Digital Electronic Circuits

Prof. Goutam Saha
Department of E & EC Engineering
Indian Institute of Technology, Kharagpur

Lecture – 60
PAL, PLA, CPLD, FPGA

Hello everybody. We are in the last class of this particular course and in this week we are discussing memory. So, in the last class we had discussed a PROM: Programmable Read Only Memory.

(Refer Slide Time: 00:34)

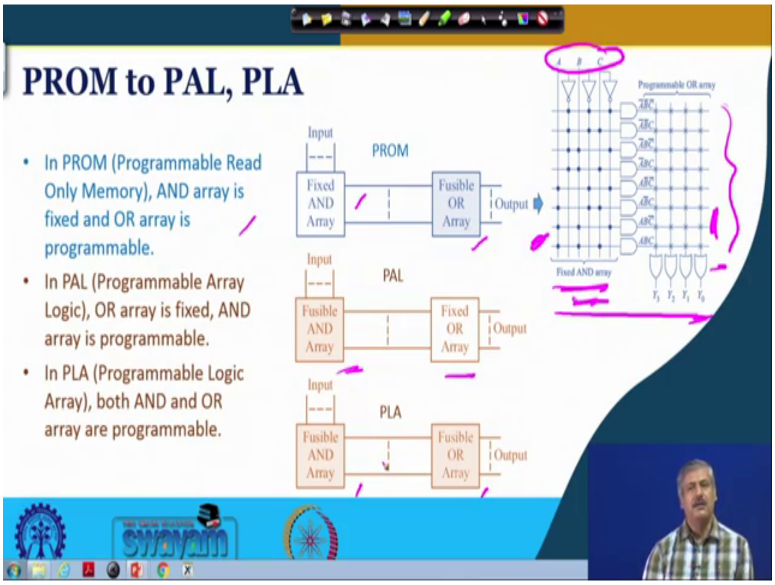
CONCEPTS COVERED

Concepts Covered:

- ❑ Programmable Array Logic (PAL)
- ❑ Programmable Logic Array (PLA)
- ❑ Realization of Multiple Outputs by PAL and PLA
- ❑ Complex Programmable Logic Device (CPLD)
- ❑ Field Programmable Gate Array (FPGA)

So, in that we had seen that the AND plane was fixed, so it actually forms the decoder and the OR plane was programmable. So, today we shall look at some other variety of this programmable logic family. So, that is Programmable Array Logic PAL, Programmable Logic Array PLA, we shall see how multiple outputs can be realized using PAL and PLA using small examples. And then we shall have a quick overview of CPLD and FPGA: Complex Programmable Logic Device and Field Programmable Gate Array.

(Refer Slide Time: 01:15)



PROM to PAL, PLA

- In PROM (Programmable Read Only Memory), AND array is fixed and OR array is programmable.
- In PAL (Programmable Array Logic), OR array is fixed, AND array is programmable.
- In PLA (Programmable Logic Array), both AND and OR array are programmable.

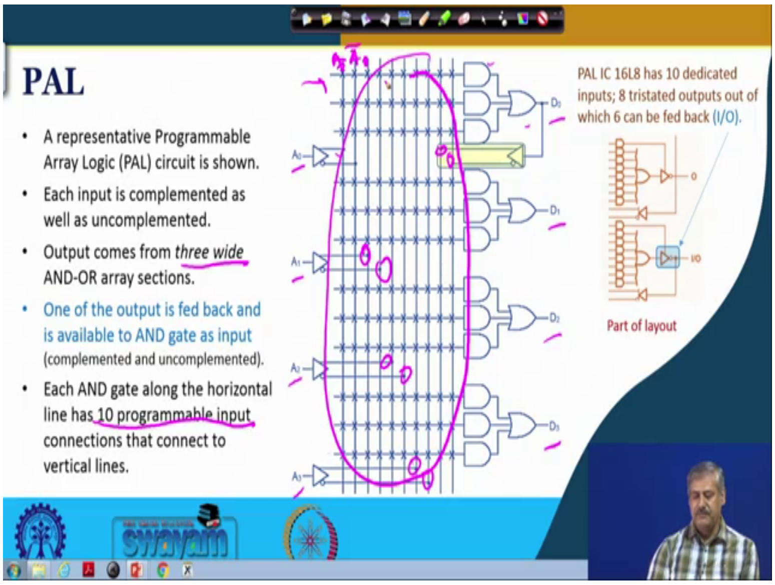
So, PROM we have already discussed in the previous class. So, in the PROM we had seen a structure like this, so this is the 3 input and which is fed to 8 AND gates. So, it is a decoder actually working generating all the mean terms right and in this OR plane right there are fusible links and depending on the connection we want the interconnection pattern that we want; so this can be programmed.

Some of the links can stay and some of the links can be burnt out and we had seen other varieties also EPROM EEPROM and so on and so forth ok, but this side was fixed and this side was programmable right. So, that was what was PROM, Programmable Read Only Memory ok, so fixed AND array and fusible OR array right. So, in PAL Programmable Array Logic this OR array is fixed, this side certain connections of the or these inputs coming from different and gates product terms that remains fixed, what you can do is program the AND plane, program the AND plane look at the point.

So, by that you can generate different kind of product terms, which gets summed up in a SOP realization of different kind of output ok. So, you can generate different functions and if we used is a use these as a memory ok. So, each of these output can be a particular memory bit. So, that part we have seen the relationship between memory and function output memory value and function output in the previous class ok. And in PLA ok, so that is Programmable Logic Array so both AND array and OR array are programmable. So, this side as well as this side this plane both the plane we can program right.

So, you can generate product term of your choice by appropriate programming of AND plane and then you can also suitably connect them by appropriate programming of OR plane ok. So, this is the PLA right Programmable Logic Array, so these are different names given by the manufacturer and that is carried forward in the textbooks ok.

(Refer Slide Time: 04:01)



PAL

- A representative Programmable Array Logic (PAL) circuit is shown.
- Each input is complemented as well as uncomplemented.
- Output comes from *three wide* AND-OR array sections.
- One of the output is fed back and is available to AND gate as input (complemented and uncomplemented).
- Each AND gate along the horizontal line has 10 programmable input connections that connect to vertical lines.

PAL IC 16L8 has 10 dedicated inputs; 8 tristated outputs out of which 6 can be fed back (I/O).

Part of layout

So, we look at one a typical PAL structure you know, so a small example actual PAL circuits will be more complex. So, in this what you can see there are 4 inputs we have named it A naught, A 1, A 2, and A3 right. And, there are 4 outputs D naught D 1, D 2, D 3 and what you can see except for this one where there is a feedback there is no such feedback for D 1, D 2, D 3. And, this OR I mean this side these OR this each one of them are OR gate has three input from three different AND gate ok. So, that is why it is called three wide AND OR array sections we have 4 such for 4 output is it ok.

So, maximum three product terms we can get for each of this sections right, so that is fixed OR plane is fixed, but which product term that we will go right that can be programmed and for which this 4 inputs you can see the input itself that is A line and this complement of these input that is A naught line A naught bar line both are available.

Similarly A 1 line is this one you can see this line is A 1 the vertical line and this line is A 1 bar similarly A 2 and A 2 bar, A 3 and A 3 bar are there ok. In addition because of the possibility of using feedback from one of the output, so this D naught over here and D naught bar that is also available in this configuration ok.

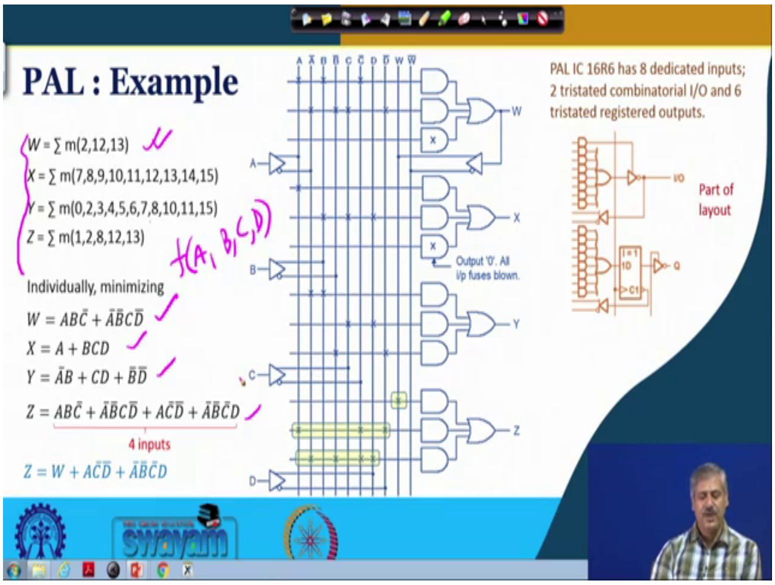
So, each of this AND gate here has got ten programmable input connection ok. So, when you are doing programming, so your are considering which of these connections are to be written and which of the connections are to be removed, is it clear ok. So, this plane is only what you are able to program is it fine. Now, what in addition we would like to discuss here is that consider one practical PAL IC 1618 in comparison to what we have shown here right as we said practical ICs are more complex in nature.

So, it has got 10 dedicated inputs 8 tri-stated outputs right, tri-stated means the output can generate logic low logic high and high implements. So, the depending on some output enable kind of input that this side is that right and out of this 8 tri-stated output 6 of them, the way we have seen over here can be feedback ok.

So, here one you can see in a part of the layout of that particular IC that this is there ok, so this is the output there is no such feedback this is whatever you see here is a part of input another input ok, so you ignore that. And what you can see over here this is the output which can be feedback as has one of the input like what you had seen in this particular column ok, so this is connected to this one.

Now, other than that what you can see? It is mentioned this is as I stroke O ok. So, this can also be used as a input I mean input for this particular you know AND plane programming ok. So, this is the meaning of it what you can see from here is it clear. So, depending on our requirement you can use it as I or O.

(Refer Slide Time: 08:44)



PAL : Example

$W = \sum m(2,12,13)$ ✓

$X = \sum m(7,8,9,10,11,12,13,14,15)$

$Y = \sum m(0,2,3,4,5,6,7,8,10,11,15)$

$Z = \sum m(1,2,8,12,13)$

$f(A,B,C,D)$

Individually, minimizing

$W = AB\bar{C} + \bar{A}\bar{B}C\bar{D}$ ✓

$X = A + BCD$ ✓

$Y = \bar{A}B + CD + \bar{B}\bar{D}$ ✓

$Z = \underbrace{AB\bar{C} + \bar{A}\bar{B}C\bar{D}}_{} + A\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D$ ✓

4 inputs

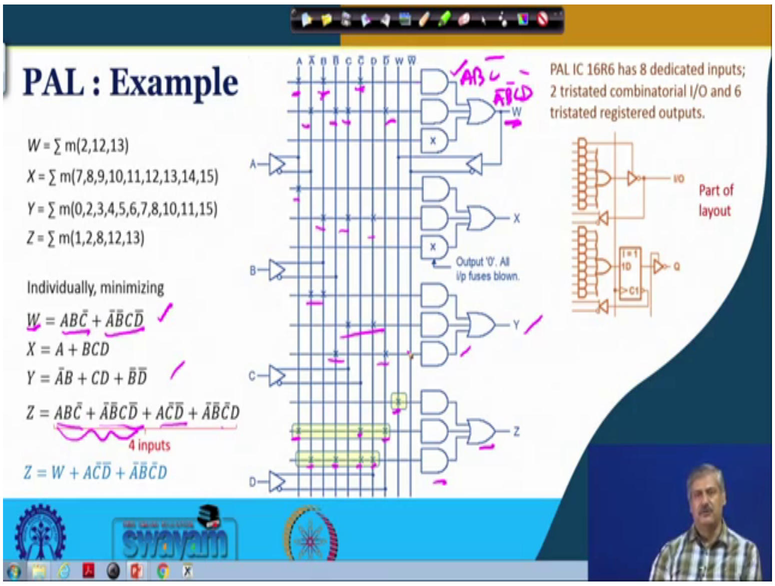
$Z = W + A\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D$

PAL IC 16R6 has 8 dedicated inputs; 2 tristated combinatorial I/O and 6 tristated registered outputs.

Now, let us look at realization of multiple output function using PAL, we take one example here right. So, in this example what you have consider there are four functions W, X, Y and Z and these are the corresponding min terms that are getting added in a SOP realization ok. So, W has got 2, 12, 13, X has got 7, 8, 9, 10, 11, 12, 13, 14, 15, Y has 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 15 and Z has got m this min terms 1, 2, 8, 12, 13 ok.

So, realize it what we shall do first each of this functions we shall individually minimize, for which we can use for very well Karnaugh map, so there are 4 variables we can name them each of this W, X, Y, Z we can consider as a function of four variables we can name them as A, B, C, D ok. So, with that four variable map Karnaugh map we can have a minimized presentation each of them individually minimized and if you go for the minimization which we have done many times before I am not shown that minimization here, but you can work it out yourself and you can see that these are the four representation the four realizations that we will get clear ok.

(Refer Slide Time: 10:25)



PAL : Example

$W = \sum m(2,12,13)$

$X = \sum m(7,8,9,10,11,12,13,14,15)$

$Y = \sum m(0,2,3,4,5,6,7,8,10,11,15)$

$Z = \sum m(1,2,8,12,13)$

Individually, minimizing

$W = AB\bar{C} + \bar{A}\bar{B}C\bar{D}$ ✓

$X = A + BCD$

$Y = \bar{A}B + CD + \bar{B}\bar{D}$ ✓

$Z = \underbrace{AB\bar{C} + \bar{A}\bar{B}C\bar{D}}_{\text{4 inputs}} + A\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D$

$Z = W + A\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D$

PAL IC 16R6 has 8 dedicated inputs; 2 tristated combinatorial I/O and 6 tristated registered outputs.

Part of layout

Now, if we have been given this you know PAL to realize it we see that for realizing W we need to generate two product term of four variables ok. So, four variables are fine that A naught to A 3 we name them as ABCD right, and corresponding this terms will be A A bar B B bar over here C C bar D D bar right this is there right and two product term we can generate like A, B, C prime. So, A is there connection we written B and C bar.

So, these three, so this is a three input and gate, so this is the way also we can you know show a connection which we studied discussed before studied before ok. So, though there is it shows one input actually there are two inputs, so basically it is A, B and C bar that is being get generated by this and gate ok, so this is A BC bar. What about this and gate, this and gate is generating this particular product term A bar, B bar C this is C line you can see and D bar right.

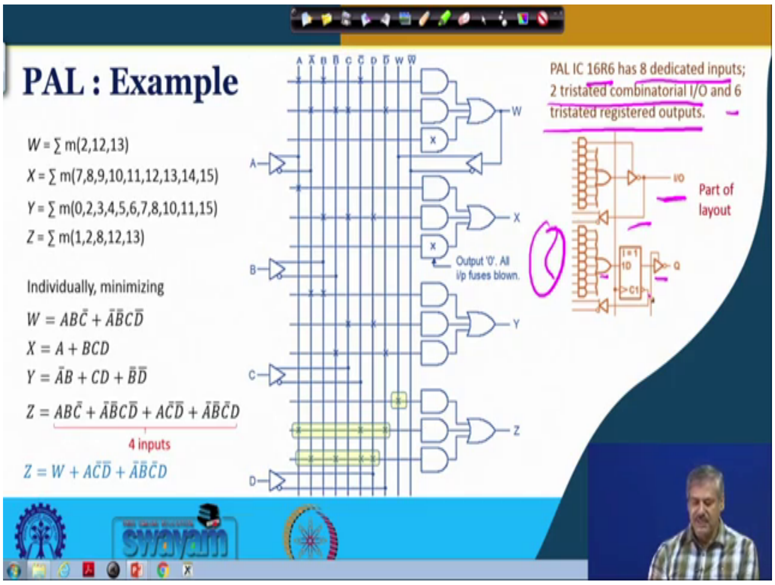
So, this is the second one is generating A bar, B bar, C and D bar and the third and gate we do not need so we should not show any connection, so it is always 0 output is always 0 because no connection is there right. So, this is the way we can generate W. Similarly X requires two such terms one is A another is B, C, D, so B, C, D. So, this is X third and gate is not used right Y requires three product terms 1 2 3 right and accordingly we use the AND plane programming and generate this individual product term and Y is obtain.

Now coming to realization of Z we see that there are four product terms 1, 2, 3, 4 and you have got only three and gates right, so it is not possible ok. We need another and gate, but if you look closely you can see that out of this four product terms that are there the first one first two ABC prime and A prime, B prime C D prime is already getting generated as W this is nothing, but W right and we have an option of using this as a feedback right.

So, if we connect, if we use this W and W bar these two lines and take connection from W, so that itself will generate two of this to the product term third product term ACD prime I can generate from A, this is C prime and this is D prime ok. And the last one A prime, B prime, C prime D. So, this is A prime, B prime, C prime and D ok. Then these three terms I connect together OR together and I can get these a ok.

So, in PAL based realization we can see the reusability of the W that is W means A function output that is getting generated and options that are available. See in this case in this example only one such feedback was available, but I had shown in the previous slide that in practical IC there may be more than one such feedback possible and accordingly we shall make use of such you know facilities that the IC provides.

(Refer Slide Time: 14:27)



PAL : Example

$W = \sum m(2,12,13)$

$X = \sum m(7,8,9,10,11,12,13,14,15)$

$Y = \sum m(0,2,3,4,5,6,7,8,10,11,15)$

$Z = \sum m(1,2,8,12,13)$

Individually, minimizing

$W = AB\bar{C} + \bar{A}\bar{B}C\bar{D}$

$X = A + BCD$

$Y = \bar{A}B + CD + \bar{B}\bar{D}$

$Z = \underbrace{AB\bar{C} + \bar{A}\bar{B}C\bar{D} + A\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D}_{\text{4 inputs}}$

$Z = W + A\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D$

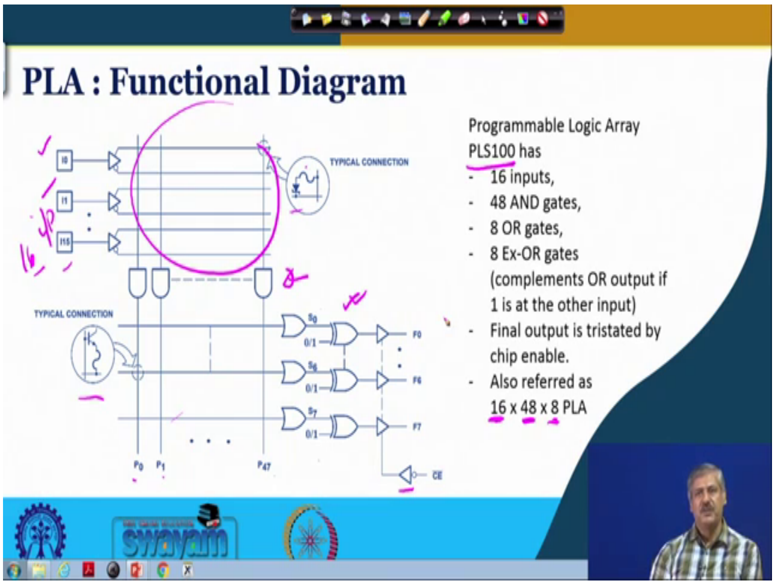
PAL IC 16R6 has 8 dedicated inputs; 2 tristated combinatorial I/O and 6 tristated registered outputs.

Part of layout

So, another practical IC this is PAL IC 16R6, it has got 8 dedicated inputs, 2 tristated combinatorial I O; I O means it is both input and output it can be use, so, there is a feedback there. So, this is the kind of thing which we had seen before alright and 6 tristated registered outputs ok. So, how does it look like? So, this is the output ok, so this is the AND gate over here this AND plane which is programmable and getting connected to OR gate. So, these OR gate goes output goes to a flip flop ok.

So, bank of such flip flop, so you are saying that it is registered and then this is tristated and then the output is available. So, this is the way also PAL output can be generated. And PAL output with such options are also considered in a another you know larger family called programmable logic device ok. So, that is PAL based field with this kind of you know flip flops involved which we shall see later.

(Refer Slide Time: 15:45)



PLA : Functional Diagram

Programmable Logic Array PLS100 has

- 16 inputs,
- 48 AND gates,
- 8 OR gates,
- 8 Ex-OR gates (complements OR output if 1 is at the other input)
- Final output is tristated by chip enable.
- Also referred as 16 x 48 x 8 PLA

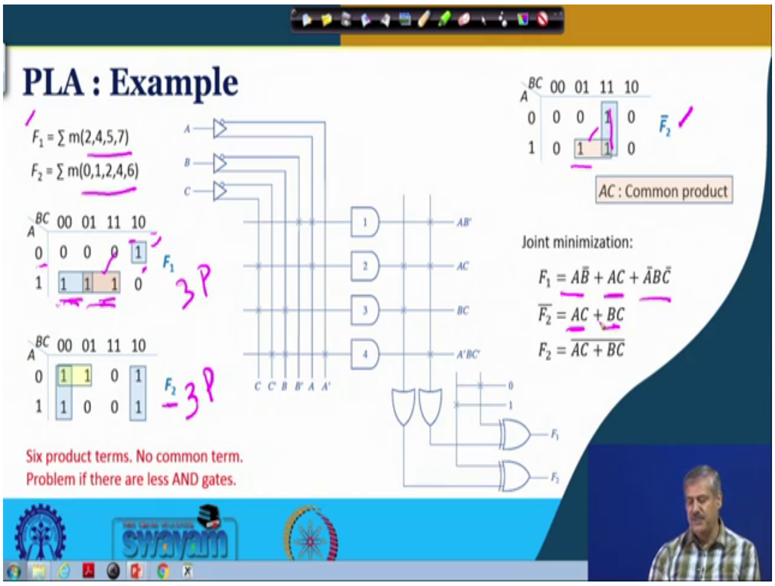
Now, comes PLA, so PLA is Programmable Logic Array right. So, in this both AND plane and OR plane are programmable ok. So, one such you know arrangement you can see over here ok. So, this is I naught, I 1, I then I 2, I 3 etcetera to I 15, so that is 16 inputs are there right and there are certain number of AND gates.

So, this AND gates this plane is programmable, so these are the interconnections you can see this is generate diode and there is a fuse, so depend initially all of them are there. So, depending on the kind of product term you want to generate ok. So, you will return the fuse or you will burn the fuse right. And this is the OR plane, so in the number of product term that is getting generated in this particular example is P naught, P 1 to P 47, so 48 product terms are getting generated ok.

So, this is a practical IC we have taken PLS 100 ok, so this are the possible number of product terms. So, this product terms are going to OR gates ok, how many OR gates are there? S naught to S 7; so, 8 outputs are there, so that is why it is also called 16 48 8 ok. Remember, with 16 inputs if you want to generate all the min terms, so number of min terms would be 2 to the power 16 ok. So, we are not talking about that we are talking about generating those product terms which are required for this realization right. And in this particular IC so this OR plane the switch is through a transistor and then a fuse and the fuse is returned or removed ok.

In addition what you can find over here that the OR gate output is pass through an Ex OR gate where if a 0 is present at its input. So, the output will be just S 0 only right and if 1 is presented Ex-OR you know this relationship we know, so the output here will be S naught bar is it fine. So, the output that is getting generated from the OR gate can go uncomplemented as well as complemented ok. So, this is possible in a PLA realization P 1 that we see over here and finally, the tri state and output that is there as an option.

(Refer Slide Time: 18:38)



PLA : Example

$F_1 = \sum m(2,4,5,7)$

$F_2 = \sum m(0,1,2,4,6)$

A \ BC	00	01	11	10
0	0	0	0	1
1	1	1	1	0

F_1 3P

A \ BC	00	01	11	10
0	1	1	0	1
1	1	0	0	1

F_2 3P

Six product terms. No common term.
Problem if there are less AND gates.

A \ BC	00	01	11	10
0	0	0	1	0
1	0	1	1	0

$\overline{F_2}$

AC : Common product

Joint minimization:

$F_1 = A\bar{B} + AC + \bar{A}B\bar{C}$

$\overline{F_2} = AC + BC$

$F_2 = \overline{AC + BC}$

Now, if you want to realize a multiple logic function using PLA what will be the approach ok? So, we take example then it will be useful right. So, here is an example simple example for the sake of understanding. So, one function F 1 is getting generated is to be generated which has got min terms 2, 4, 5, 7 another one 0, 1, 2, 4, 6 ok.

So, F 1 if you put it in the form of a Karnaugh map it looks something like this. So, how many you know if you minimize it, how many such product terms will be there three product terms will be there, so this is 1, this is 1 and this is 1 right and if you write it then it will be this one is AB bar, first one is this one is AB bar, then this one is AC and the third one which cannot be included any group of the one member. So, that is A prime BC prime ok, so this is the third one right.

Now, for F 2 if you just put them in the Karnaugh map, so these are the 1's five 1's that are there 0, 1, 2, 4 and 6 and if you group them three terms are coming now these three terms you will see that we will require ok. So, there is no overlap between them, so here

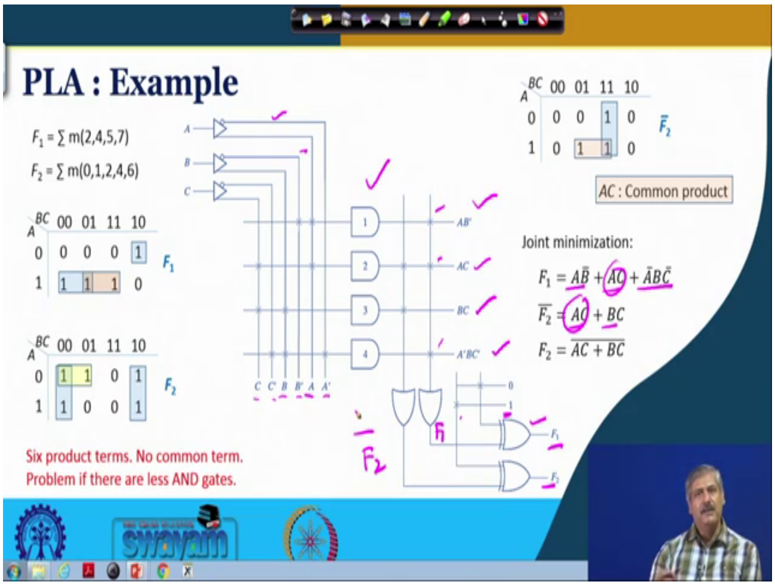
2 product term to be need to be generated and here 3 product term need to be generated. So, six product term need to generated and then they need to be hold right.

Now, consider that if the number of the PLA size is given is such that you do not have as many number of AND gates to generate 6 product term. Do you get the point right? Otherwise picking can we generate less number of you know product term and still realize these two functions this is what the problem statement is and for that since we know that the PLA output has got these Ex-OR gate which provides an option to get the output in complemented as well as uncomplemented form.

So, we look at we look at this F 2 which has got three 0's realization in a different manner ok. So, this F 2 instead of F 2 if you look at F 2 bar realization, then these are the you know three 1's that will be there wherever 0 is there 1 and wherever one 0 will be there that is the prime of you know complement of F 2.

So, now if you see there are two product terms that need to be generated and one product term in this color over here you see they are common which is AC and another product term is there which is this one which is BC; is it clear?

(Refer Slide Time: 22:03)



PLA : Example

$F_1 = \sum m(2,4,5,7)$

$F_2 = \sum m(0,1,2,4,6)$

A \ BC	00	01	11	10
0	0	0	0	1
1	1	1	1	0

F_1

A \ BC	00	01	11	10
0	1	1	0	1
1	1	0	0	1

F_2

Six product terms. No common term.
Problem if there are less AND gates.

A \ BC	00	01	11	10
0	0	0	1	0
1	0	1	1	0

$\overline{F_2}$

AC : Common product

Joint minimization:

$F_1 = A\bar{B} + AC + \bar{A}B\bar{C}$

$\overline{F_2} = AC + BC$

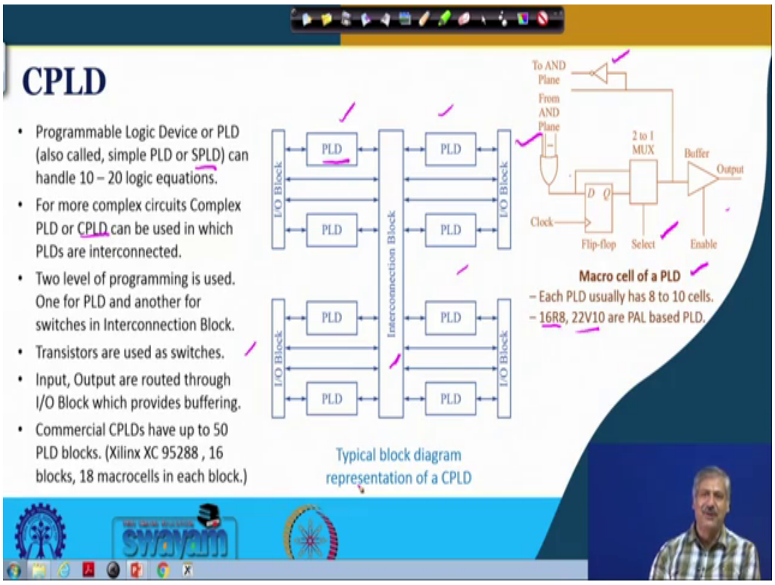
$F_2 = \overline{AC + BC}$

So, F 2 prime has got AC and BC and F 1 has got AB prime AC and A prime B prime C prime and this AC is common. So, how many product term then we need to generate 1, 2, 3, 4 to realize F 1 and F 2 prime is it fine. So, if you do that right then a PLA with having only 4 AND gates would do we do not need to go up to something which is of higher size that is requiring six ok. So, you can realize this is A, B, C and A prime, so this is A and this is A prime ok. So, these are the lines that are there A A prime B B prime C C prime are generated. So, you can generate A and B prime put together and so this is a two input AND gate, so that is the you know connections that you are retaining there.

So, it is generating AB prime AC BC and finally, this is A prime, B prime C prime right and then if you or them this way. So, this will generate a b prime AC and AB, so, this two three. So, this will generate F 1 and this will generate F 2 prime ok. And now we have got these EX OR gate available, so in case of F 1 you just considered 0, so it will go as F 1 and in this case of F 2 prime you put the other input as one then it will go as F 2 ok.

So, you can use this thing to reduce the number of product terms that will be required because in the earlier example this is a simple example. So, there will be where 48 AND gates, but number of inputs where also 16 and possible combination of the 10 input is 2 to the power 16 right. So, just to see that required number of product terms are there I mean minimum number of product terms, then we can look at these aspect of PLA where option for this complimented and uncomplemented outputs both are available. So, in PLA we are looking at possibility of joint minimization in PAL individual minimization was there because no shear term and all shear product term ok.

(Refer Slide Time: 24:29)



CPLD

- Programmable Logic Device or PLD (also called, simple PLD or SPLD) can handle 10 – 20 logic equations.
- For more complex circuits Complex PLD or CPLD can be used in which PLDs are interconnected.
- Two level of programming is used. One for PLD and another for switches in Interconnection Block.
- Transistors are used as switches.
- Input, Output are routed through I/O Block which provides buffering.
- Commercial CPLDs have up to 50 PLD blocks. (Xilinx XC 95288 , 16 blocks, 18 macrocells in each block.)

Typical block diagram representation of a CPLD

Macro cell of a PLD

- Each PLD usually has 8 to 10 cells.
- 16R8, 22V10 are PAL based PLD.

So, with this we go to what is called complex programmable logic device, so before that we just take a look at what we consider a macro cell of a programmable logic device ok. So, the way we had seen it before; the way we had seen before that there is a AND plane ok. So, from the AND plane this OR gate is taking the product terms alright this product term is going to a flip flop right. So, from this flip flop ok, so this flip flop we can see that we can select it to go to the next level ok.

So, either this value or the Q value right and this then it can go back to the AND plane in a this manner or the other manner right, the complemented value or the other value and then there is a buffer. So, this is essentially forms a macro cell right is it ok. So, you can bypass the flip flop or you can take the output from the flip flop, so that is what this select line gives you gives you with the option is it alright ok.

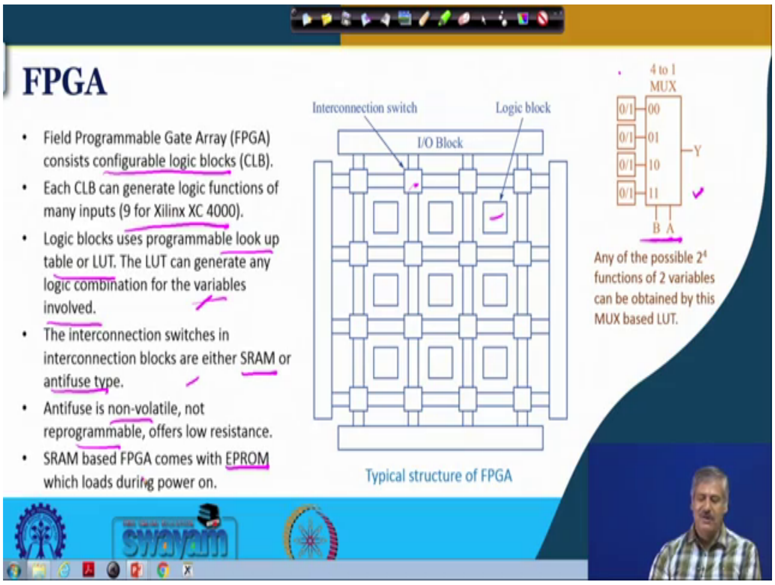
So, this is basically one macro cell and a programmable logic device or PLD will be having about 8 to 10 such cells typically it will have like this right and examples are the 16R8, 20 V 22V10 these are all PAL based PLD ok. So, this type of PLD programmable logic device is also called simple PLD or SPLD which has got a set of such macro cell alright.

And this can handle such PLD can handle about 10 to 20 logic equation alright because number of cells are of that order. Now, for more complex circuits what we use is called pro complex programmable logic device which we can use and it is also abbreviated as CPLD ok, so this is CPLD ok, where multiple PLDs are interconnected right.

So, for which this CPLD requires two level of programming one is for this PLD programming another is for these interconnection the switches are there they need to be program, so that appropriate interconnections are made. And in this transistors are used as switches and inputs outputs are routed through I O block which offers buffering and commercial CPLDS can have up to 50 PLD blocks right.

For examples Xilinx XC 952 double 8, it has got 16 blocks and in each block there are 18 macro cells ok. So, as we said we are towards the end of this course we finishing it today. So, these are for advanced level use of digital electronic circuit, but it is good to get into reduced at this level right.

(Refer Slide Time: 28:09)



FPGA

- Field Programmable Gate Array (FPGA) consists configurable logic blocks (CLB).
- Each CLB can generate logic functions of many inputs (9 for Xilinx XC 4000).
- Logic blocks uses programmable look up table or LUT. The LUT can generate any logic combination for the variables involved.
- The interconnection switches in interconnection blocks are either SRAM or antifuse type.
- Antifuse is non-volatile, not reprogrammable, offers low resistance.
- SRAM based FPGA comes with EPROM which loads during power on.

Typical structure of FPGA

Any of the possible 2^4 functions of 2 variables can be obtained by this MUX based LUT.

And when we talk about CPLD we should also talk about another such IC called family called FPGA ok. So, this field programmable gate array, so this is further you know advancement of CPLD, CPLD cannot provide much of a you know packing density and has got its limitations, so people have moved forward. So, these FPGAs consist of Configurable Logic Blocks or CLB ok. And each CLB can generate functions of many different inputs right each CLB ok.

So, this is one particular logic block we are talking about can have many inputs for example, Xilinx XC4000 can have nine such inputs ok. And this logic blocks to generate output uses programmable lookup table or LUT. The LUT can generate any logic combination for the variables involved ok. So, let us just look at an example what it means. So, here you can see there is a 4 to 1 multiplexer right and there are two variables B and A.

So, with two variables B and A we know we had seen it in the beginning how many different possible functions can be generated ok. So, one possibilities for you know four possible cases of B and A 0 0 0 0 has all output, then 0 0 0 1 has all output ok, then 0 0 1 0 0 0 1 1. So, these are the possible cases so 2 to the power 4 is not it 2 to the power 2 2 to the power 2. So, that is 16 possible functions are possible to be generated ok.

So, 1 multiplexer based you know lookup table of 2 variables. So, this is BA right can have either 0 or 1 at its input ok. So, depending on your requirement you can place 0 or 1 and for different values of this BA 4 possible combinations it will take up 0 or 1 and that particular function will be realized ok. So, this is an example of you know look up table based realization of functions. So, this is what is indicated here.

Now, the interconnections which is in the interconnections blocks the interconnections switches is, so that is u c ok. So, these are the different logic blocks. So, these can be of either SRAM or something called antifuse type ok. So, antifuse type switches are non volatile, but they cannot be reprogram and when in a particular switch it is offering low resistance in another case it offers open circuit very high resistance ok. Whereas, SRAM based you know FPGA when this interconnections ok. So, it is reprogrammable you all know, but it is volatile also ok.

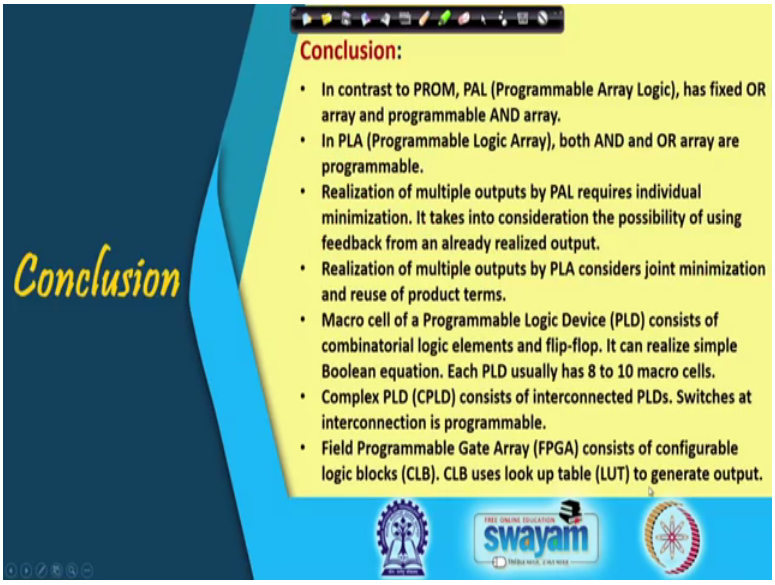
So, during powered on an EPROM will be there to load the necessary values in this SRAM switches ok. So, these are some very basic things and more when you use them in higher level course and when you learn how to program it and get more complex digital circuit implemented on PGA platform or Xilinx platform.

(Refer Slide Time: 31:54)

References:

- ❑ Donald P. Leach, Albert P. Malvino, and Goutam Saha, Digital Principles & Applications 8e, McGraw Hill
- ❑ M. Morris Mano and Michael D. Ciletti, Digital Design 5e, Pearson
- ❑ https://www.digchip.com/datasheets/parts/datasheet/364/PLS100-pdf.php
- ❑ https://www.xilinx.com/support/documentation/data_sheets/DS063.pdf

(Refer Slide Time: 31:54)



Conclusion:

- In contrast to PROM, PAL (Programmable Array Logic), has fixed OR array and programmable AND array.
- In PLA (Programmable Logic Array), both AND and OR array are programmable.
- Realization of multiple outputs by PAL requires individual minimization. It takes into consideration the possibility of using feedback from an already realized output.
- Realization of multiple outputs by PLA considers joint minimization and reuse of product terms.
- Macro cell of a Programmable Logic Device (PLD) consists of combinatorial logic elements and flip-flop. It can realize simple Boolean equation. Each PLD usually has 8 to 10 macro cells.
- Complex PLD (CPLD) consists of interconnected PLDs. Switches at interconnection is programmable.
- Field Programmable Gate Array (FPGA) consists of configurable logic blocks (CLB). CLB uses look up table (LUT) to generate output.

So, with this we conclude today's class, so what we have seen that in contrast to PROM PAL has fixed OR array and programmable AND array right and in PLA both and AND, OR array are programmable. And realization of multiple outputs by PAL requires individual minimization, it takes into consideration the possibility of using feedback from an already realized output and for PLA, we considered joint minimization and reuse of product terms ok.

Macro cell of a programmable logic device PLD consists of combinatorial logic elements and flip flop it can realize simple Boolean equations; Boolean equation and each PLD will have 8 to 10 typically macro cells ok. And complex PLD or CPLD consist of interconnected PLDs and switches of these interconnections are also programmable other than the PLD switches are also programmable FPGA; a field programmable gate array consist of configurable logic blocks and or CLB CLB uses Look Up Tables to generate output.

Thank you.