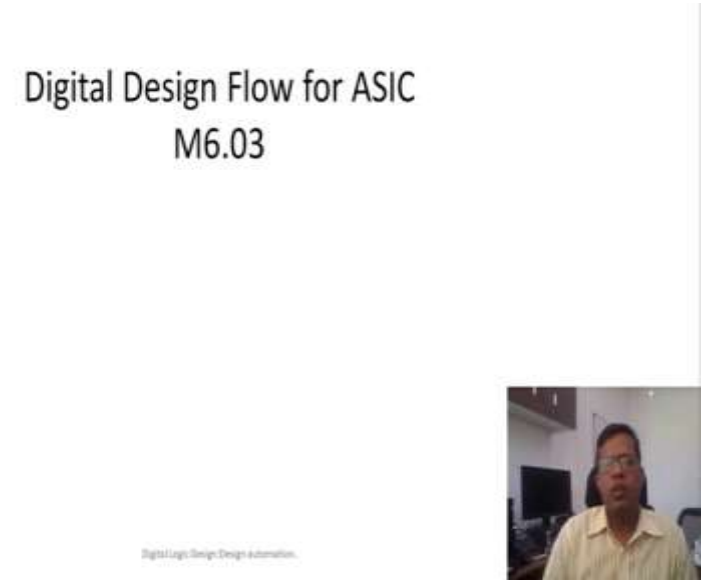
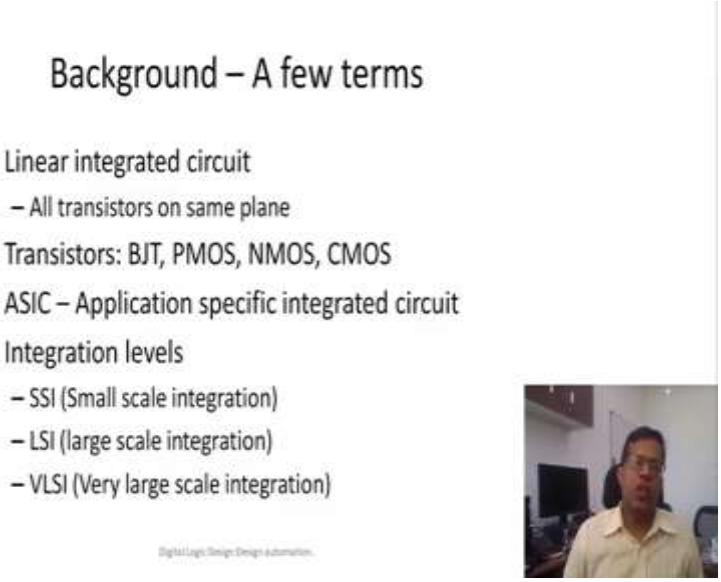
Digital System Design
Professor Neeraj Goel
Department of Computer Science Engineering
Indian Institute of Technology, Ropar
Introduction to ASIC design flow Part - 1

(Refer Slide Time: 0:17)



Hello everyone, in our previous lecture we have discussed about a design flow for FPGA, today we are going to discuss about design flow for ASIC.

(Refer Slide Time: 0:30)



So, let us first understand what is ASIC. To understand ASIC let us also go through couple of more terms, one important term is linear integrated circuit or we also call it IC integrated circuit and linear integrated circuit. So, basically integrated circuit is when number of transistors or your whole circuit is integrated onto one chip and this is specifically called linear integrated circuit because all the transistors, all the basic component of our circuit are laid out onto the same plane.

So, when you say one plane it is a 2D plane where all the transistors are placed at different positions so this is called linear integrated circuit. And what is a transistor? Transistor is a device which has usually three inputs. For example, if you see the other passive elements like registers, inductor, capacitor all of them had two inputs but transistor used to have three inputs so the earlier transistors were, so you would have heard of these names BJT bipolar junction transistor, PMOS or it is also called sometime simple MOS, PMOS, NMOS so and also CMOS, CMOS will have both PMOS as well as NMOS, it is a design method not but still it is called transistor.

So, if any transistor will have both the components PMOS as well as NMOS then it is called CMOS, CMOS is complementary MOS. So, now these are the various type of transistors which has been invented or which has been designed in last couple of years, last 40, 50 years but today most of the designs are usually focused on to the CMOS type of transistors which has both PMOS as well as NMOS, specifically in the digital design domain.

So, there is another term which is ASIC which we have said in the starting of this lecture ASIC is application specific integrated circuit. So, an integrated circuit which we are designing for a particular application, you can also say a circuit which is designed, being designed for a particular application is ASIC.

Now, by circuit here usually it means that it is a digital circuit, a digital circuit which is designed for a particular application is called an ASIC. You also see this word integrated here, integrated means how many transistors are put onto one particular chip that is the that also defines the scale of integration.

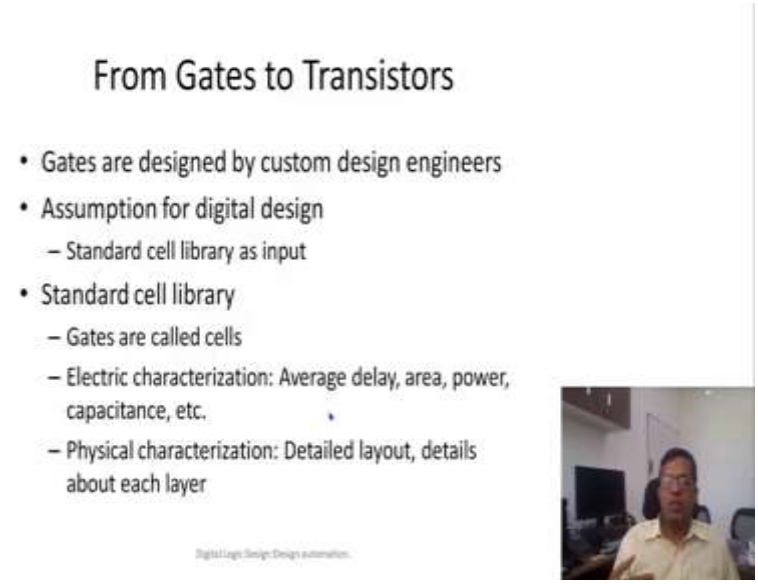
So, in small scale integration SSI circuits these number of transistors could be in the order of thousands and less than thousands and in a large scale integration when number of transistors grows beyond thousands and when the number of transistors will be in the range of hundreds of thousand then it will become VLSI very large scale integration.

So, when you say you will hear some of the related terms like VLSI design so that means a design which would have large scale integration involved, so that means we are focusing on a design which would eventually have more than thousands of transistors or hundreds of thousands of transistors.

So, when we started this course we talked about Morse law, so because of this Morse law because of this integration, increase in integration or increasing number of transistors per chip it has gone beyond hundreds of thousand limits. So, right now we have hundreds of million transistors or some of the chip also have more than a billion transistors on one integrated circuit or one chip.

So, that you can also call it very ultra large scale or very very large scale, you can coin all those words but they become meaningless because VLSI itself means it is going to be very large scale, it is the scale is beyond the limit of our hand crafted design. So, VLSI also mean that there has to be some sort of automation which should be there so that the number of transistors could be fitted in on to chip and could be designed. So, in today's lecture we will briefly go through this process of automation so that VLSI design could be feasible particularly the digital design.

(Refer Slide Time: 5:50)



So, now when we have seen in this particular course we were talking only and only about the gates not the transistors, we were talking about basic gates like AND gate, OR gate and NOT gate and we also talked about universal gates like NAND gate and NOR gate and then we have

also designed various blocks which were including which were designed using these gates and then those could be utilized to design something else.

So, for example, we have predominantly used multiplexers or flip flops and using multiplexers flip flop and adders we have designed further more sophisticated circuits. So, but we have not talked about transistors so who will design the transistor that is a one question which will come in our mind that who will design the transistor so that our digital design could be successful.

So, the answer to this question is usually it is designed by custom design engineers or analog design engineers not the digital design engineers, we assume that in our digital design we assume that these gates are available to us, somebody has designed it. So, when somebody has designed so possibly they have designed it and they have created a sort of library and this library is taken or assumed as a input.

So, what would be there in that library? See in that library there could be you can also say there could be only one gate universal NAND gate but no because of we would like to optimize our design for area sometime we would like to optimize our design for performance or power because of that various variants of each and every gate would be there.

For example, if there is a NAND gate there would be a different design for two input NAND gate, three input NAND gate, four input NAND gate. And similarly, for and OR gates also there will be multiple designs further because it is going to be designed by custom design engineers and analog design engineers, they will also think about what would be the load.

So, basically one output of the gate if it is going to multiple different gates then the it should be able to take care of the load capacitance, it should be able to take care of the fan out capacitance or fan out effect of fan out. So, to take care of the effect of fan out or design of the gate also need to be modified so that it can still be high performance.

So, sometime multiple variants of the gates would be designed based on the circumstance that if it is to be used for used in a case of high fan out then there would be internal design would be different, slightly different. So, when the standard, so this is called standard cell library, standard cells, what is standard here?

So, internally there is some physical dimensions like height of the gate, physical height of the gate is fixed that is why it is called standard cell. So, if the standard cell library is the input so basically what do we mean when we say standard cell library is the input so it means that whenever digital design would take place some of the estimation has to take place.

Yesterday we have talked about that after placement and routing we clearly know what is going to be the delay, what is going to be the effect on the latencies, etc. So, all those latencies will come because the standard cell libraries are assumed as a input and because they are assumed as an input so there is some characterization which is involved.

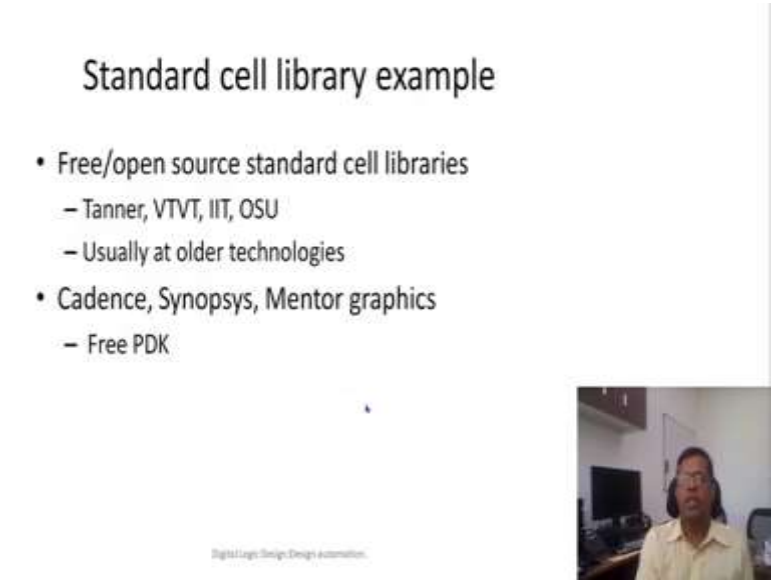
So, for example, electrical characterization which means that for a particular gate what is the average delay. Characterization will also involve it will clearly tell what is the delay when output will go from high to low, what would be the delay when output will go from low to high.

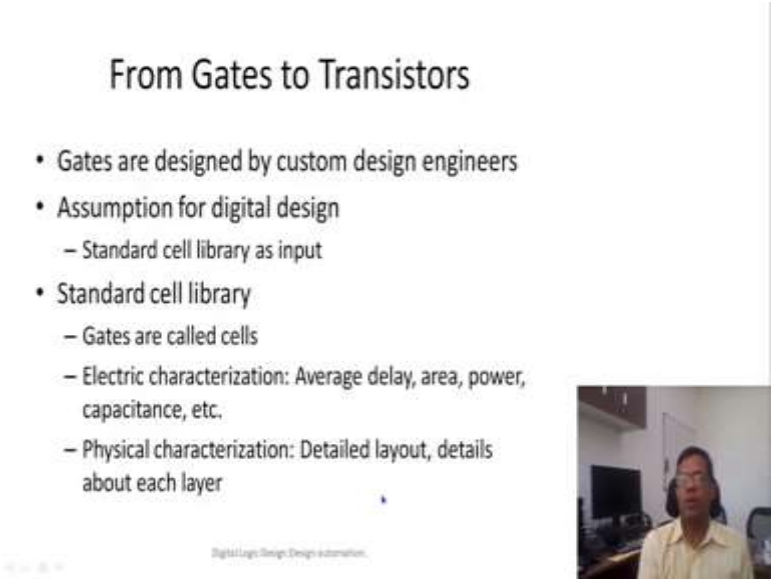
Similarly, what would be the power consumption in each and every case, what is going to be the area, what is going to be the output capacitance, tray capacitance, all those electrical details are present to us in some abstracted manner so that whenever we analyze using the standard cell library, using this particular assuming this input then we can have some estimation that what is going to be the overall latency or what is going to be the minimum or maximum latency for our design.

So, that we can calculate a clock period, so that we can say that in one clock period how much design could be fit in. So, all those information is required so that we can choose, so that we, so that we can design our final circuit. So, this is the detail which is required electrical characterization is the detail which is required at so that we can design at our gate level.

But to move forward the physical characterization, physical detail is also required that what is the layout and what are the details about each and every layer. So, what are these layers, layout, etcetera we will talk in couple of minutes.

(Refer Slide Time: 11:34)





So, also take some example of standard cell library, so yeah before that I will tell you that there is a couple of libraries standard celebrities which are present so some of them are free and open source, all these libraries also depends on that which technology we are working in so you these technologies refer to the fabrication technology, so you might have heard that 20 nanometer, 45 nanometer or 90 nanometer so these are all fabrication technologies, lower technology means more transistors could fit in.

And it also means that a lower technology node, basically the latest technology node, let us say 22 nanometer or 14 nanometer, it would have lesser delay, it would also consume lesser power

so and it would also be able to integrate more number of transistors per mm square or per area, per unit area.

So, earlier like 20 years back there were many free and open source technology libraries like tanner, VTVT, or Illinois, Institute of Technology have published their own open source library, OSU is another university which has published their own open source libraries so the very various open source standard cell libraries that has been published over internet and we can access them and the access is required so that we can have a complete design flow for digital design.

But the challenge with most of these technologies, most of these open source libraries were that they are usually designed in for older technologies like 180 nanometer or 350 nanometer so basically they were at the micron level. But today we have moved on to the latest technology which is 14 nanometer, 22 nanometer or 45 nanometers.

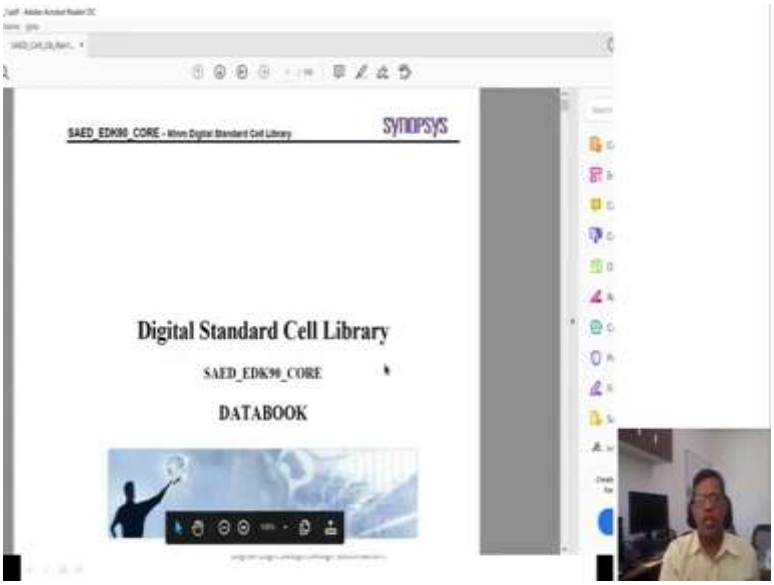
So, for the latest technologies we can still use the libraries which has been published by commercial companies like Cadence, Synopsys and Mentor graphics. So, these are the three major companies in electronics design automation field so which provide most of the software which is required for design automation.

So, what they, they do not give open source but they give a free pdk, pdk is process development kit so this is still tied to a particular development node like 45 nanometer or 22 nanometer so or 90 nanometer and then it is free in the sense that they provide all the characterization information. For example, all these information so that we can use them in our design and we can design our digital design or ASIC based on these libraries.

(Refer Slide Time: 14:30)

So, to give you some more detail I will share one of the document which shares the detail of this library.

(Refer Slide Time: 14:38)



Parameter	Slope Values (ns)		
Data Input Slope	$0.5T_{sr}$	$1T_{sr}$	$5T_{sr}$
Clock Input Slope	$0.5T_{sr}$	$1T_{sr}$	$5T_{sr}$

6. Digital Standard Library Cells List

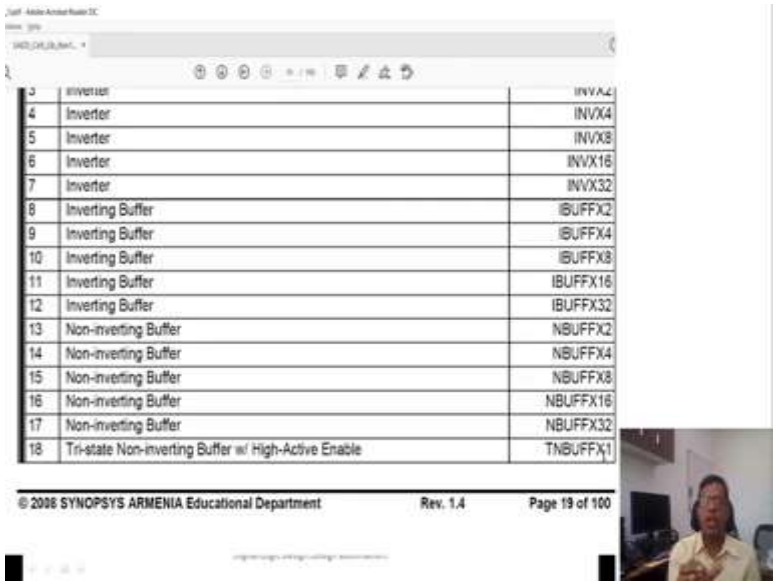
SAED_EDK90_CORE Digital Standard Cell Library contains 249 cells in total, the list of which is shown in Table 6.1.

Table 6.1. Digital Standard Library Cells List

No	Cell Description	Cell Name
	Inverters, Buffers	
1	Inverter	INVX0
2	Inverter	INVX1
3	Inverter	INVX2
4	Inverter	INVX4
5	Inverter	INVX8
6	Inverter	INVX16
7	Inverter	INVX32
8	Inverting Buffer	IBUFFX2
9	Inverting Buffer	IBUFFX4
10	Inverting Buffer	IBUFFX8
11	Inverting Buffer	IBUFFX16
12	Inverting Buffer	IBUFFX32
13	Non-inverting Buffer	NBUFFX2
14	Non-inverting Buffer	NBUFFX4
15	Non-inverting Buffer	NBUFFX8
16	Non-inverting Buffer	NBUFFX16
17	Non-inverting Buffer	NBUFFX32

Table 6.1. Digital Standard Library Cells List

No	Cell Description	Cell Name
	Inverters, Buffers	
1	Inverter	INVX0
2	Inverter	INVX1
3	Inverter	INVX2
4	Inverter	INVX4
5	Inverter	INVX8
6	Inverter	INVX16
7	Inverter	INVX32
8	Inverting Buffer	IBUFFX2
9	Inverting Buffer	IBUFFX4
10	Inverting Buffer	IBUFFX8
11	Inverting Buffer	IBUFFX16
12	Inverting Buffer	IBUFFX32
13	Non-inverting Buffer	NBUFFX2
14	Non-inverting Buffer	NBUFFX4

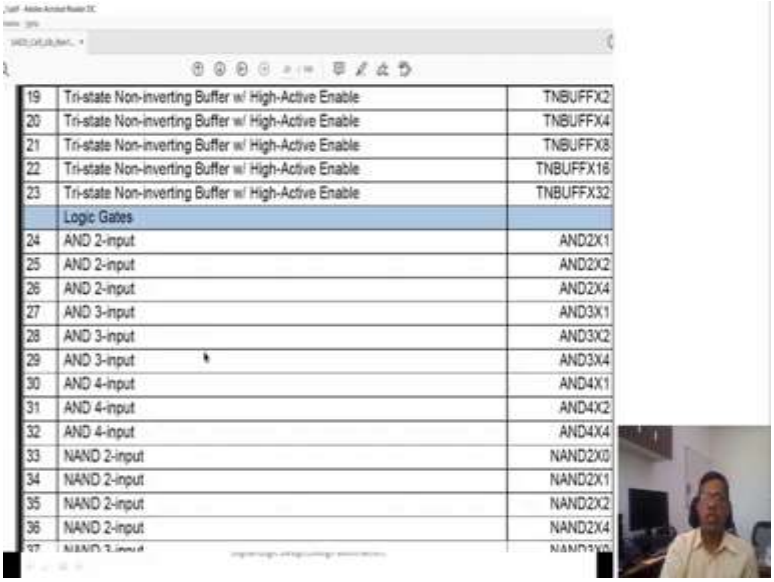


4	Inverter	INVX4
5	Inverter	INVX8
6	Inverter	INVX16
7	Inverter	INVX32
8	Inverting Buffer	IBUFFX2
9	Inverting Buffer	IBUFFX4
10	Inverting Buffer	IBUFFX8
11	Inverting Buffer	IBUFFX16
12	Inverting Buffer	IBUFFX32
13	Non-inverting Buffer	NBUFFX2
14	Non-inverting Buffer	NBUFFX4
15	Non-inverting Buffer	NBUFFX8
16	Non-inverting Buffer	NBUFFX16
17	Non-inverting Buffer	NBUFFX32
18	Tri-state Non-inverting Buffer w/ High-Active Enable	TNBUFFX1

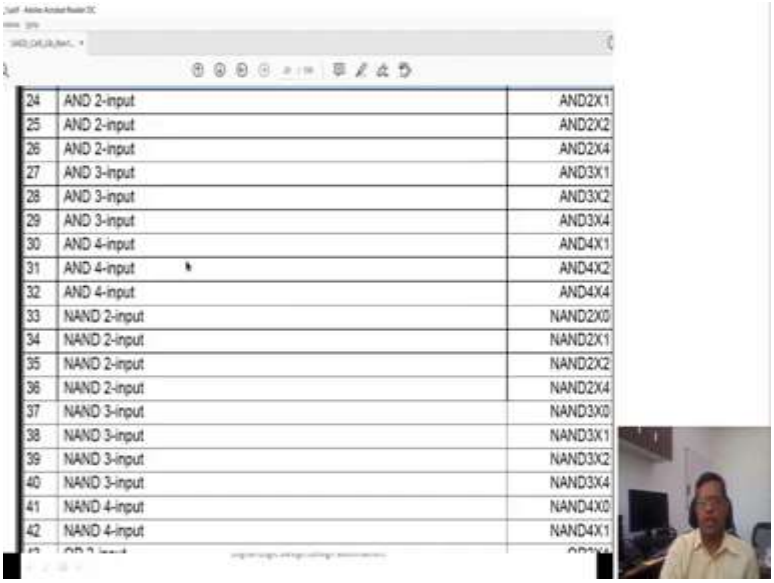
So, this is a digital standard cell library from the synopsis, it is for a 90 nanometer edk. So, when we see the detail so couple of things I would like to show in this detail. So, one thing I would like to show is the type of standard cells available. So, you see there are some six or seven type of inverters which are present, you see there are some six or seven type of inverters which are present.

So, when they say inverter x0 so that means the load is the minimum, here it can take inverter x32 can take up to fan out of 32. So, similarly, known inverting, inverting buffers are there so these buffers would be able to take more load. So, similarly non-inverting buffers so which are just the, so let us say one particular cell would have one particular gate has multiple fan out, so then non-inverting buffer so that the signal will not change but it would act as a buffer, it would be able to take much more fan out.

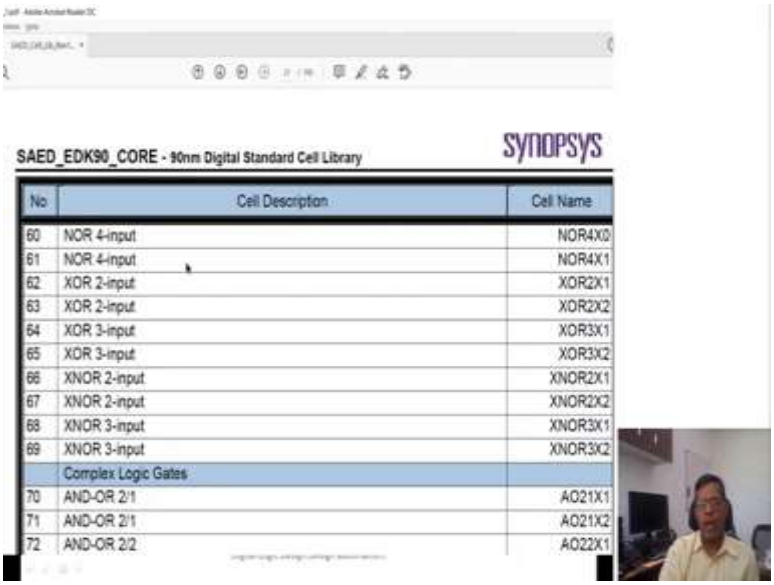
(Refer Slide Time: 16:07)



19	Tri-state Non-inverting Buffer w/ High-Active Enable	TNBUFFX2
20	Tri-state Non-inverting Buffer w/ High-Active Enable	TNBUFFX4
21	Tri-state Non-inverting Buffer w/ High-Active Enable	TNBUFFX8
22	Tri-state Non-inverting Buffer w/ High-Active Enable	TNBUFFX16
23	Tri-state Non-inverting Buffer w/ High-Active Enable	TNBUFFX32
	Logic Gates	
24	AND 2-input	AND2X1
25	AND 2-input	AND2X2
26	AND 2-input	AND2X4
27	AND 3-input	AND3X1
28	AND 3-input	AND3X2
29	AND 3-input	AND3X4
30	AND 4-input	AND4X1
31	AND 4-input	AND4X2
32	AND 4-input	AND4X4
33	NAND 2-input	NAND2X0
34	NAND 2-input	NAND2X1
35	NAND 2-input	NAND2X2
36	NAND 2-input	NAND2X4



24	AND 2-input	AND2X1
25	AND 2-input	AND2X2
26	AND 2-input	AND2X4
27	AND 3-input	AND3X1
28	AND 3-input	AND3X2
29	AND 3-input	AND3X4
30	AND 4-input	AND4X1
31	AND 4-input	AND4X2
32	AND 4-input	AND4X4
33	NAND 2-input	NAND2X0
34	NAND 2-input	NAND2X1
35	NAND 2-input	NAND2X2
36	NAND 2-input	NAND2X4
37	NAND 3-input	NAND3X0
38	NAND 3-input	NAND3X1
39	NAND 3-input	NAND3X2
40	NAND 3-input	NAND3X4
41	NAND 4-input	NAND4X0
42	NAND 4-input	NAND4X1



So, you see in the logic gate also there are two input AND gates with multiple load factors, three input NAND, two input AND gate, three input AND gate, four input AND gate. So, this is also one of the reason that during the course, during the course we were kind of advocating that if the input is more than four input it is going to detroit.

So, because in the standard cell libraries like this they also take at most four input then after that they will break it into multiple of the AND gate. So, let us say if we would like to design a nine input AND gate so it would be designed using a four input and then another four input and another four input, two input, so or maybe two levels of three input AND gate.

So, similarly NAND gates are there or NOR, NOR, XOR, and XNOR, so all of these gates are available as a basic gate.

(Refer Slide Time: 17:13)

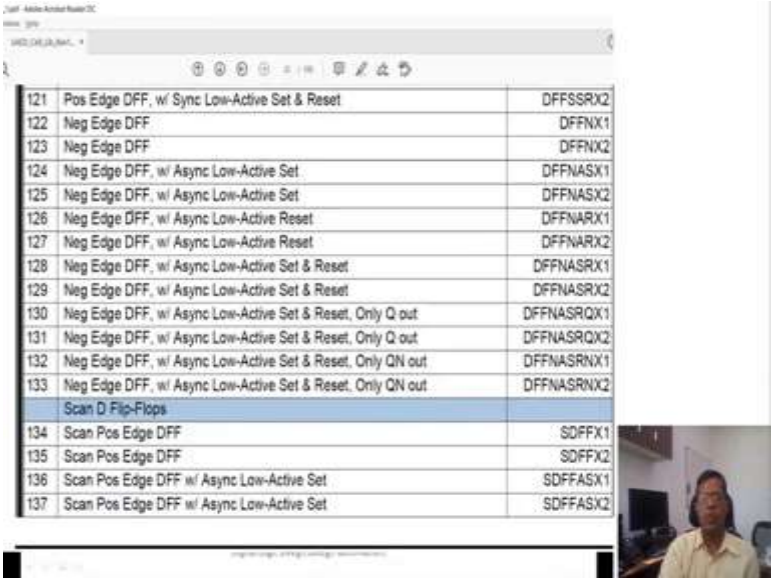
68	XNOR 3-input	XNOR3X1
69	XNOR 3-input	XNOR3X2
	Complex Logic Gates	
70	AND-OR 2/1	AO21X1
71	AND-OR 2/1	AO21X2
72	AND-OR 2/2	AO22X1
73	AND-OR 2/2	AO22X2
74	AND-OR 2/2/1	AO221X1
75	AND-OR 2/2/1	AO221X2
76	AND-OR 2/2/2	AO222X1
77	AND-OR 2/2/2	AO222X2
78	AND-OR-Invert 2/1	AOI21X1
79	AND-OR Invert 2/1	AOI21X2
80	AND-OR-Invert 2/2	AOI22X1
81	AND-OR-Invert 2/2	AOI22X2
82	AND-OR-Invert 2/2/1	AOI221X1
83	AND-OR-Invert 2/2/1	AOI221X2
84	AND-OR-Invert 2/2/2	AOI222X1
85	AND-OR-Invert 2/2/2	AOI222X2
86	OR-AND 2/1	OA21X1

103	Multiplexer 2 to 1	MUX21X2
104	Multiplexer 4 to 1	MUX41X1
105	Multiplexer 4 to 1	MUX41X2
	Decoders	
106	Decoder 2 to 4	DEC24X1
107	Decoder 2 to 4	DEC24X2
	Adders and Subtractors	
108	Half Adder 1 bit	HADDX1
109	Half Adder 1 bit	HADDX2
110	Full Adder 1 bit	FADDX1
111	Full Adder 1 bit	FADDX2
	D Flip-Flops	
112	Pos Edge DFF	DFFX1
113	Pos Edge DFF	DFFX2
114	Pos Edge DFF, w/ Async Low-Active Set	DFFASX1
115	Pos Edge DFF, w/ Async Low-Active Set	DFFASX2
116	Pos Edge DFF, w/ Async Low-Active Reset	DFFARX1
117	Pos Edge DFF, w/ Async Low-Active Reset	DFFARX2
118	Pos Edge DFF, w/ Async Low-Active Set & Reset	DFFASRX1

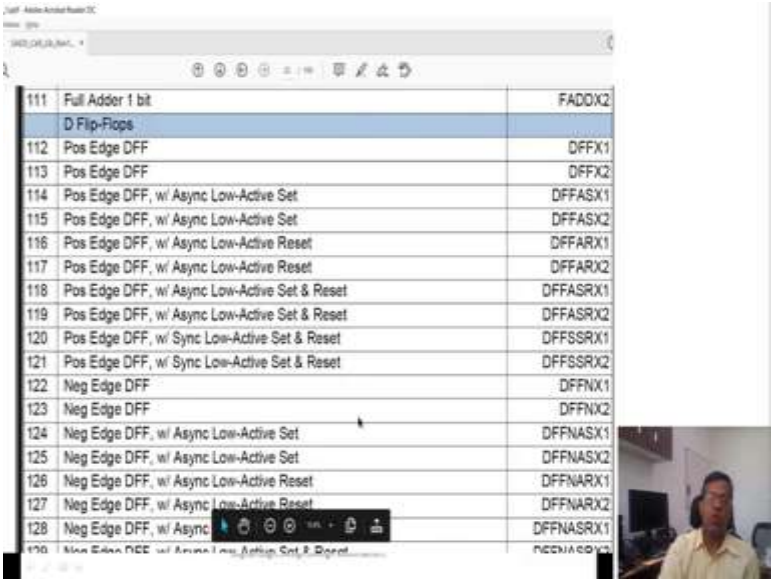
Some of the complex gates are also available where AND and OR are there so basically it takes two input to the AND gate and one of the input as or gate so this is a combination, so this is, this will take two AND gate and two OR gate as input, so it is a combination of gates so that in certain cases we can optimize, directly optimize, directly map our circuit to these gates and these gates are going to be much more optimized than if we use one AND gate and one OR gate.

So, such complex gates are also available in the standard cell library which would help in the improvement of the performance. So, some of the small decoders, multiplexers, half adders are also there.

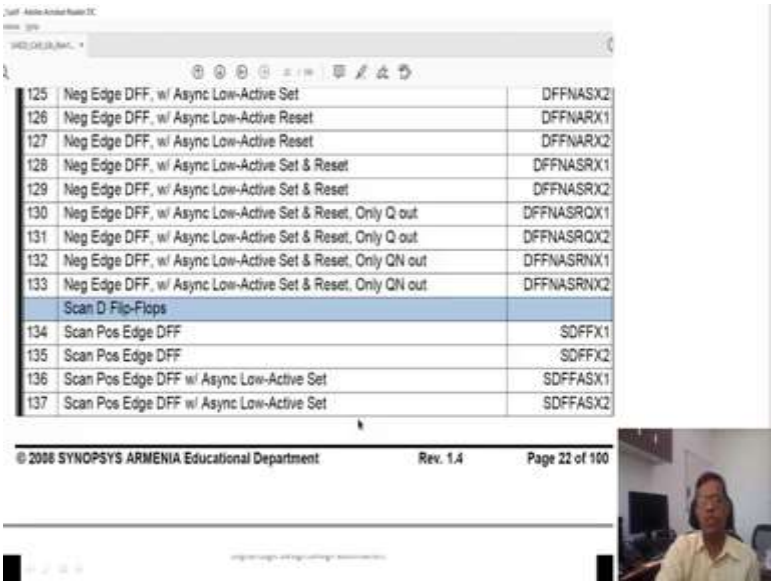
(Refer Slide Time: 18:04)



121	Pos Edge DFF, w/ Sync Low-Active Set & Reset	DFFSSRX2
122	Neg Edge DFF	DFFNX1
123	Neg Edge DFF	DFFNX2
124	Neg Edge DFF, w/ Async Low-Active Set	DFFNASX1
125	Neg Edge DFF, w/ Async Low-Active Set	DFFNASX2
126	Neg Edge DFF, w/ Async Low-Active Reset	DFFNARX1
127	Neg Edge DFF, w/ Async Low-Active Reset	DFFNARX2
128	Neg Edge DFF, w/ Async Low-Active Set & Reset	DFFNASRX1
129	Neg Edge DFF, w/ Async Low-Active Set & Reset	DFFNASRX2
130	Neg Edge DFF, w/ Async Low-Active Set & Reset, Only Q out	DFFNASRQX1
131	Neg Edge DFF, w/ Async Low-Active Set & Reset, Only Q out	DFFNASRQX2
132	Neg Edge DFF, w/ Async Low-Active Set & Reset, Only QN out	DFFNASRNX1
133	Neg Edge DFF, w/ Async Low-Active Set & Reset, Only QN out	DFFNASRNX2
	Scan D Flip-Flops	
134	Scan Pos Edge DFF	SDFFX1
135	Scan Pos Edge DFF	SDFFX2
136	Scan Pos Edge DFF w/ Async Low-Active Set	SDFFASX1
137	Scan Pos Edge DFF w/ Async Low-Active Set	SDFFASX2

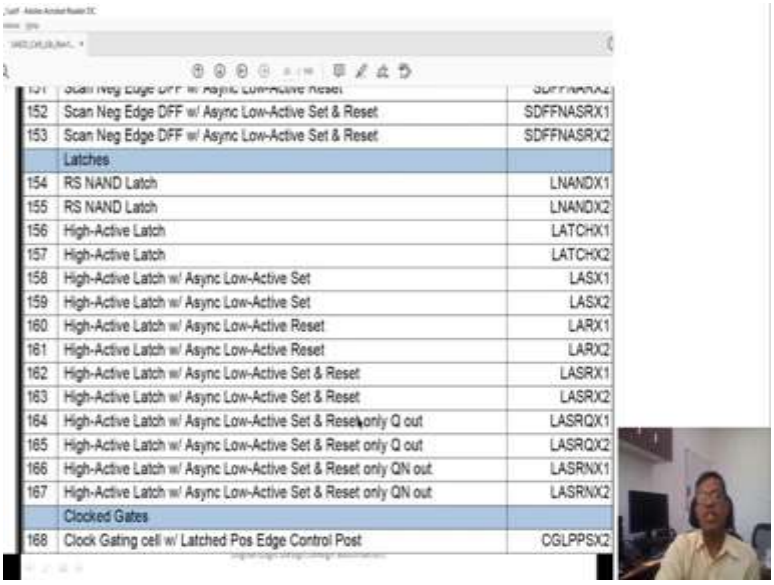


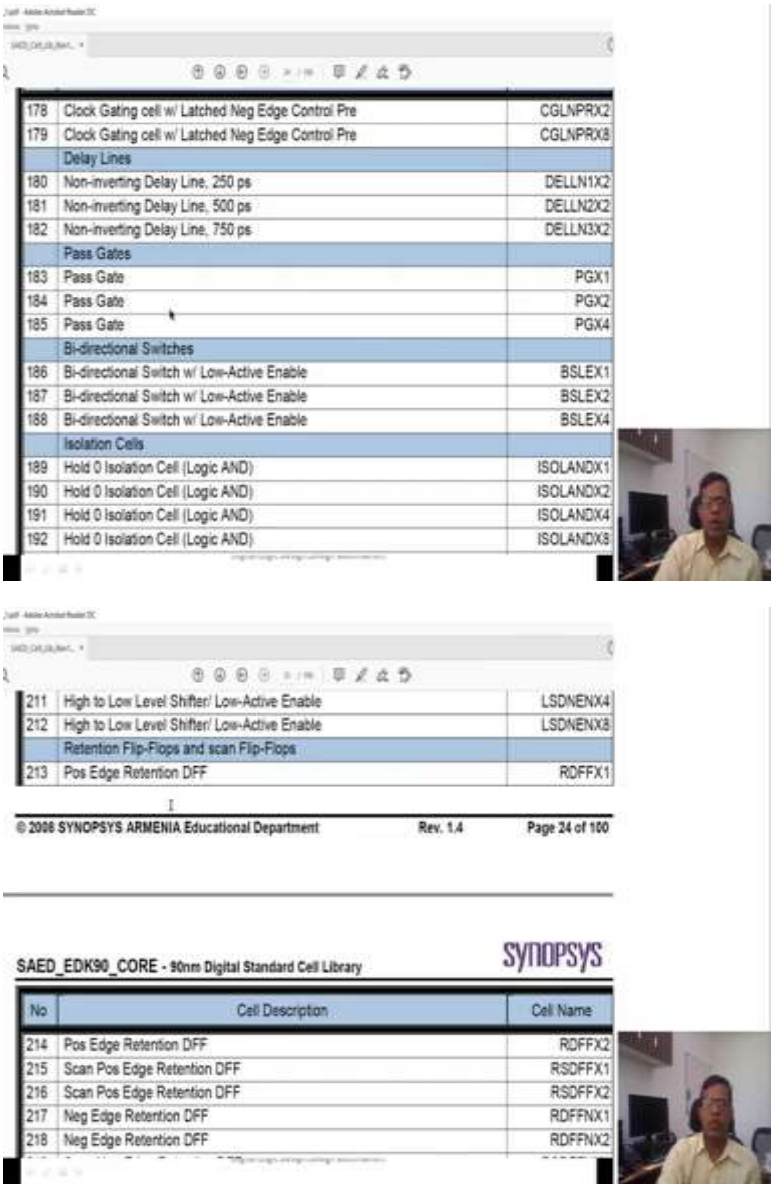
111	Full Adder 1 bit	FADDX2
	D Flip-Flops	
112	Pos Edge DFF	DFFX1
113	Pos Edge DFF	DFFX2
114	Pos Edge DFF, w/ Async Low-Active Set	DFFASX1
115	Pos Edge DFF, w/ Async Low-Active Set	DFFASX2
116	Pos Edge DFF, w/ Async Low-Active Reset	DFFARX1
117	Pos Edge DFF, w/ Async Low-Active Reset	DFFARX2
118	Pos Edge DFF, w/ Async Low-Active Set & Reset	DFFASRX1
119	Pos Edge DFF, w/ Async Low-Active Set & Reset	DFFASRX2
120	Pos Edge DFF, w/ Sync Low-Active Set & Reset	DFFSSRX1
121	Pos Edge DFF, w/ Sync Low-Active Set & Reset	DFFSSRX2
122	Neg Edge DFF	DFFNX1
123	Neg Edge DFF	DFFNX2
124	Neg Edge DFF, w/ Async Low-Active Set	DFFNASX1
125	Neg Edge DFF, w/ Async Low-Active Set	DFFNASX2
126	Neg Edge DFF, w/ Async Low-Active Reset	DFFNARX1
127	Neg Edge DFF, w/ Async Low-Active Reset	DFFNARX2
128	Neg Edge DFF, w/ Async	DFFNASRX1



Now, you see only the D flip flop is present, you will not see JK flip flop, RS flip flow or T flip flop because, but D flip flop is present in almost different kind of variants like D flip flop with a synchronous reset set, with asynchronous reset set as well as reset, so these positive edge, similarly negative edge so all variants of D flip flops are present but not the other flip flops. Since, if we have D flip flow we can also convert, we can design other flip flops also. So, scan D flip flop is a very special type of D flip flop which is used for sequential purpose.

(Refer Slide Time: 18:50)





178	Clock Gating cell w/ Latched Neg Edge Control Pre	CGLNPRX2
179	Clock Gating cell w/ Latched Neg Edge Control Pre	CGLNPRX8
	Delay Lines	
180	Non-inverting Delay Line, 250 ps	DELLN1X2
181	Non-inverting Delay Line, 500 ps	DELLN2X2
182	Non-inverting Delay Line, 750 ps	DELLN3X2
	Pass Gates	
183	Pass Gate	PGX1
184	Pass Gate	PGX2
185	Pass Gate	PGX4
	Bi-directional Switches	
186	Bi-directional Switch w/ Low-Active Enable	BSLEX1
187	Bi-directional Switch w/ Low-Active Enable	BSLEX2
188	Bi-directional Switch w/ Low-Active Enable	BSLEX4
	Isolation Cells	
189	Hold 0 Isolation Cell (Logic AND)	ISOLANDX1
190	Hold 0 Isolation Cell (Logic AND)	ISOLANDX2
191	Hold 0 Isolation Cell (Logic AND)	ISOLANDX4
192	Hold 0 Isolation Cell (Logic AND)	ISOLANDX8

211	High to Low Level Shifter/ Low-Active Enable	LSDNENX4
212	High to Low Level Shifter/ Low-Active Enable	LSDNENX8
	Retention Flip-Flops and scan Flip-Flops	
213	Pos Edge Retention DFF	RDFFX1

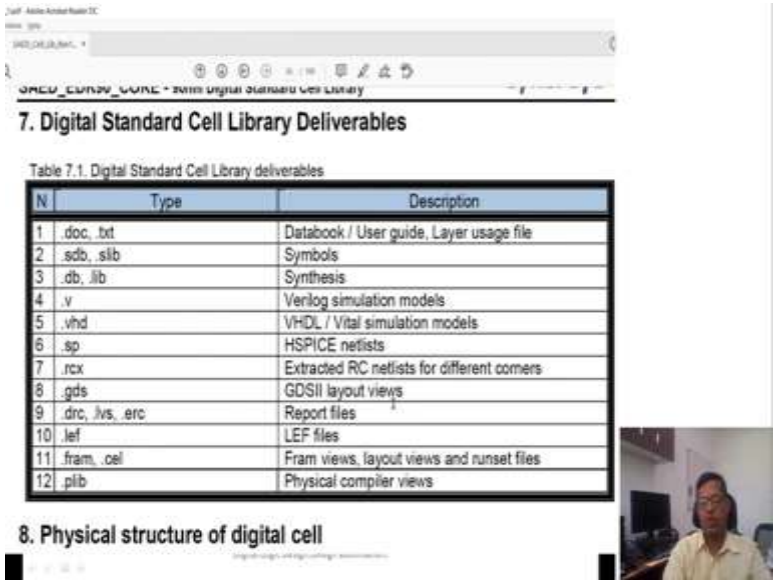
SAED_EDK90_CORE - 90nm Digital Standard Cell Library

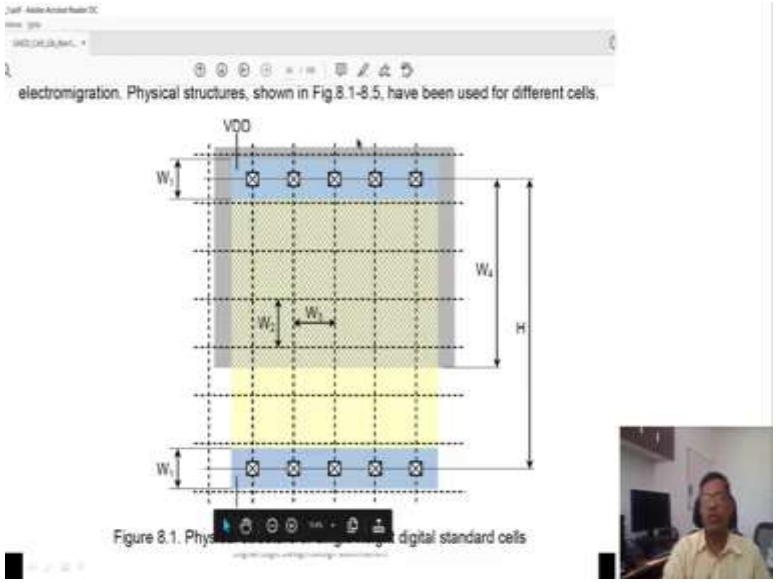
SYNOPSYS

No	Cell Description	Cell Name
214	Pos Edge Retention DFF	RDFFX2
215	Scan Pos Edge Retention DFF	RSDFFX1
216	Scan Pos Edge Retention DFF	RSDFFX2
217	Neg Edge Retention DFF	RDFFNX1
218	Neg Edge Retention DFF	RDFFNX2

Now, different kind of latches, in latches we have RS NAND, RS NAND type latch. So, all of these other gates pass transistors delay lines so these are isolation gates these are for low power or different use cases. So, that was not the purpose.

(Refer Slide Time: 19:16)





Now, the other thing which yeah, so once a digital standard cell library has been given as an input it also gives basically symbols, synthesis and verilog simulation models, HSPICE netlists, so all of these things are will come with the standard cell library, so it can help in the digital design so that we can estimate area, power, etcetera and we can also simulate the design.

And on the second side it can also help us in the physical compilation which we will talk about in couple of slides. So, physical compilation view is there, report files, GDSII layout view is there so that we can convert into layout also. So, yeah, so this is what is the meaning of physical layout that it means that if this is the overall cell that from the VDD so from the power supply to

ground this is going to be the dimension, this is the width 1, width 2 and it is formed in the form of a great.