus a b, in this way also we can write my carry.

(Refer Slide Time: 41:30)



Indian Institute of Technology, Kharagpur

Design of Full Adder

$S = c\ \text{EXOR}\ (\underline{a\ \text{EXOR}\ b})$ half-adder

$= c'(ab'+a'b) + c(ab'+a'b)'$

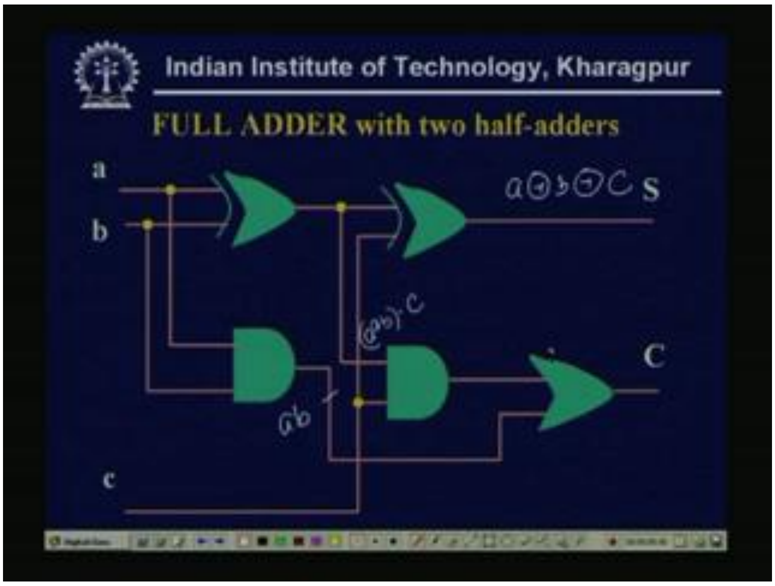
$= c'(ab'+a'b) + c(ab+a'b')$

$= ab'c' + \underline{a'bc'} + abc + \underline{a'b'c}$

C = c(ab'+b'a) + ab = ab'c + a'bc + ab

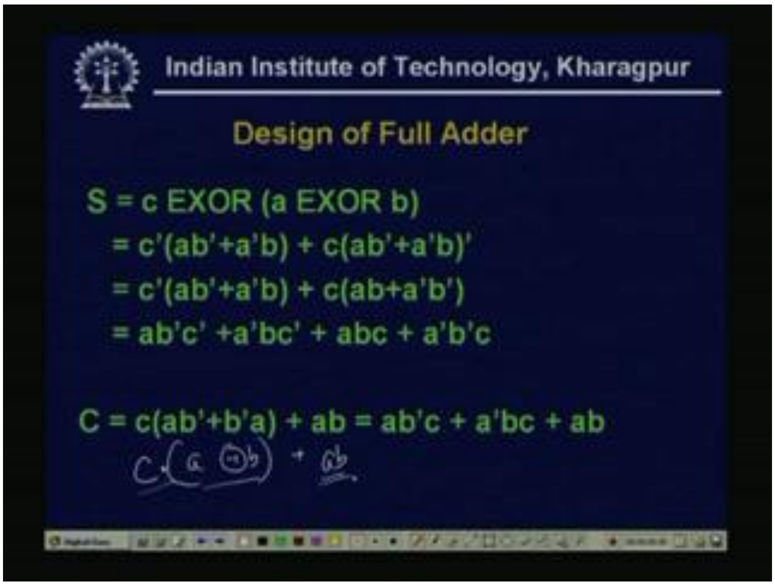
So, now if we design this thing, see that from here actually I can take this thing see my half adder was simple EXOR, Or that was see again, we go back to our half adder, see our half adder, it was simple a EXOR b and the carry was a b. That was the half adder circuit, because it was the truth table. So, if we from here, we identify that thing, see this is from here a dash b c dash and a dash b c dash, this is nothing but EXOR circuits. That means, this is c EXOR a EXOR b, now a EXOR b, I can write that at this is a half adder circuit.

(Refer Slide Time: 43:43)


Indian Institute of Technology, Kharagpur
FULL ADDER with two half-adders
a
b
c
$a \oplus b \oplus c$
S
$(a \oplus b) \cdot c$
ab
C

So, now we can realize that full adder with 2 half adders, so see that thing that this is a a EXOR b and now a EXOR b, again another half adder, that means another a EXOR. And one input is the carry itself; that means, we can write this thing is nothing but my S is a EXOR b EXOR c, what just not I have draw for the first circuit. Now, for the carry C, again this is my a b and for the remaining portion see this is a EXOR b, this is a EXOR b and that is ended with c and it is simple OR

(Refer Slide Time: 45:17)



Indian Institute of Technology, Kharagpur

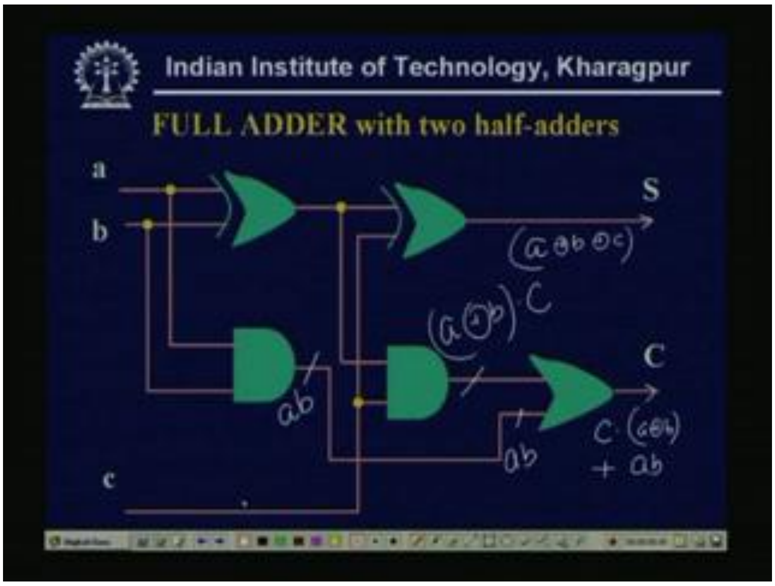
Design of Full Adder

$$S = c \text{ EXOR } (a \text{ EXOR } b)$$
$$= c'(ab'+a'b) + c(ab'+a'b)'$$
$$= c'(ab'+a'b) + c(ab+a'b')$$
$$= ab'c' + a'bc' + abc + a'b'c$$

$$C = c(ab'+b'a) + ab = ab'c + a'bc + ab$$
$$c.(a \oplus b) + ab$$

So, just remember the previous one previous1, that see this is c this is actually the c and this is a EXOR b with a b. So, see this is C AND, here 1 AND gate is needed, one, two input EXOR gate is needed and for this a b 1, again this is a two input AND gate.

(Refer Slide Time: 46:04)



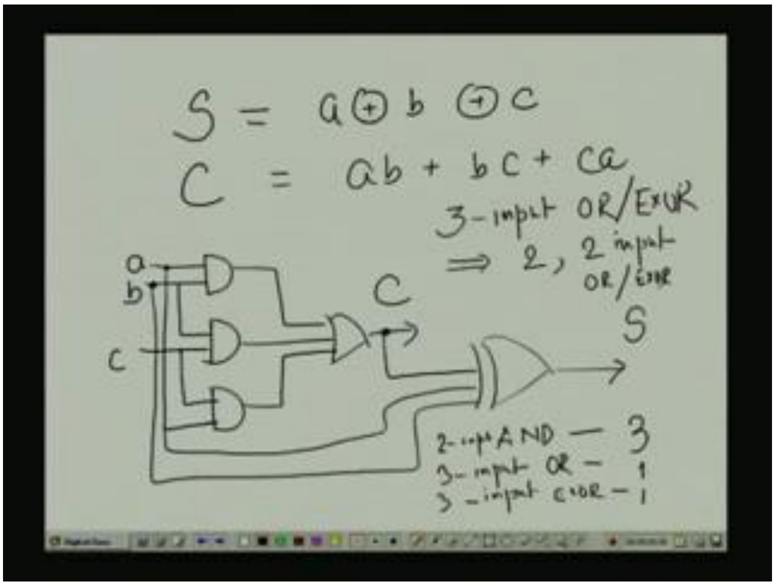
Indian Institute of Technology, Kharagpur

FULL ADDER with two half-adders

So, what we have done, say this C that this equation is mainly doing that thing, so this is a b, actually this AND input is a b, and this AND in output, this is one input is a EXOR b ended with c. And this is a b, so this c is C dot a EXOR b plus a b, so this is one realization and this S is the a EXOR b EXOR c. Now, there can be many other realizations possible for this full adder circuit.

Now, again that what we have discussed for the half adder circuit, that is true, that what will be what type of gates are available for my realization, that can be a the traditional AND OR gate realization, Or that can be that multi inputs gates. Even, it is a AND, OR gate that can be multi input gate or it is a 2 two 2 or three input gate or the EXOR, EXNOR gates with the inverters, also may or may not be there with that realization, it is possible.

(Refer Slide Time: 48:07)



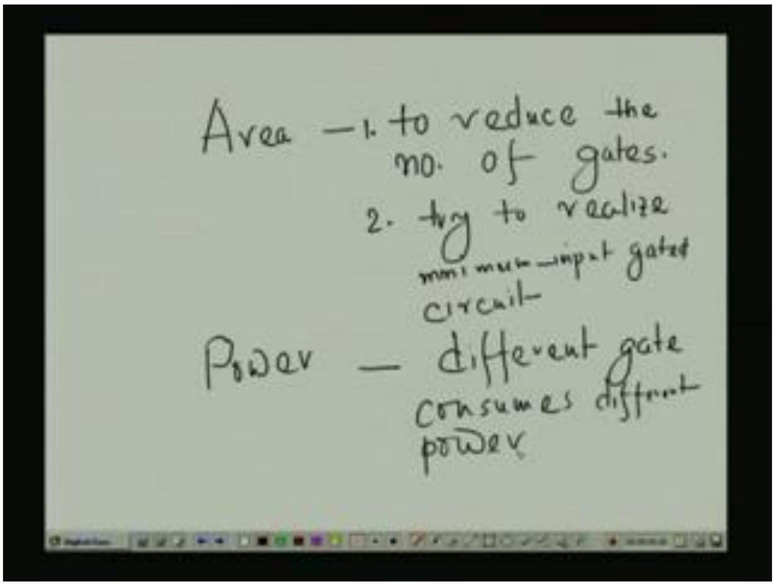
See I can take another 1, my S is a EXOR b, EXOR c, and my carry is a b plus b c plus c a, so simple that if it is a two input AND gate, only say I am thinking that, my un complemented variables are not available. I want to realize a circuit, where only the un complemented variables are available, complemented variables are not available. So, first I am realizing carry, so a b, then b c and c a, so these are then this is OR, this is my carry C.

Now, say 1 EXOR gate, I have given, so this carry is a one input, again a another input and C, I can take b is another input, so this is my S. So, this is also another kind of realization, where AND gates are needed and see two input AND gate. This is two input AND gate, number of two inputs AND gate needed is 3, three input OR 1 and three input EXOR, that is also 1.

And always, that the traditional things are there that means, 1 three input, three input OR gate, three input OR gate, that we can realize by 2, two input OR gate. Similarly, the three input EXOR; that means, I am writing OR or EXOR that can be realize 2, two input OR or EXOR. So, in real life actually there are several type of realization possible and they consists of different type of gates.

Now, another thing I want to mention, that what type of gates I want to select, actually that depends on the different type of parameters, that area another is called that power. That means, now mainly our aim, that circuit, we are we actually discussing, they are utilizing for preparing a chip or designing a chip.

(Refer Slide Time: 52:40)



Area — 1. to reduce the no. of gates.
2. try to realize minimum-input gated circuit

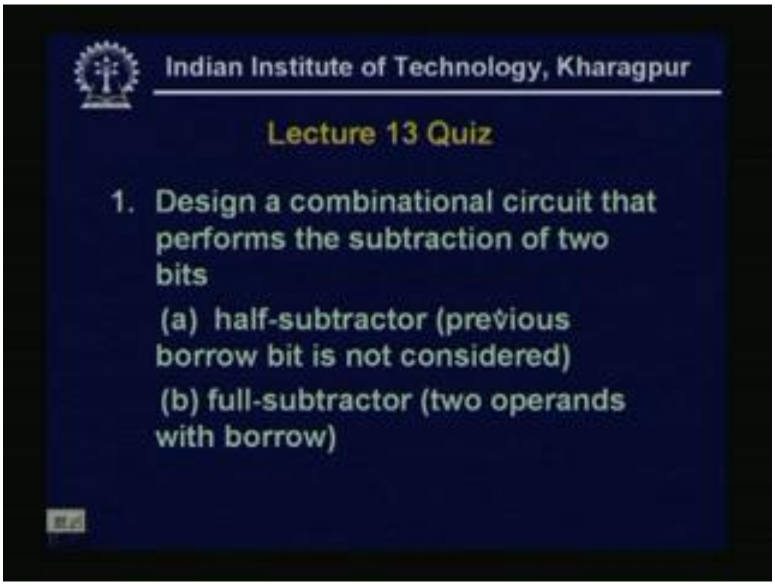
Power — different gate consumes different power

So, mainly the parameters that we have to consider, that one is the area means always we try to reduce the number of gates as well as another point. We can tell that, always we try to realize minimum input gates or gated circuit, means that, if the four input EXOR is needed, then always that. We will use that 3, two inputs EXOR and another is the power.

So, for different type of or different gates consumes different power, so if we see that, my EXOR is consuming more power, then AND OR circuit. Then, depending on that, what type of application the circuit, we are designing and then we will select that type of gates. So, mainly these are the criteria for selecting the gates for different type of circuit realization.

So, if we want to summarize, mainly by using this 2 type of adders, what we have shown that for each type of adder, half adder and full adder a number of different type of realization is possible. And mainly depending on the application; that means, whether we want to reduce area or we want to reduce the power, that we will select that type of realization, consisting of AND, OR gates or EXOR gates or the NOR gates etcetera. Now, today's quiz question is again this is a combination circuit design and very similar type, that we have discussed the adder.

(Refer Slide Time: 55:34)



Indian Institute of Technology, Kharagpur

Lecture 13 Quiz

1. Design a combinational circuit that performs the subtraction of two bits
   (a) half-subtractor (previous borrow bit is not considered)
   (b) full-subtractor (two operands with borrow)

And now the problem is design a combination circuit, that performs the subtraction of two bits, mainly the addition, we have discussed and the subtraction for similar to the half adder and full adder. First we have to do the half subtractor and then it is full subtractor, so first we have to define, what do we mean by half subtractor, half subtractor means that again this will be a single bit minus, where that previous borrow bit, the where the in for addition, it was the carry bit.

And now it is a borrow bit, that is not considered and for full subtractor the two operands as well as the borrow bit from the previous subtraction should be considered. So, these are the quiz question for this current lecture.

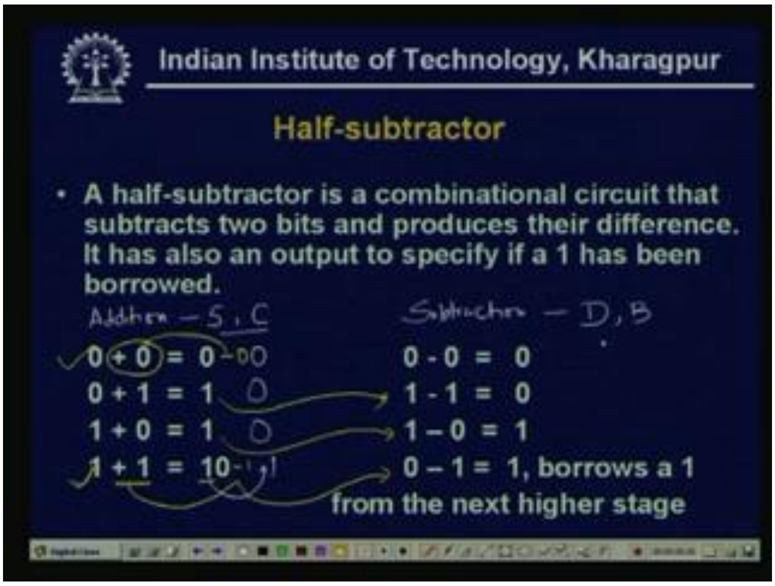
Thank you.

Digital System Design
Prof. D.Roychoudhury
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture -14
Design of Subtractor Circuits

In the last class, we have read how to design the different of adders, mainly the half adders the full adders and the different type realization using a variety of gates. Today, we will see how we can design the subtractor circuit.

(Refer Slide Time: 57:14)



Indian Institute of Technology, Kharagpur

Half-subtractor

- A half-subtractor is a combinational circuit that subtracts two bits and produces their difference. It has also an output to specify if a 1 has been borrowed.

Adder – S, C   Subtractor – D, B

0 + 0 = 0 - 00   0 - 0 = 0
0 + 1 = 1 0   1 - 1 = 0
1 + 0 = 1 0   1 – 0 = 1
1 + 1 = 10 - 1, 1   0 – 1 = 1, borrows a 1 from the next higher stage

Now, a half subtractor is a combinational circuit, that subtracts two bits and produces their difference, it has also an output to specify, if a 1 has been borrowed. So, the similarity with the adder is that, in case of addition, for addition it was the carry signal, for addition 1 sum and 1 carry has been generated. For subtraction, that sum is replaced by the difference, because it computes a difference and the carry here, the carry concept is actually they borrow.

So, we denote this as a B and the difference is D, now how the operator operates on the two operands, so again first we see the addition operation. See here, that 0 plus 0 equal to 0 and the carry is 0, 0 plus 1 is 1, carry is 0, 1 plus 0 is 1 carry 0, 1 plus 1 is actually 1 0, so sum is 0 and carry is 1. Now, see if I want to change this operation addition as subtraction, then what we can do, this 0 plus 0 equal to 0, so as if one operand from here, see this 0, I can.