DEPARTMENT OF COMPUTER SYSTEM ENGINEERING

Digital Integrated Circuits - ENCS333

Dr. Khader Mohammad

Lecture #7

Static CMOS Logic

Digital Integrated Circuits

	Course topics and Schedule
	Subject
1	Introduction to Digital Integrated Circuits Design
2	Semiconductor material: pn-junction, NMOS, PMOS
3	IC Manufacturing and Design Metrics CMOS
4	Transistor Devices and Logic Design The CMOS inverter
5	Combinational logic structures
6	Layout of an Inverter and basic gates
7 8	***Static & Dynamic CMOS Logic***
9	Sequential logic gates; Latches and Flip-Flops
	Parasitic Capacitance Estimation
	Device modeling parameterization from I-V curves.
	Short Test
	Arithmetic building blocks
	Interconnect: R, L and C - Wire modeling
	Timing
	Power dissipation;
	SPICE Simulation Techniques (Project)
	Memories and array structures
	Midterm
	Clock Distribution
	Supply and Threshold Voltage Scaling
	Reliability and IC qualification process
	Advanced Voltage Scaling Techniques

PROCESS FILE LINK:

- https://github.com/siwS/vlsi/blob/master/vlsi_2011/MW3.1.7/MW3.1.7/rules/cmos65n.rul
- http://ptm.asu.edu/
- www.mosis.orgwww.mosis.org
-

Agenda

- **General complementary logic design,**

perspective, stick-figure circuit diagrams

- **•Examples: constructing PDN/PUN duals,**

logic -> circuit, circuit -> logic,

circuit -> layout, layout -> circuit,

cross sectional views of layout

What is a MOSFET?

A resistor: $$R_n = \frac{1}{\mu_n C_{ox}(V_{GS} - V_{Tn})}\left(\frac{L}{W}\right)$$

- (among other things ...) Increasing W decreases the resistance; allows more current to flow

Oxide capacitance $C_{ox} = \varepsilon_{ox} / t_{ox}$ [F/cm^2]

Transconductance $\beta_n = \mu_n C_{ox}\left(\frac{W}{L}\right) = k'_n\left(\frac{W}{L}\right)$

Gate capacitance $C_G = C_{ox} WL$ [F]

nFET vs. pFET

$$R_n = \frac{1}{\beta_n (V_{DD} - V_{Tn})} \qquad \beta_n = \mu_n C_{ox} \left(\frac{W}{L}\right)_n$$

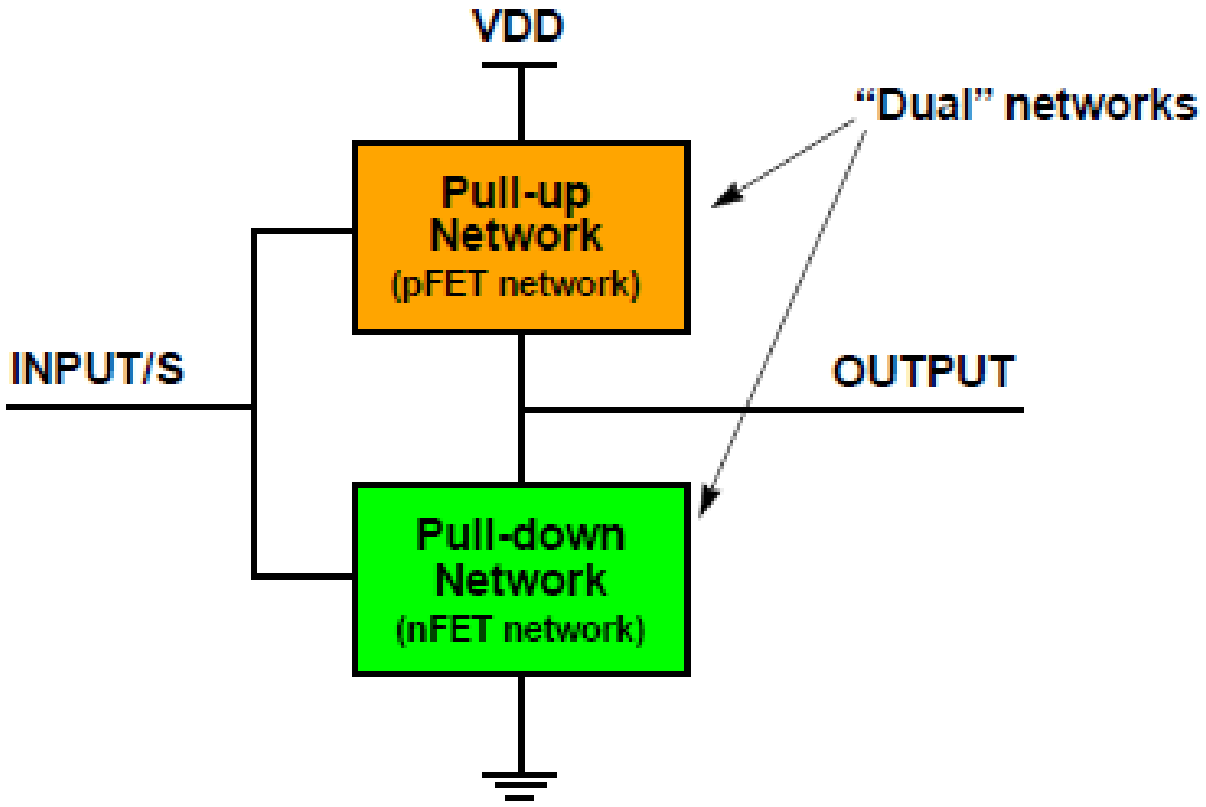
$$R_p = \frac{1}{\beta_p (V_{DD} - |V_{Tp}|)} \qquad \beta_p = \mu_p C_{ox} \left(\frac{W}{L}\right)_p$$

$$\frac{\mu_n}{\mu_p} = r \quad \text{Typically } (2 \ldots 3)$$

(μ is the carrier mobility through device)

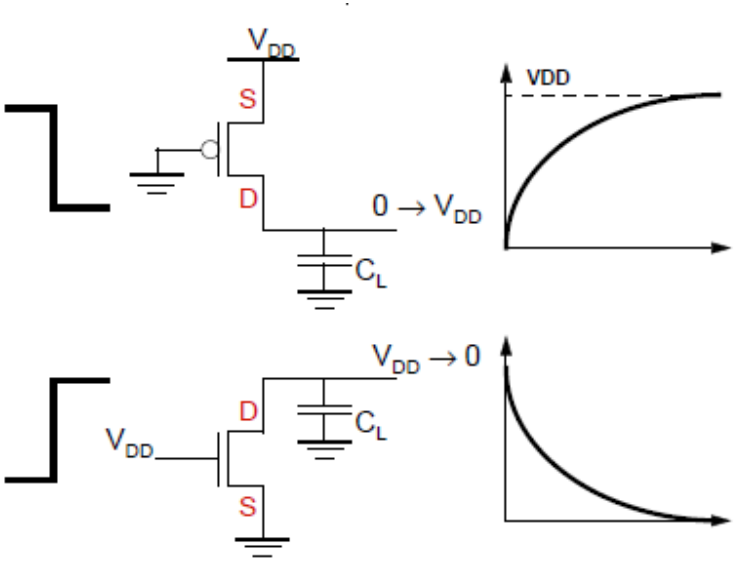
(We will return to this later …)

Complementary Design

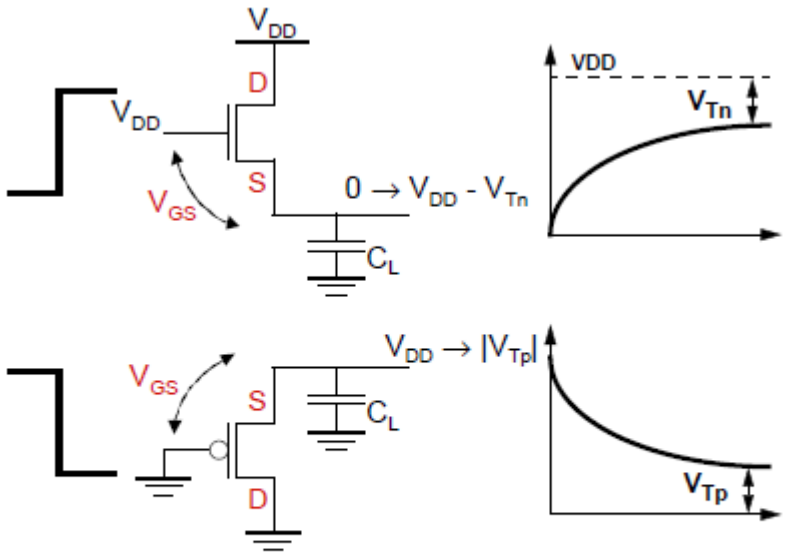


- PMOS "Pull-Up" Network (PUN)
- NMOS "Pull-Down" Network (PDN)

Why Division?

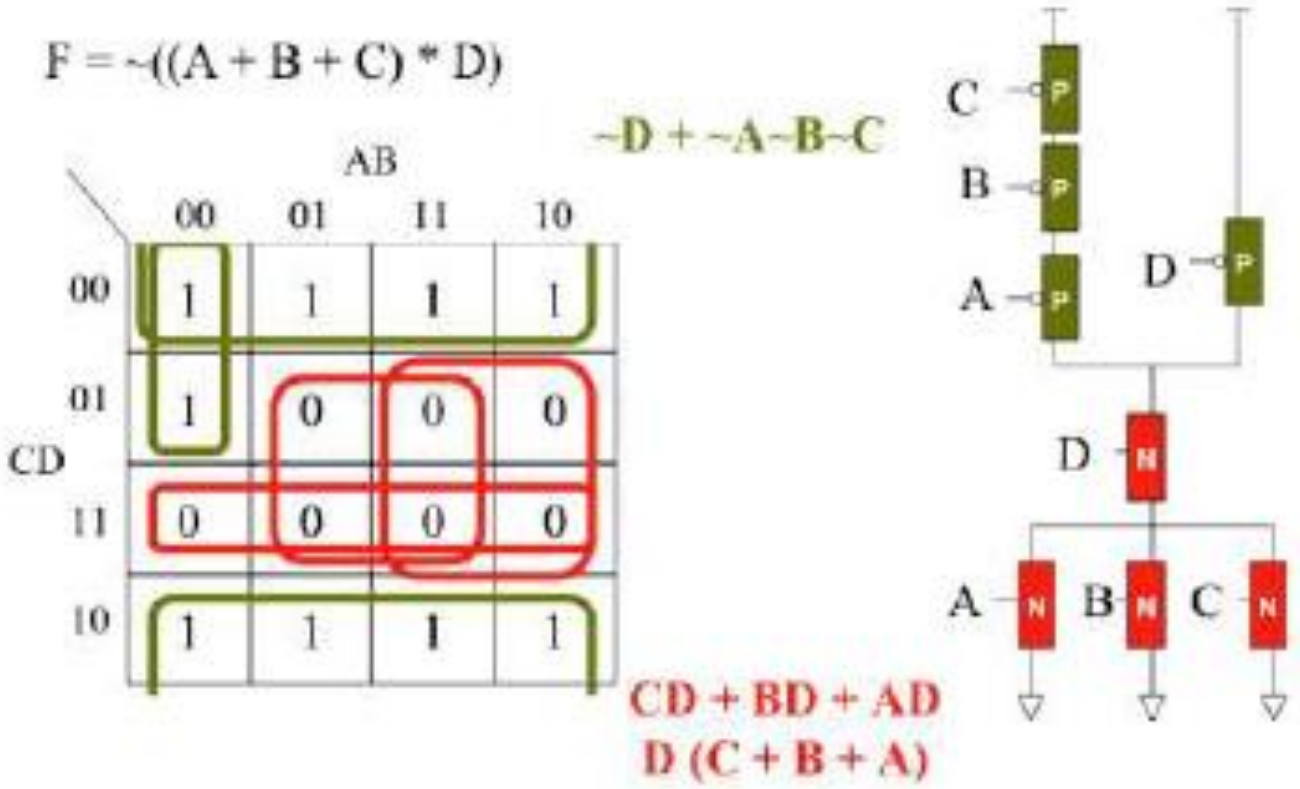


PMOS will pass a "1"
NMOS will pass a "0"



NMOS will not pass a "1"
PMOS will not pass a "0"

Implements Arbitrary Logic



- Pull-Up Network: on when function = 1
- Pull-Down Network: on when function = 0

Static CMOS: Perspective

- "Static" as in "output is logic function of inputs, and, given stable inputs, does not change over time"

- Propagation delay function of load capacitance and resistance of transistors

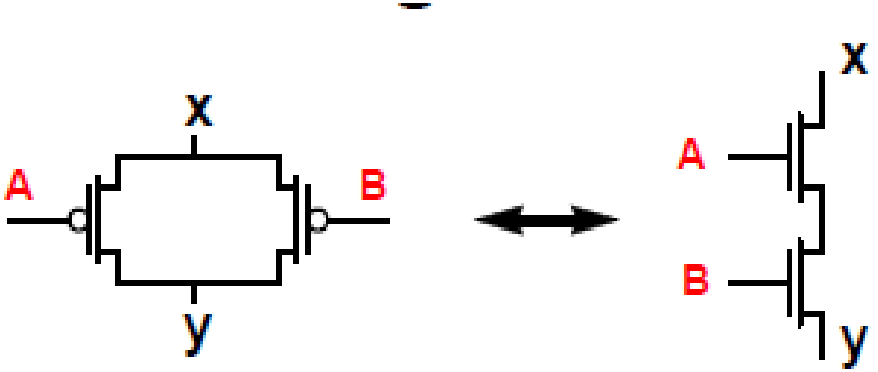
PROS:

- Full rail-to-rail swing; high noise margins
- Logic levels not dependent upon relative device sizes
- Always a path to Vdd or Gnd in steady state; low output impedance
- Extremely high input resistance; nearly zero steady-state input current
- No direct path steady state between power and ground; no static power dissipation

CONS:

- N inputs => 2N transistors in design

Constructing the Dual

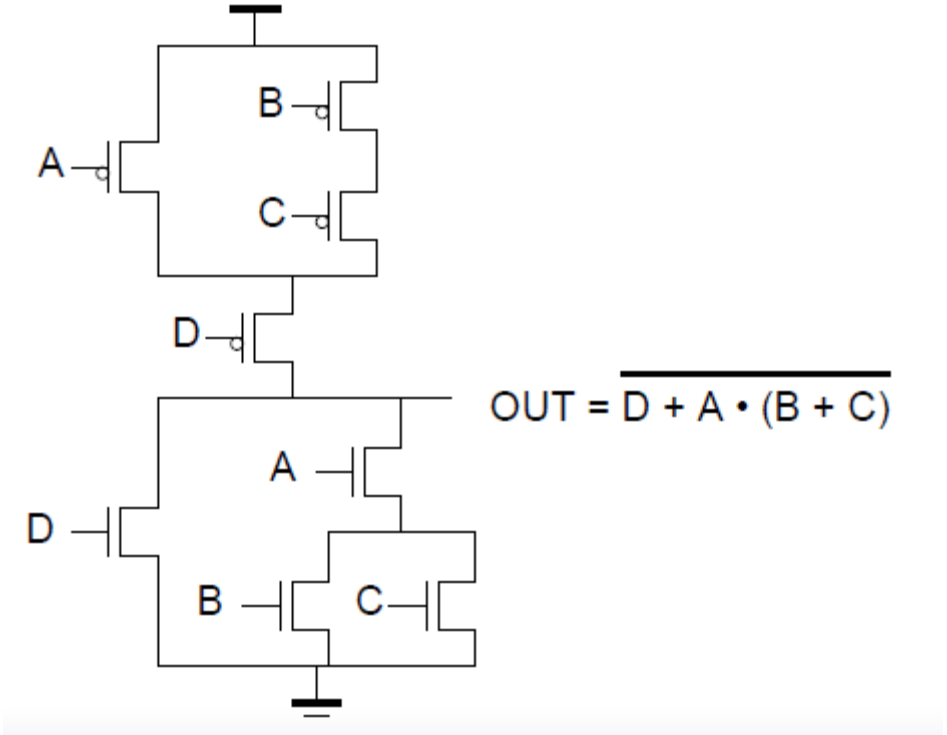


Transistors in parallel are in series in dual; transistors in series are in parallel in dual

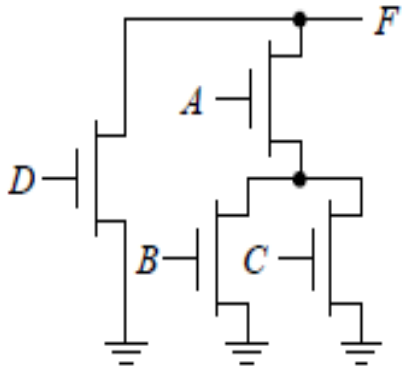
This is the physical realization of DeMorgan's theorems:

- **A + B = A • B [!(A + B) = !A • !B *or* !(A | B) = !A & !B]**
- **A • B = A + B [!(A • B) = !A + !B *or* !(A & B) = !A | !B]**

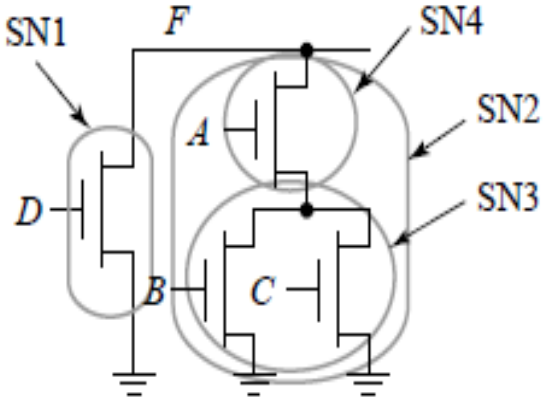
Complex CMOS Gate



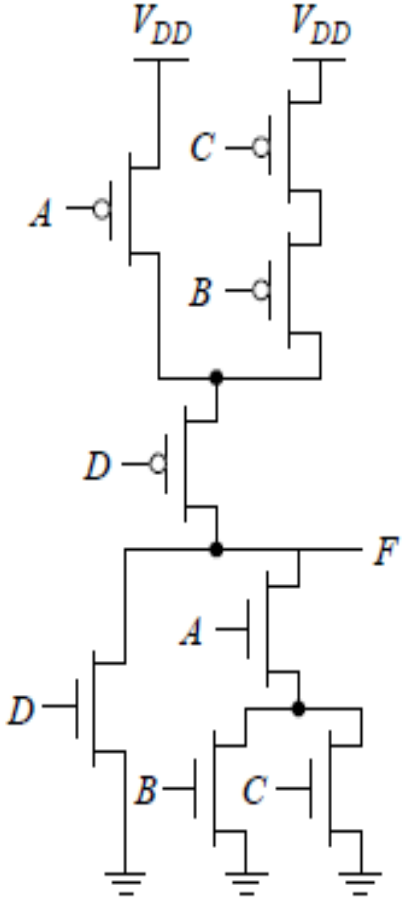
Constructing a Complex Gate



(a) pull-down network



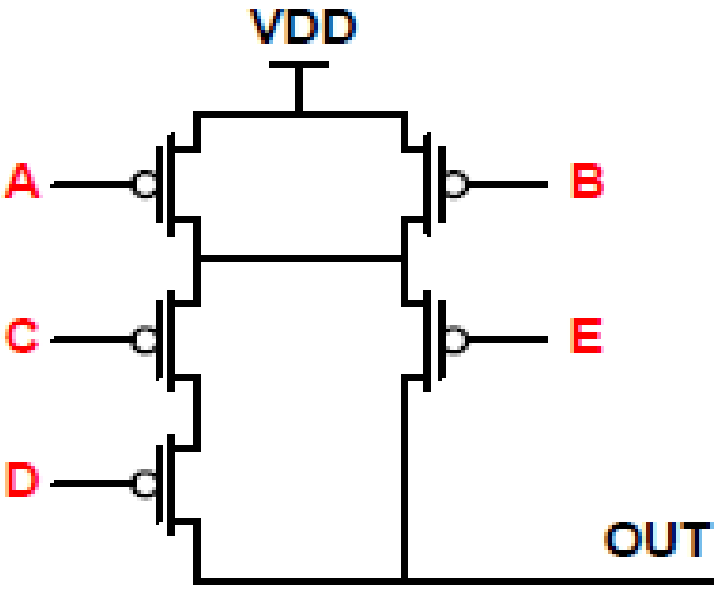
(b) Deriving the pull-up network hierarchically by identifying sub-nets



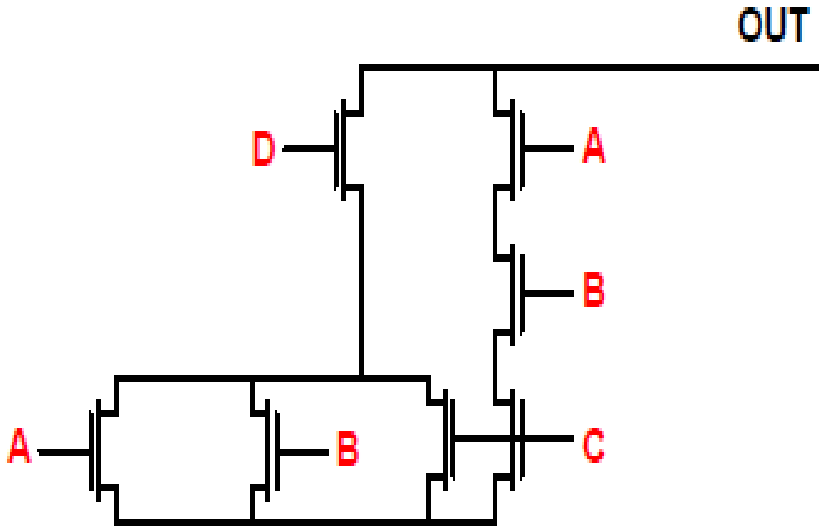
(c) complete gate

Complex CMOS Gate

Constructing the Dual

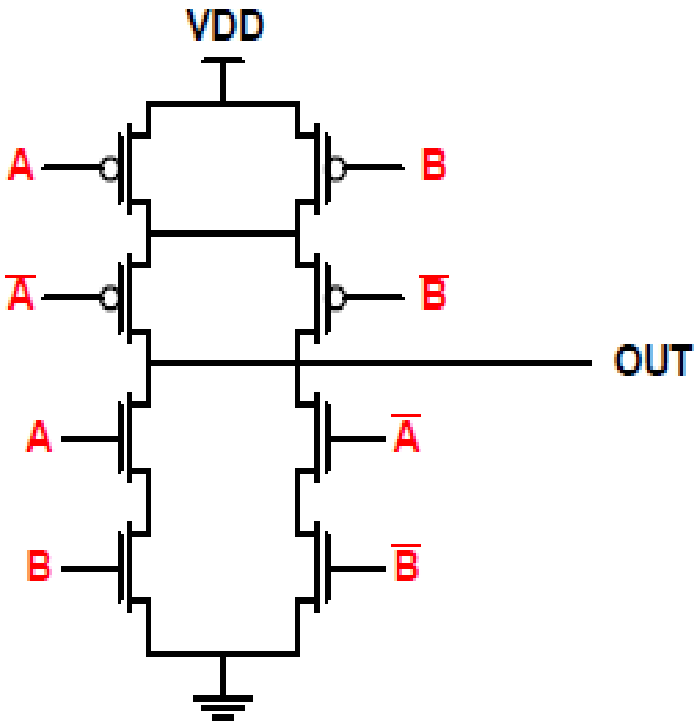


Constructing the Dual



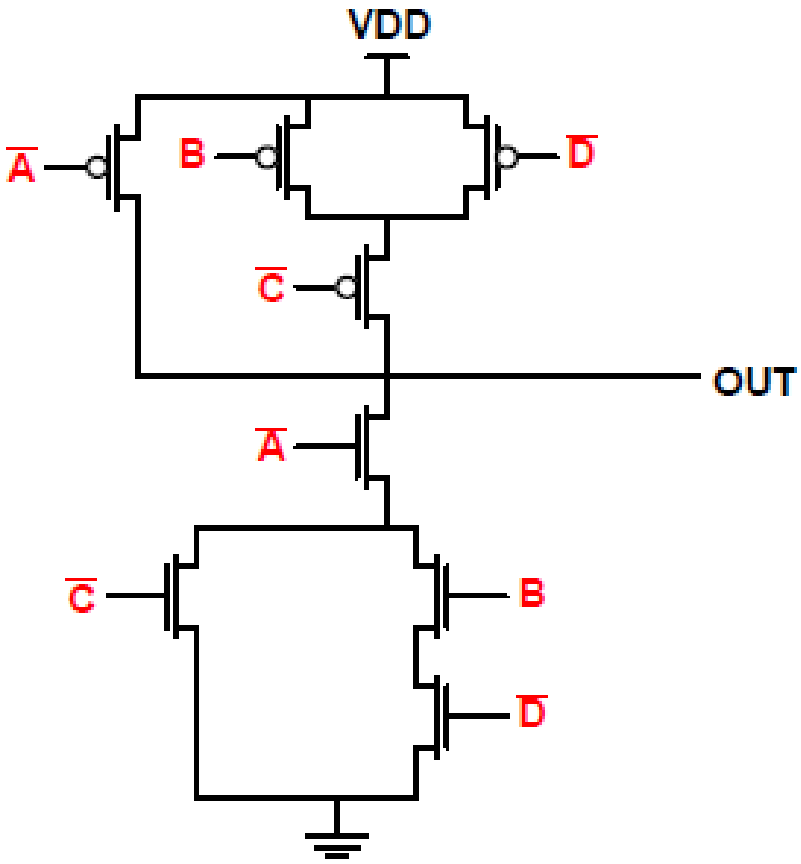
Examples: Logic <-> Circuit

XOR [out = ~($\overline{A}\,\overline{B}$ + A B)]



Examples: Logic <-> Circuit

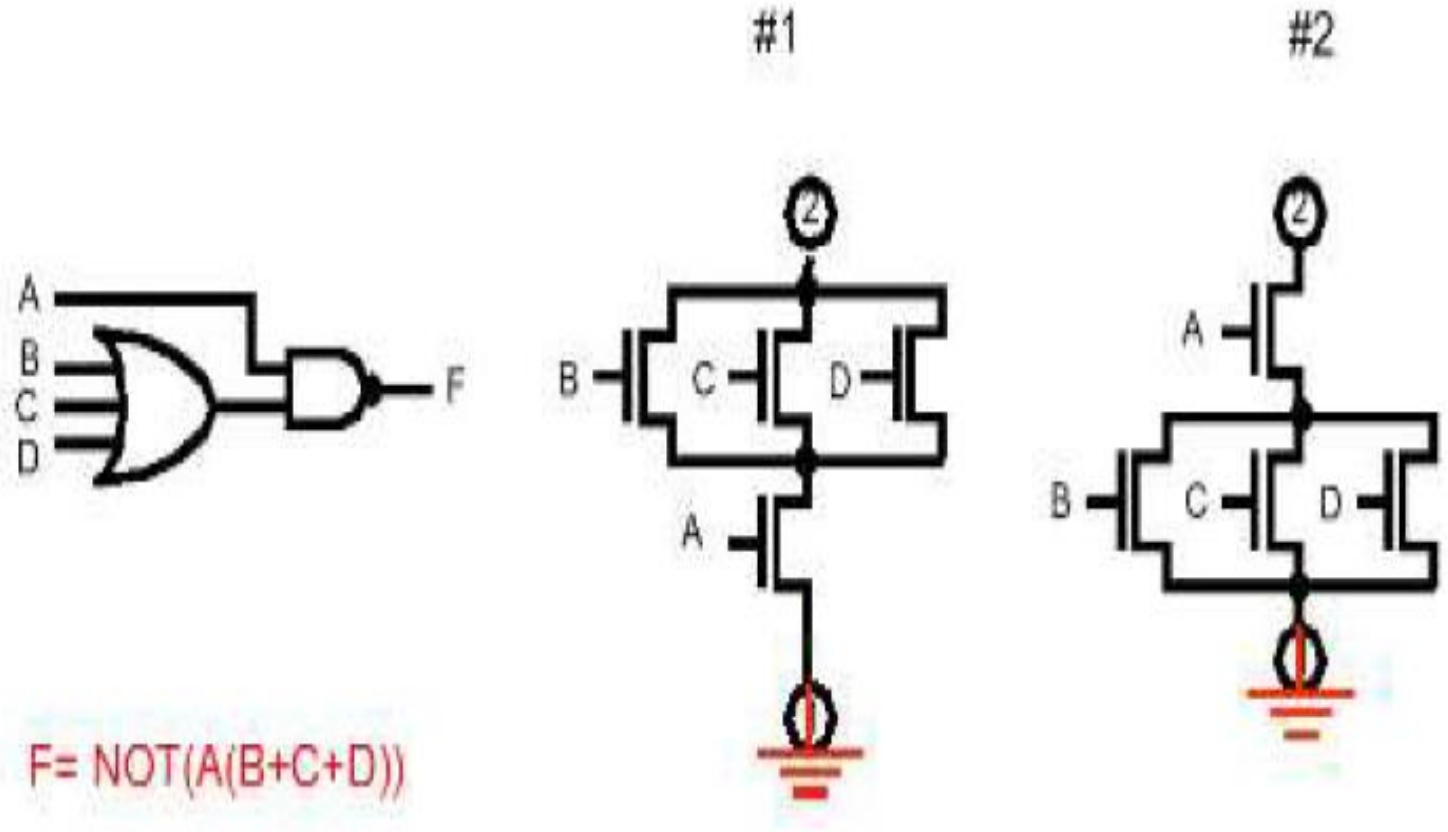
$out = \sim(\overline{A} \cdot (\overline{C} + B\overline{D}))$



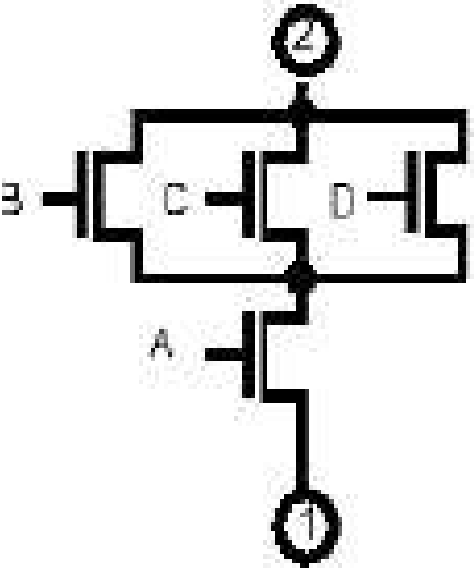
Examples: Layout <-> Circuit

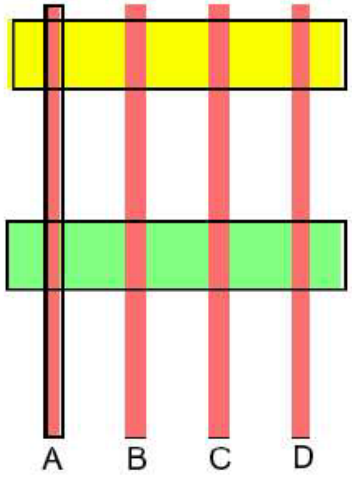
NOT ALL LAYOUTS ARE CREATED EQUAL

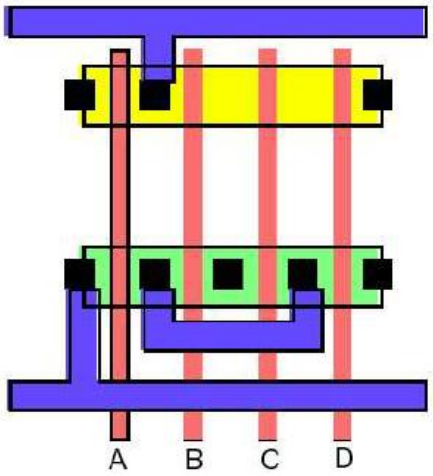
Let's look at two "equivalent" approaches

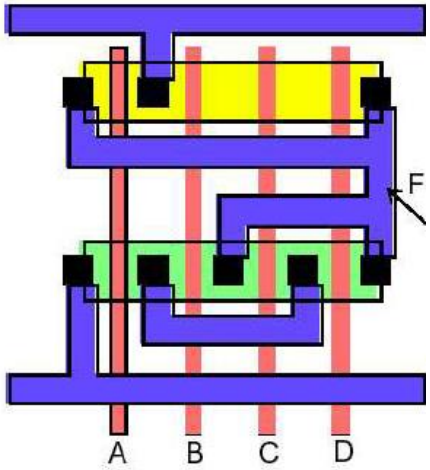


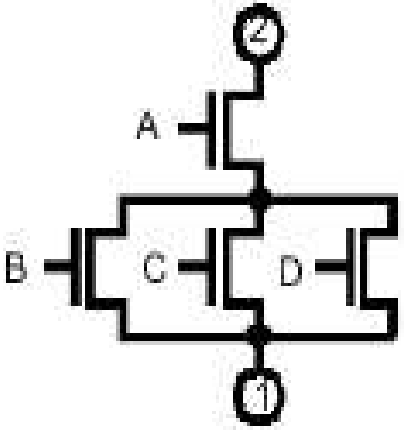
Examples: Layout <-> Circuit

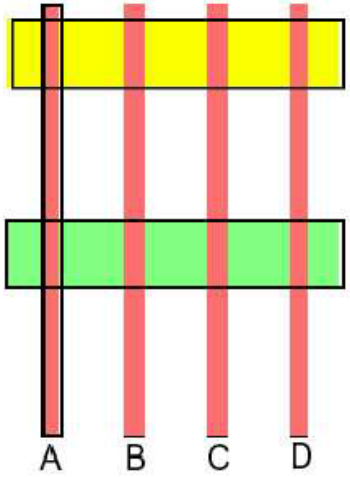


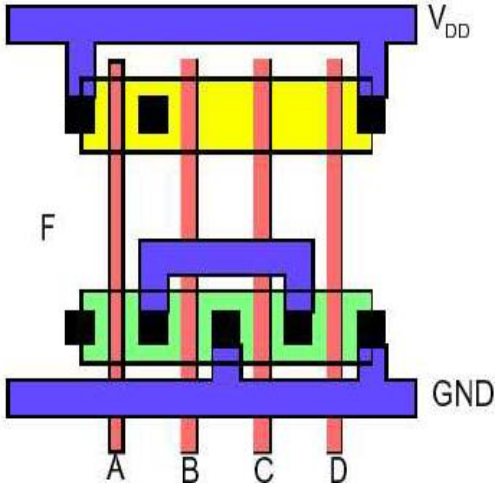


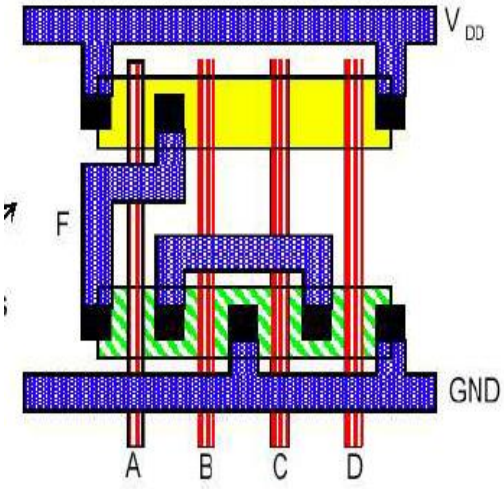






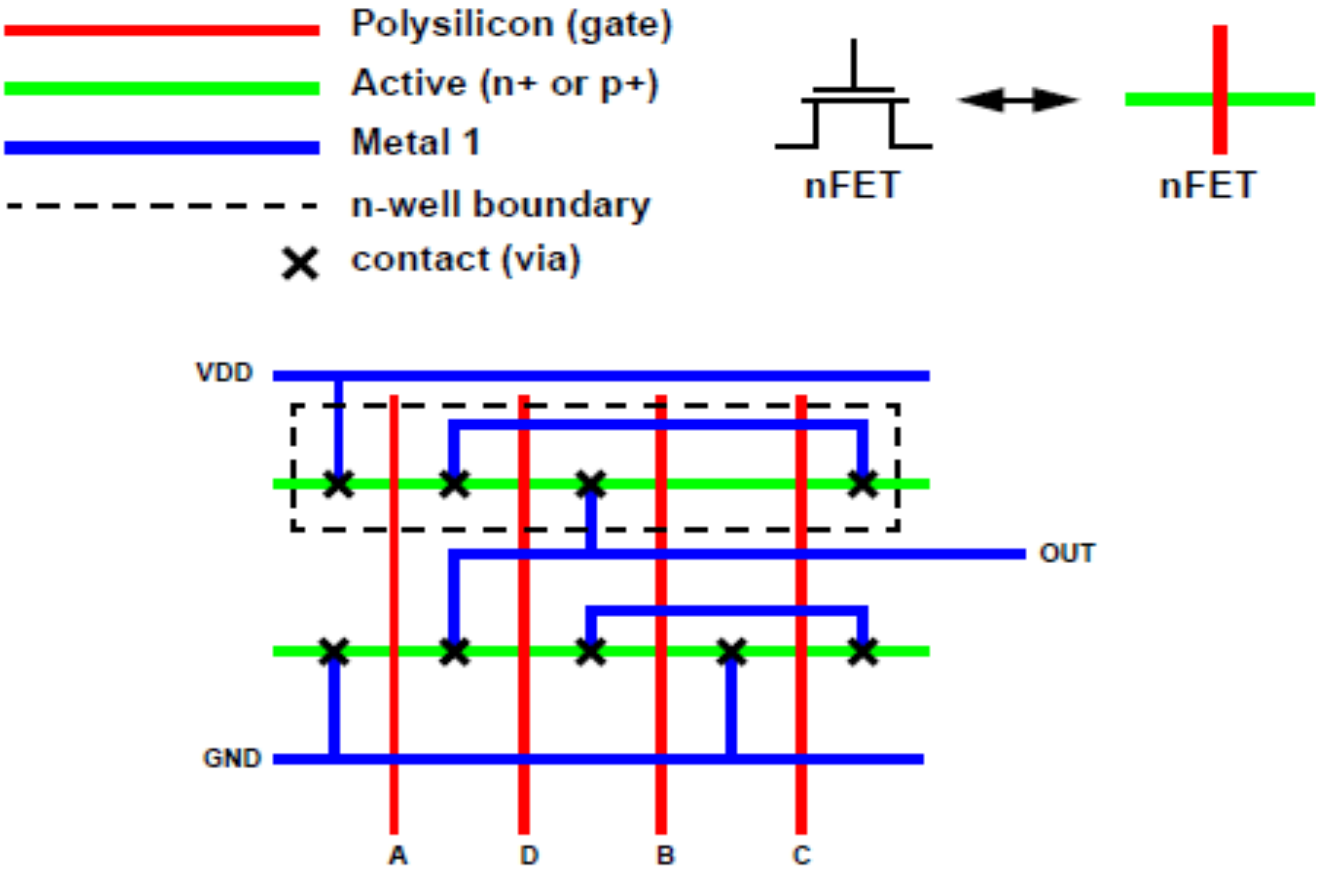






Stick Diagrams

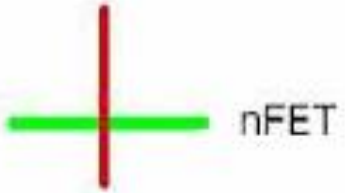
- **Introduced by Mead & Conway in 80's**
- **Every line of conduction-material layer is represented by line of distinct color**



Stick Diagrams

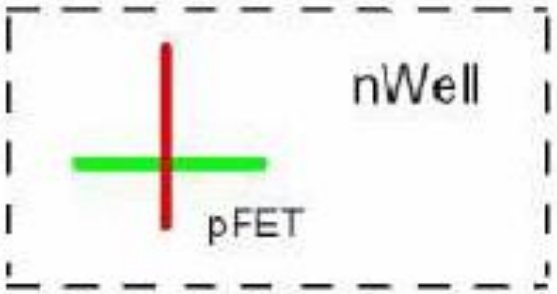
In terms of stick diagrams, we thus say that an nFET is formed whenever

Red (Poly) crosses over Green (Active)



This is consistent with a top view of the transistor.

A pFET is described by the same "red over green" coding, but the crossing point is contained within an nWell boundary

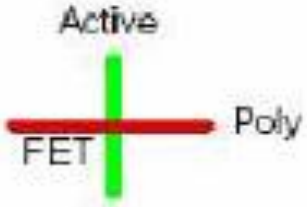


Stick Diagrams

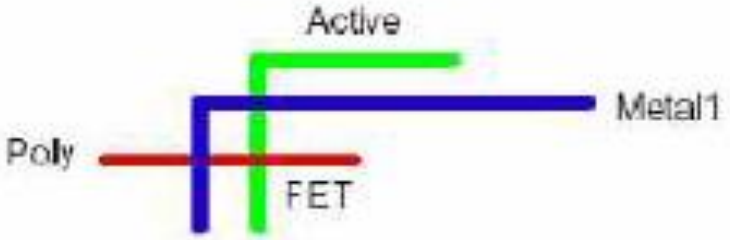
The rules for constructing stick diagrams are based on the characteristics of the conducting layers.

- Only the routing is important, not the line widths

- Red over green gives a FET
  (Poly) (Active)

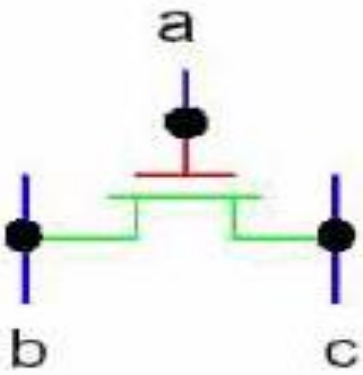


- Blue may cross over green or red without a connection
  (Metal1) (Active) (Poly)

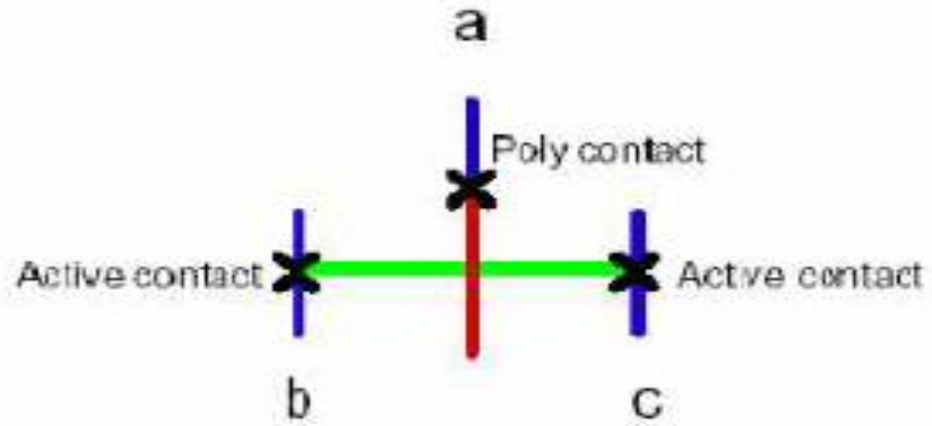


Stick Diagrams

Connections between layers are specified by ✖



Schematic

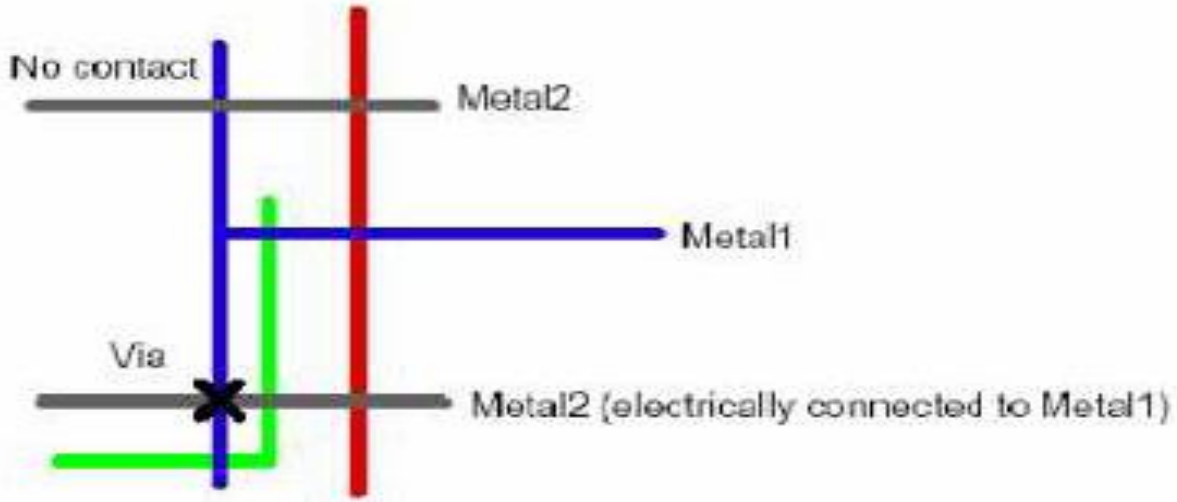


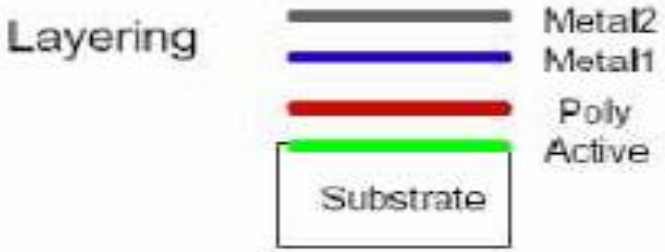
Stick diagram

Poly contact: Metal1-to-Poly
Active Contact: Metal1-to-Active

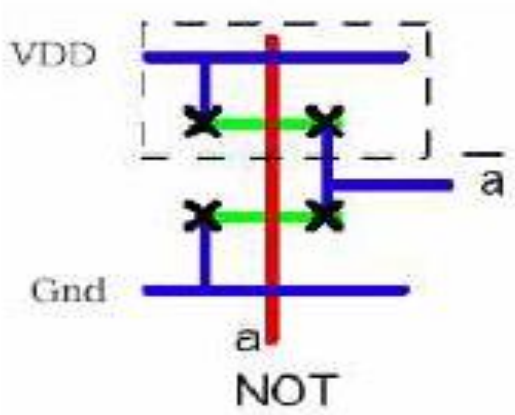
Stick Diagrams

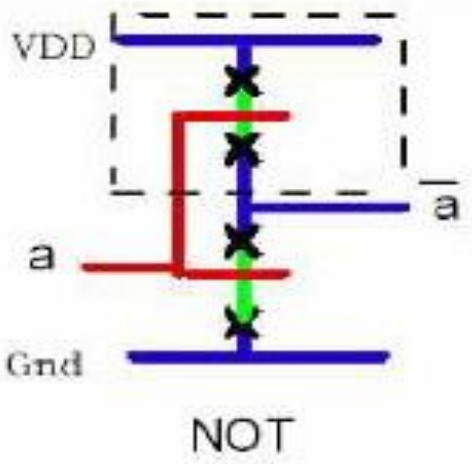
Metal lines on different layers can cross one another.
Contacting two metal lines requires a via

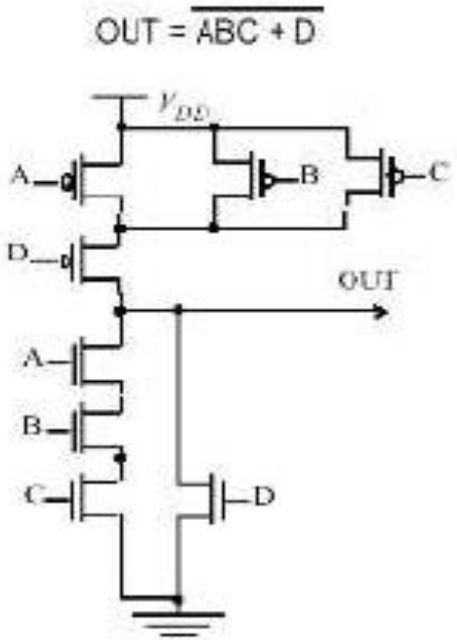


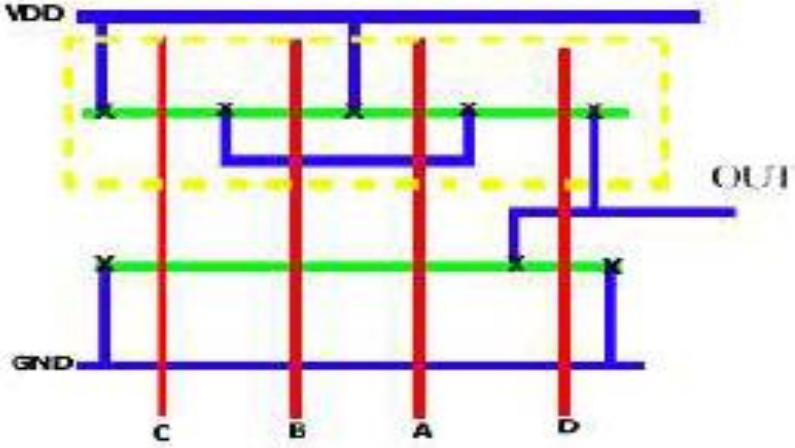


Stick Diagrams



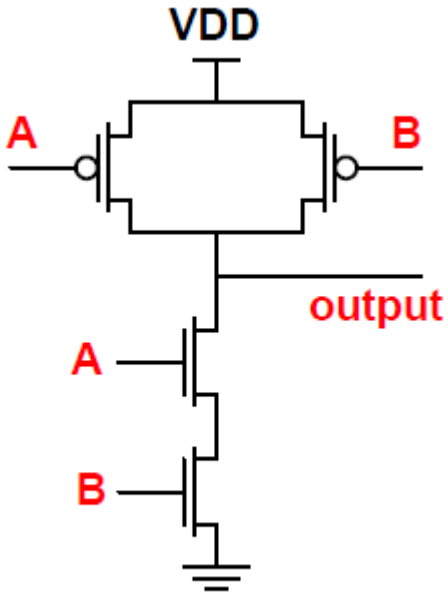


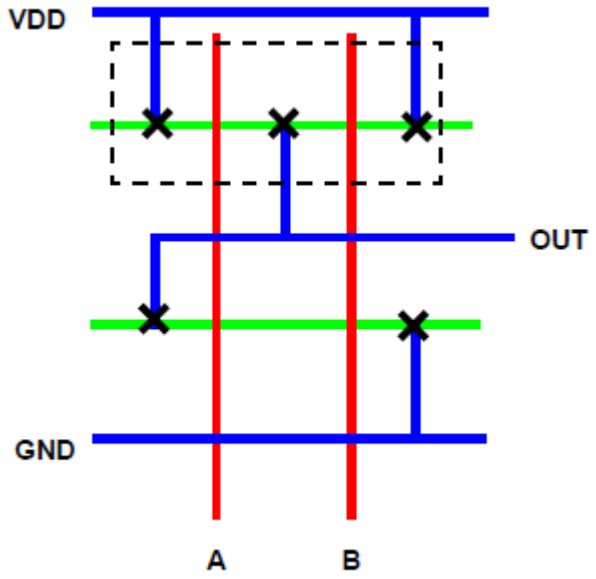




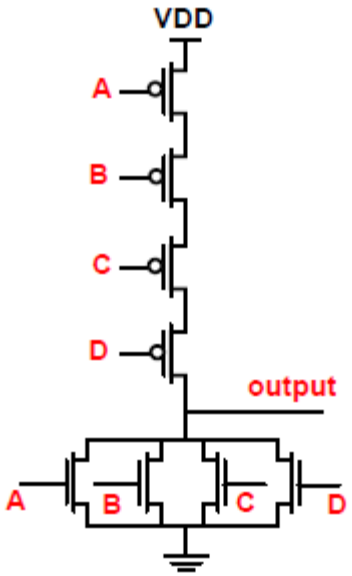
Examples

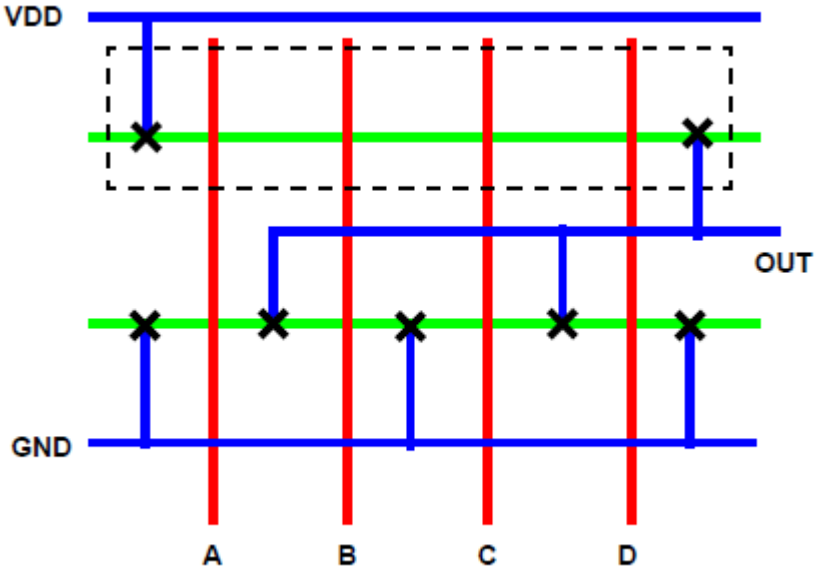
2-input NAND





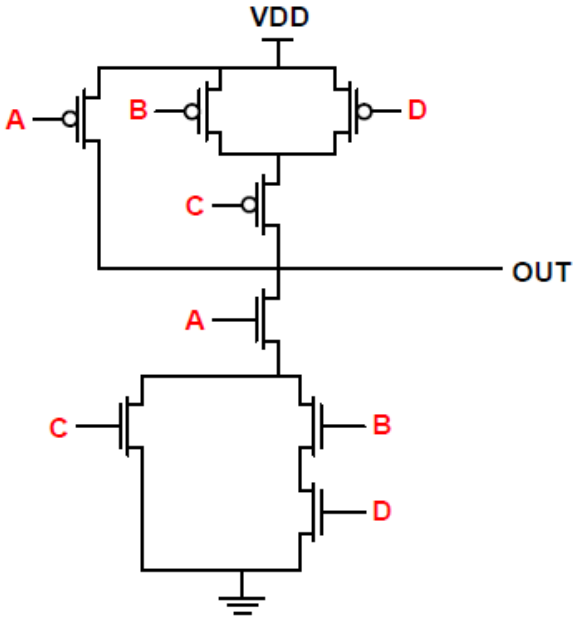
4-input NOR

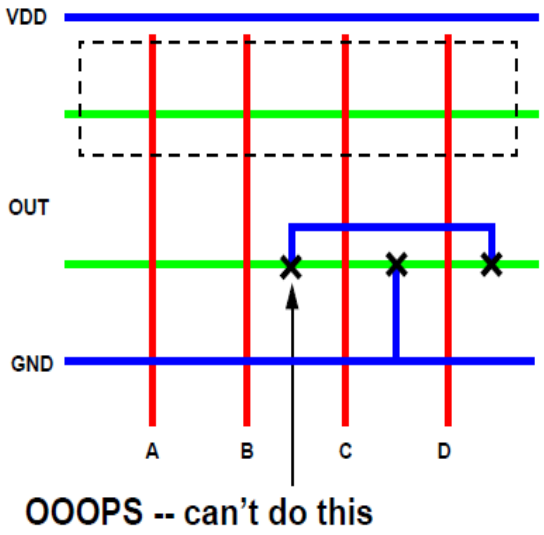




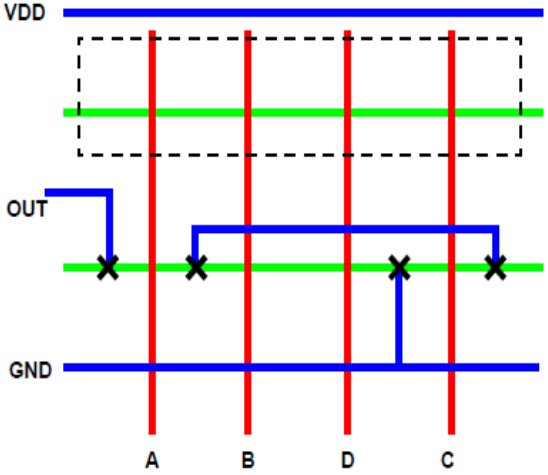
Examples

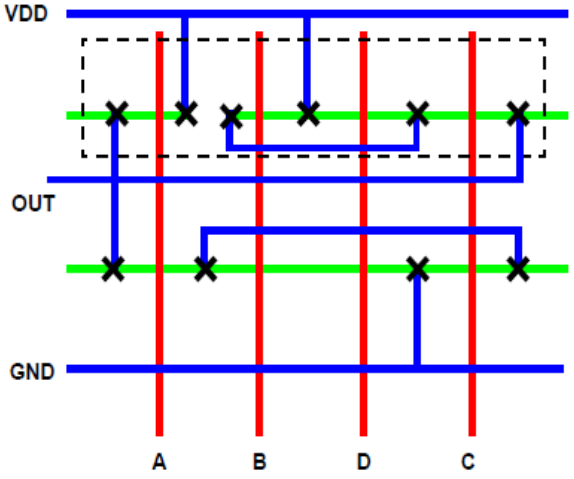
out = ~(A • (C + BD))





OOOPS -- can't do this





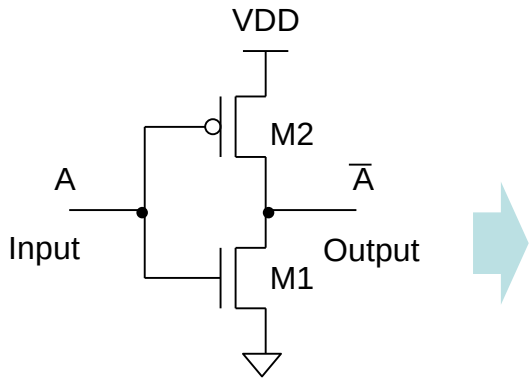
For PUN, either re-route poly or cut p-diff

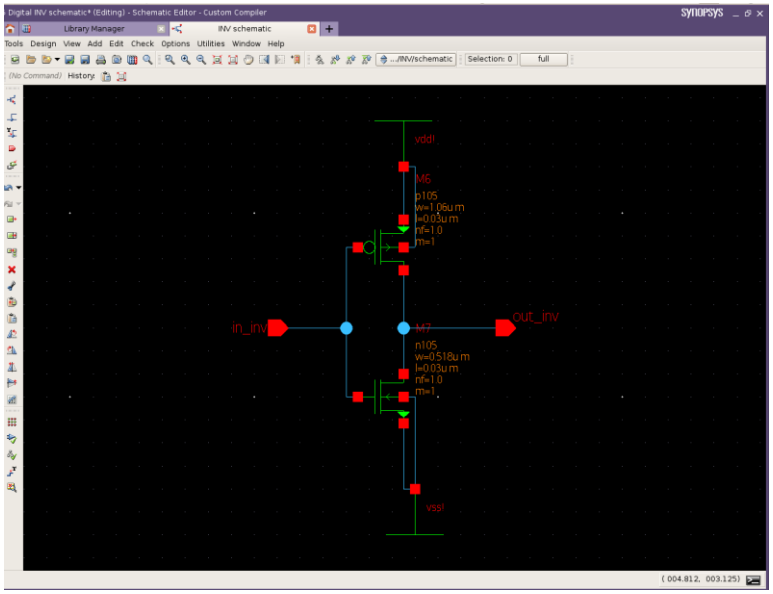
Schematic Design Goal

- The aim of schematic design is to create a circuit which works at operating conditions defined in specification and have the parameter values needed
- Schematic design
  - Structural synthesis
  - Parametric synthesis
  - Parametric optimization

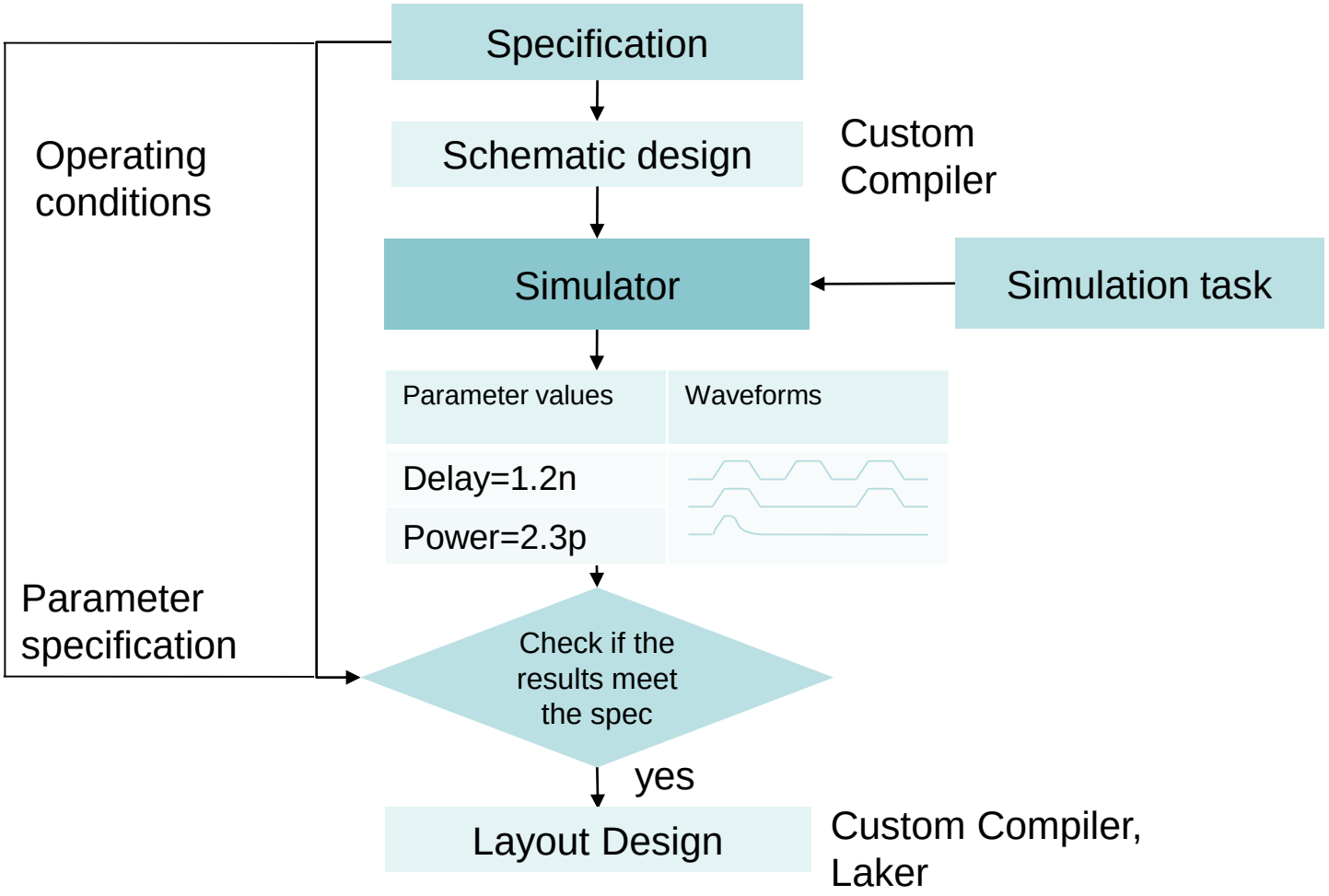
Circuit Selection

- Usually a known circuit structure is selected
- Design can find a convenient structure which is known to be good for the problem being solved



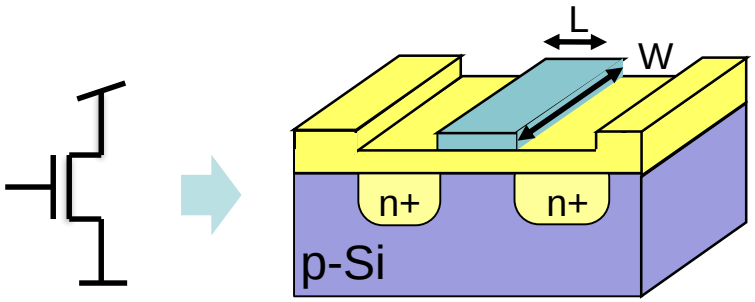


Schematic Design

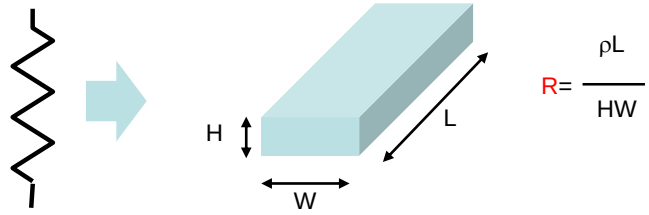


Parametric Optimization

- Each device has configurable parameters
- Schematic designer changes these parameters to get a circuit which meets the specification



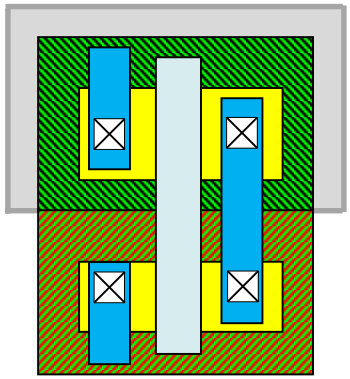
W - gate width, L - gate length

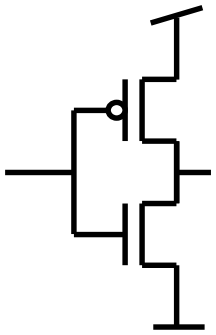


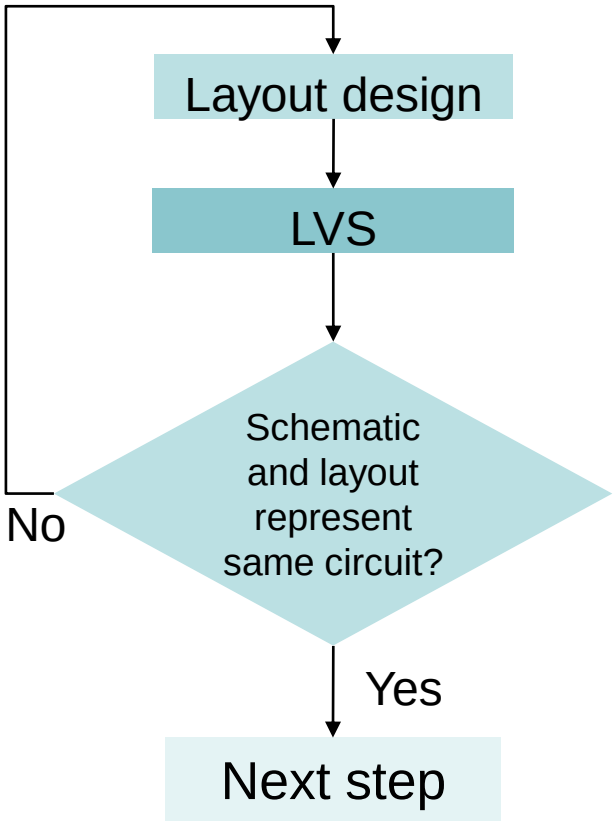
W - resistor width, L - resistor length

Transistor gate width and length, or resistor dimensions can be changed to change their electrical characteristics.

Layout Versus Schematic (LVS)

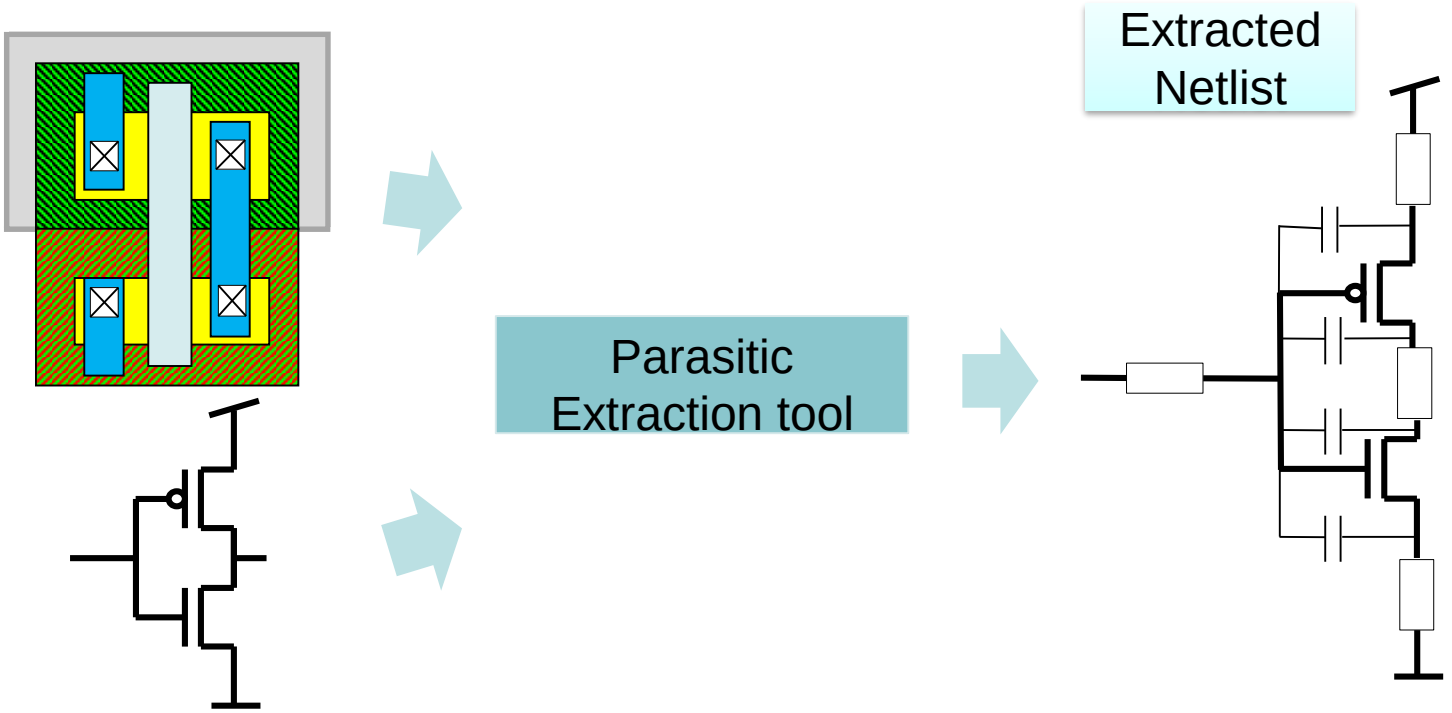






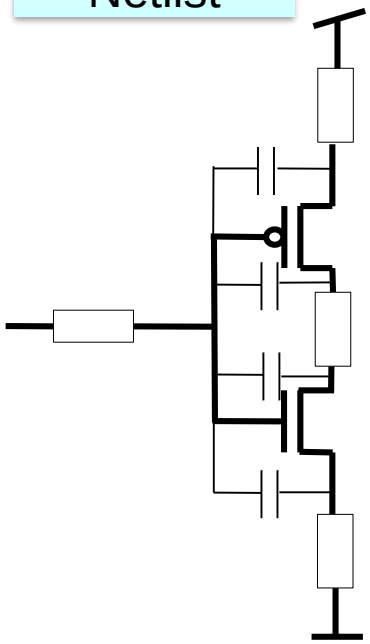
Layout Parasitic Extraction (LPE)

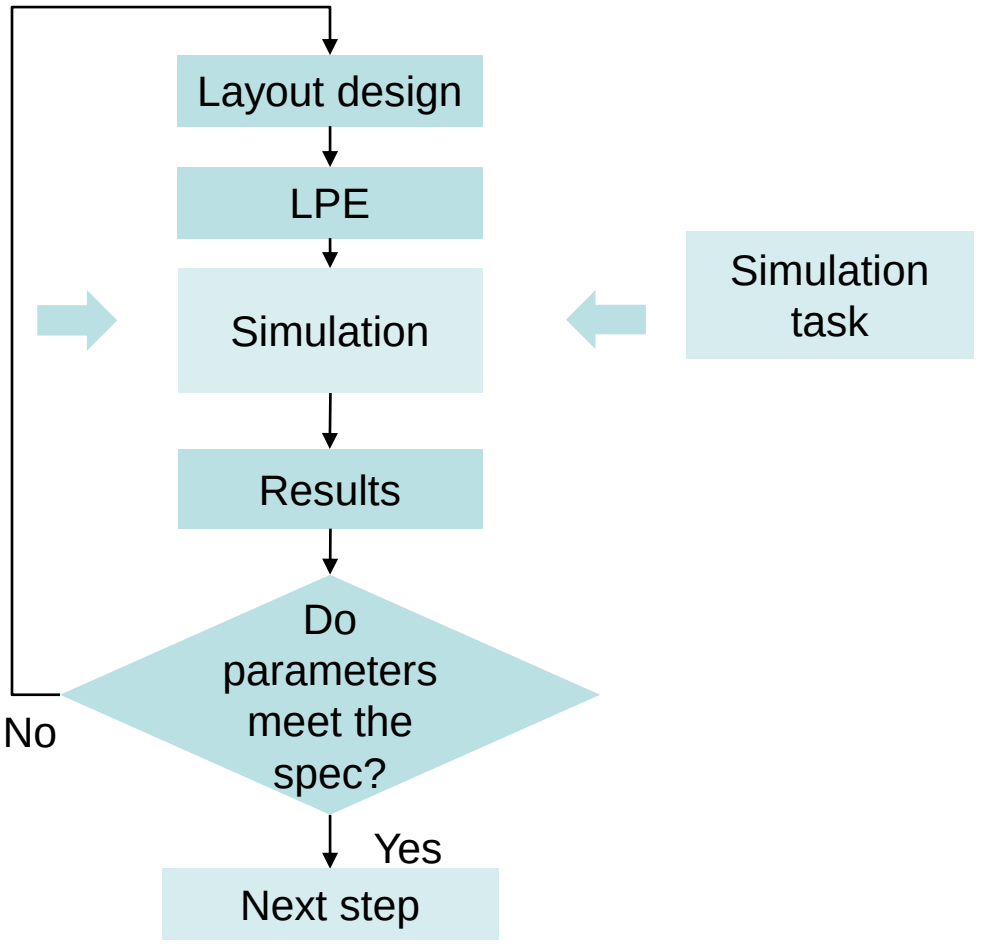
- There is a parasitic extractor tool which calculates parasitic devices present in layout adds them back to circuit



Simulation of Extracted Netlist

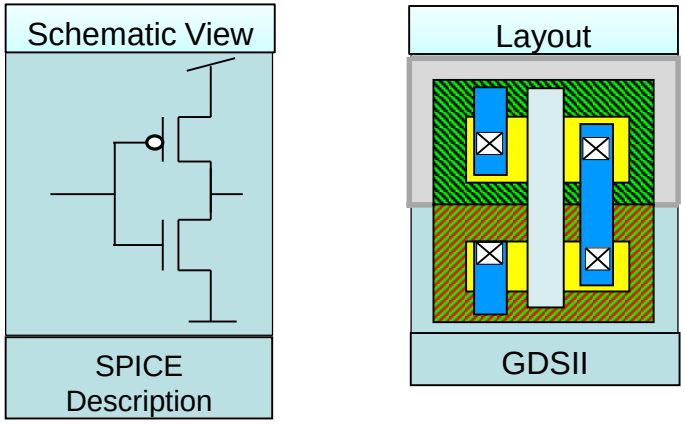
Extracted Netlist





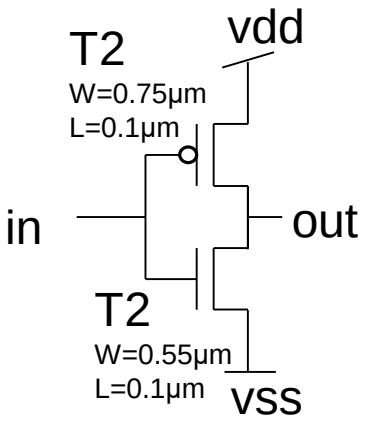
Deliverables

- Completed design is a set of files which represent different design views:
  - SPICE netlist format is used to deliver schematic view
  - GDSII binary format is used to deliver layout of the circuit



SPICE Description Example

- SPICE is a hardware description language (HDL) which enables to describe circuit at device level
- It has text-based format so is readable and can be easily modified

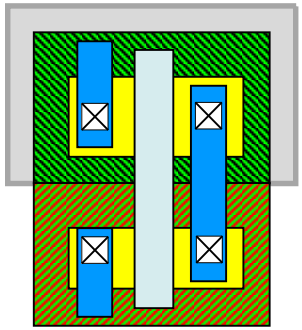


Inverter.sp

```
.subckt inverter
        mt1 out in vdd nmos l = 0.1u w = 0.75u
        mt2 out in vdd pmos l = 0.1u w = 0.55u
.ends
```

An Example of GDSII File

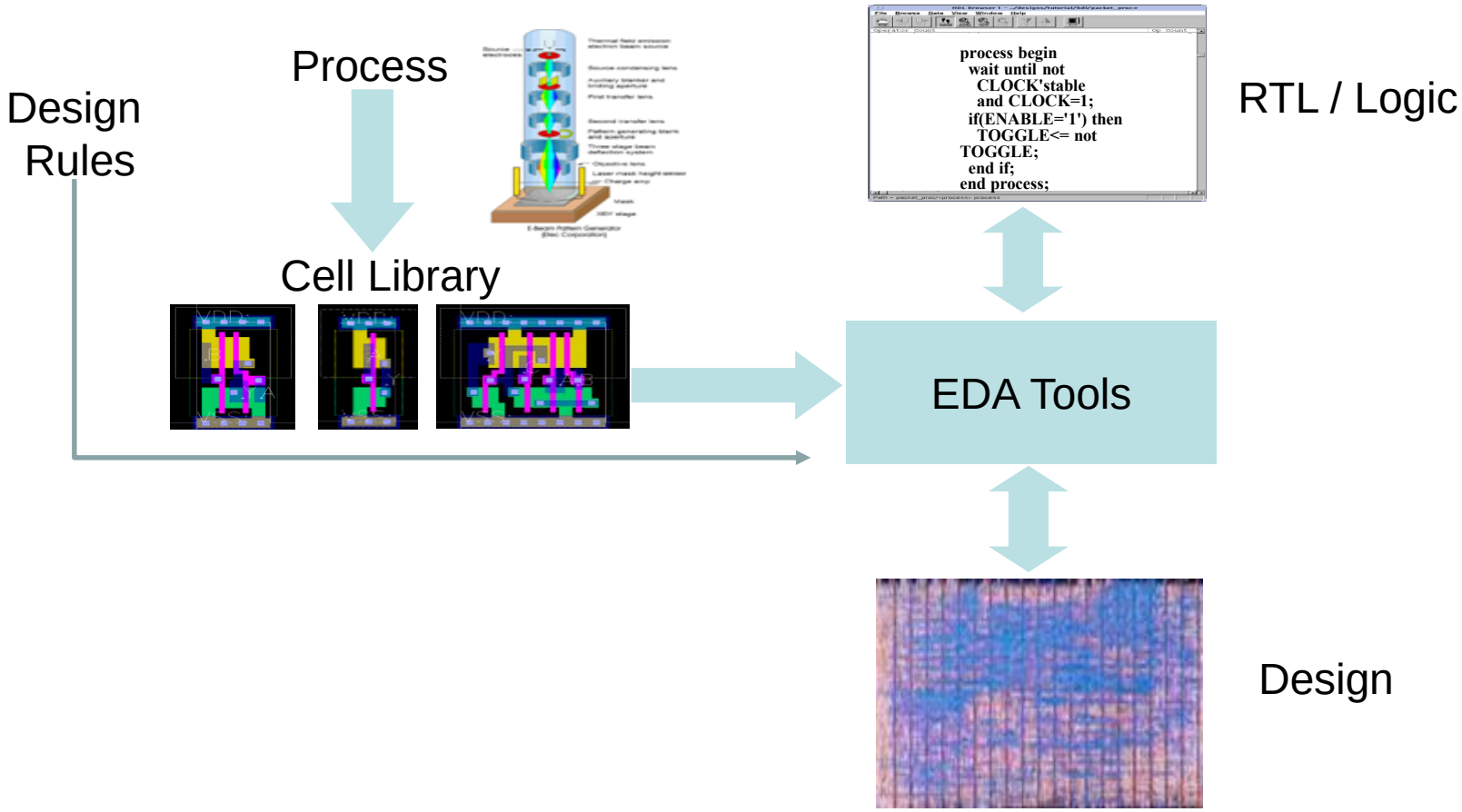
- GDSII is binary format, therefore it is not readable



Inverter.gds

```
02000200 60000201 1C000300 02000600 ..........`.... 000000
01000E00 02000200 60002500 01000E00 .....%.`........ 000010
42494C45 4C504D41 58450602 12002500 .%....EXAMPLELIB 000020
413E0503 14000300 02220600 59524152 RARY.."".......>A 000030
1C00545A 9BA02FB8 4439EFA7 C64B3789 .7K...9D./..ZT.. 00004
60000000 01000E00 02000200 60000205 ...`..........` 000050
58450606 0C001100 01000E00 02000200 .............EX 000060
0100020D 06000008 04000045 4C504D41 AMPLE..... 000070
0000F0D8 FFFF0310 2C000000 020E0600 .......,........ 000080
FFFF204E 00001027 0000204E 00001027 '...N ..'...N .. 000090
0000F0D8 FFFFF0D8 FFFFF0D8 FFFFF0D8 ..... 0000A0
00000004 04000007 04000011 04001027 '..... 0000B0
00000000 00000000 00000000 00000000 ..... 0000C0
```

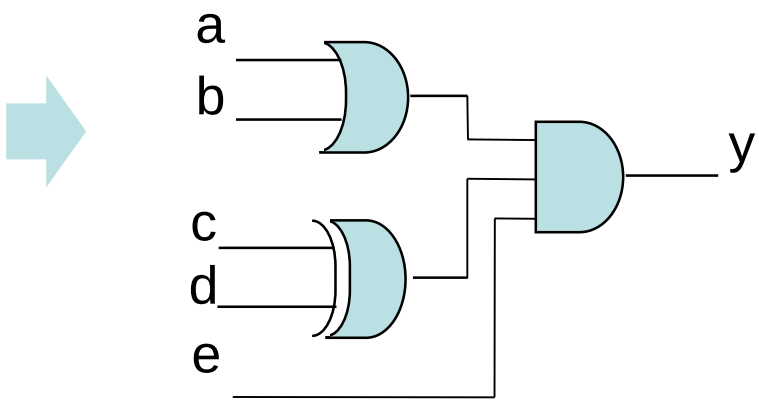
Cell Based Automated Design



Concept of Automated Design

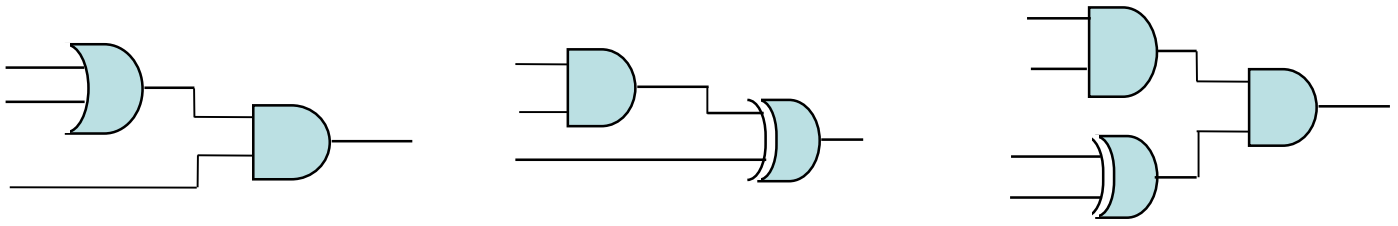
- Any digital function can be easily converted to a logic circuit. Synthesis tool uses this to automatically synthesize circuit from functional description.

$Y=(a+b)\&(c\oplus d)\&e$

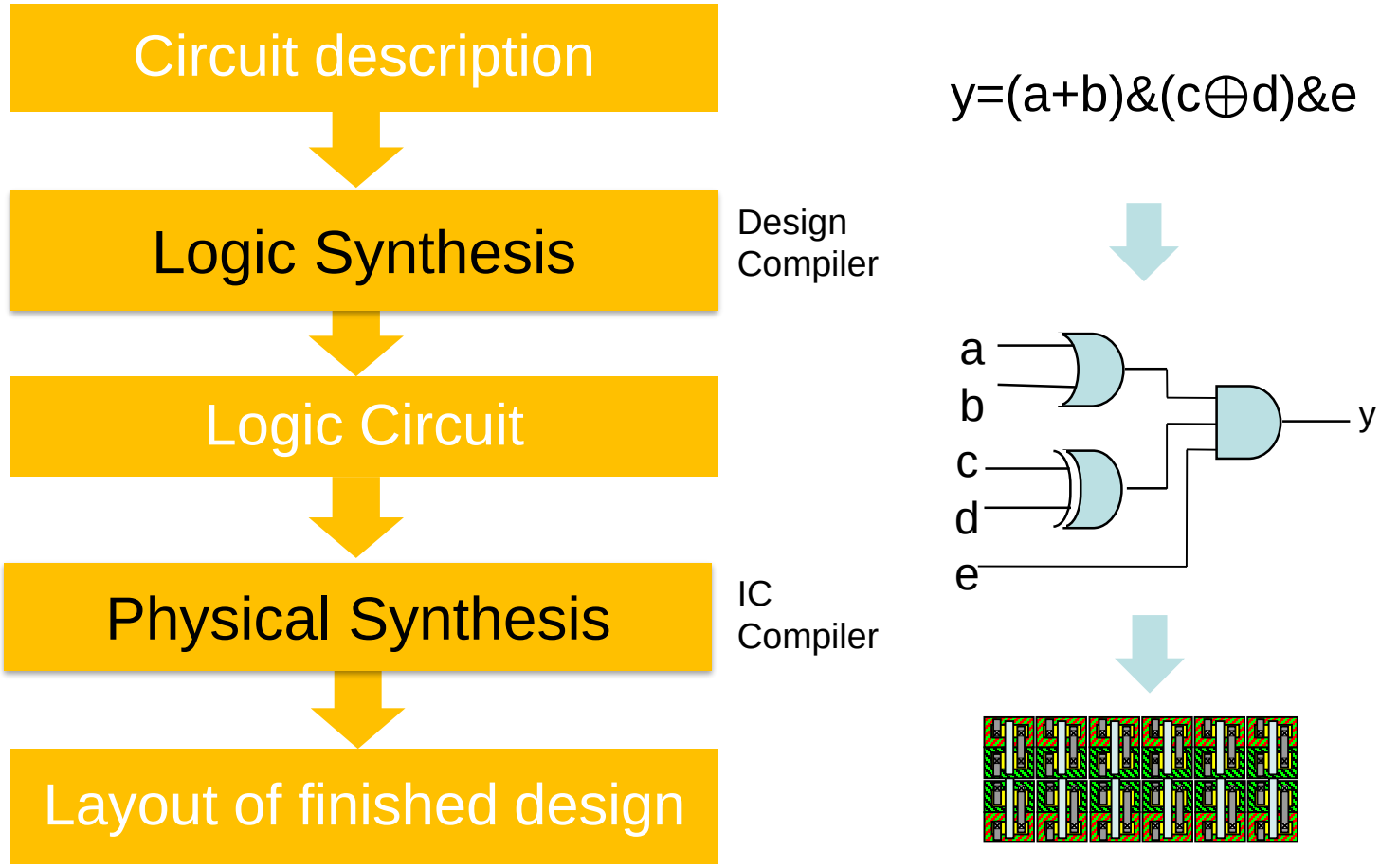


Concept of Automated Design (2)

- If primitive (standard) cells are previously designed, large number of various digital circuits can be built using these parts



Basic Steps of Synthesis

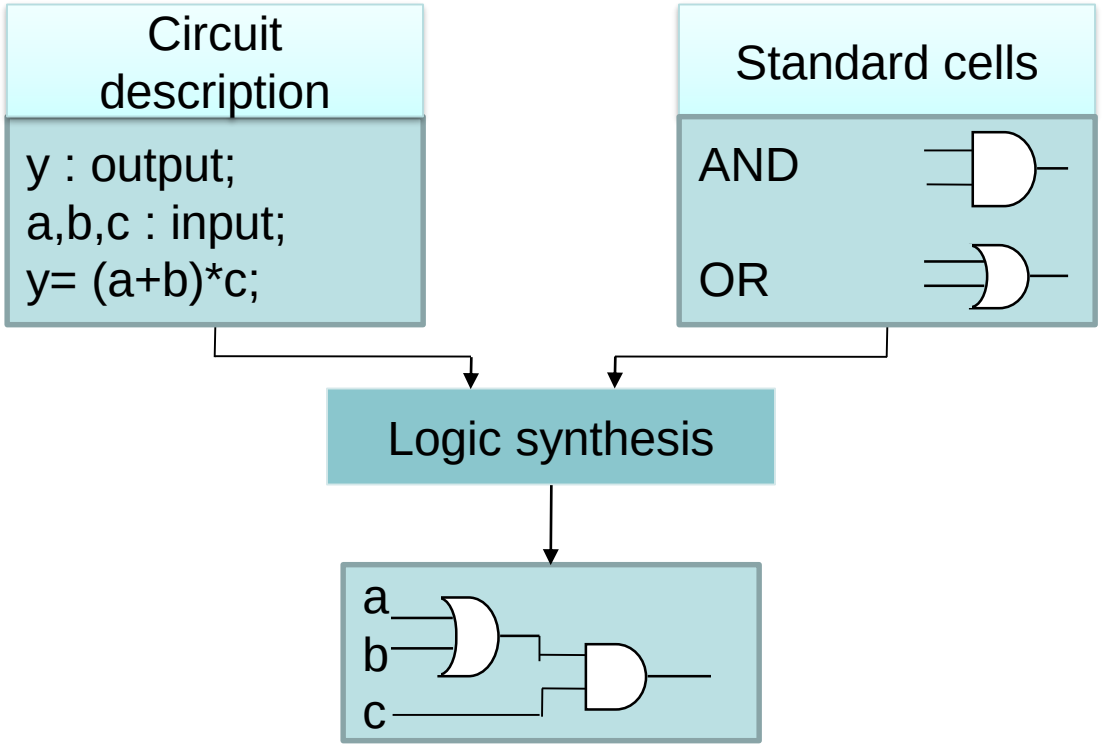


Digital IC Specification

- Description of Digital IC functionality
- With the help of Verilog or VHDL in the specification
- An example of specification line:
  - if incoming_call AND line_is_available then RING;
- The specifications of contemporary Digital IC can contain millions of lines, can be created by a collective of numerous participants within a few months

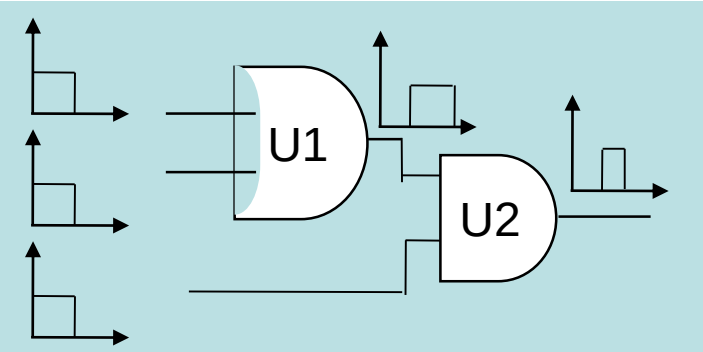
Logic Synthesis

- Logic synthesis is the process which produces logic circuit from circuit description

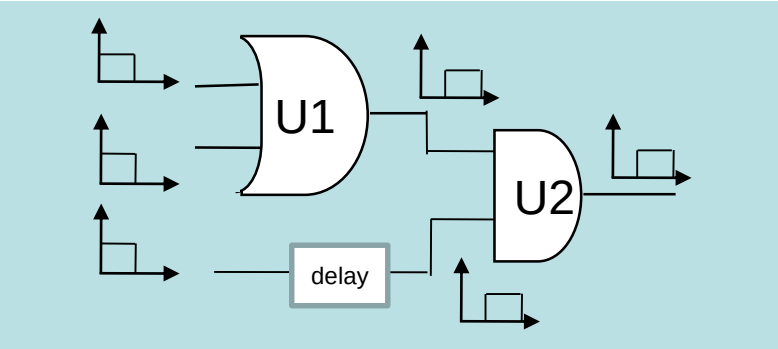


Logic Synthesis (2)

- Logic synthesis also optimizes the circuit.
  - The problem:
    - circuit simply created from function can possibly operate not as expected.



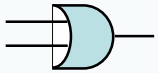
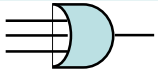
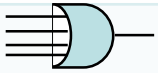
The delay of U1 element will affect final result



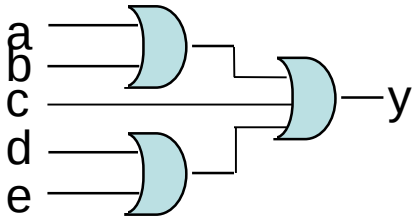
Additional elements should be added to the circuit to ensure correct operation

Main Optimization Trade-Offs

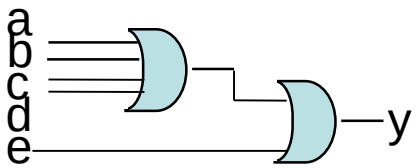
- Circuit design is a trade-off of timing, power and area
- Timing optimization
  - Goal: small delays
- Power optimization
  - Goal: low power consumption
- Area optimization
  - Goal: small area

Cell	Power
	2
	2.5
	3

Same function: Y=a+b+c+d



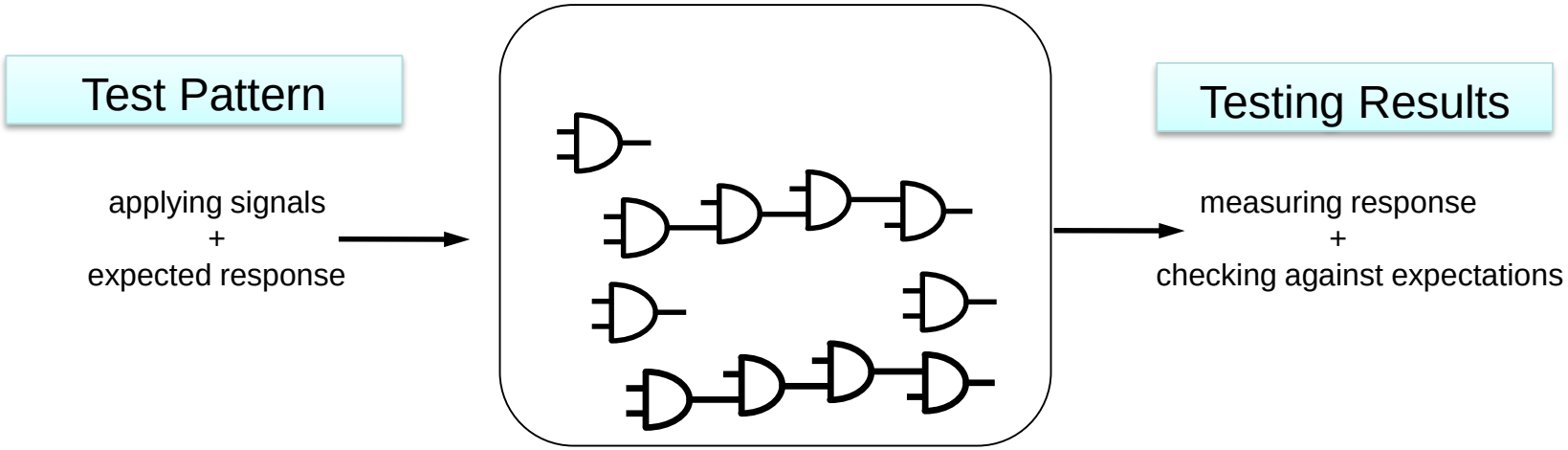
Total power:~6



Total power:~5

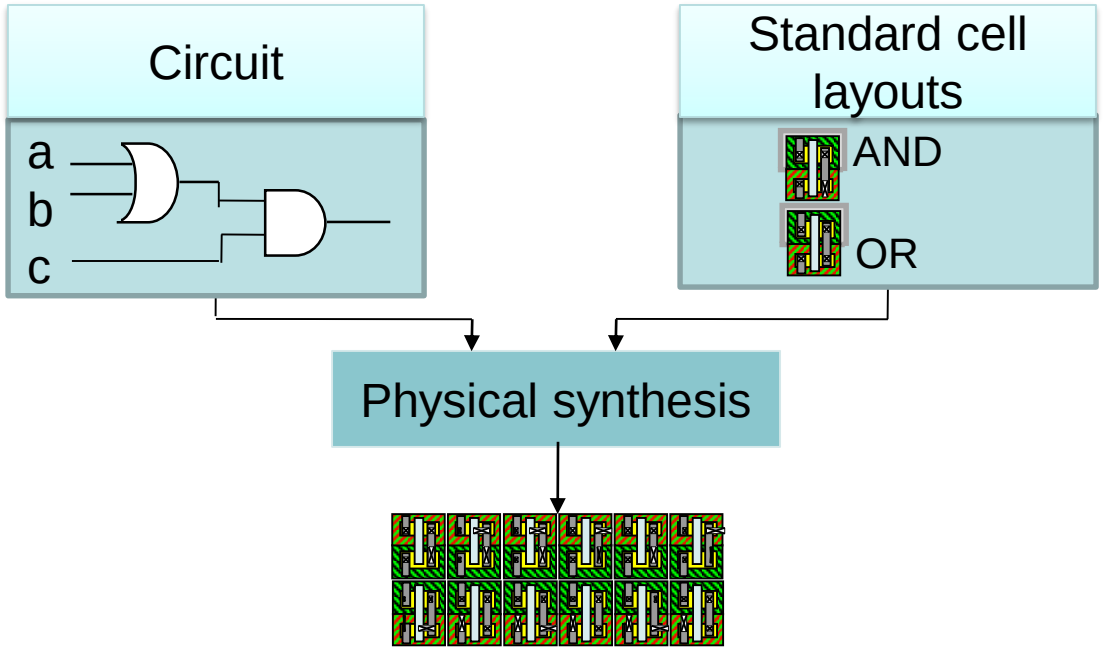
Test Creation

- Automatic test patterns (ATPG) are generated for synthesized circuit that can be used to test the design after fabrication.



Physical Synthesis

- Physical synthesis is the process that produces layout of logic circuit.

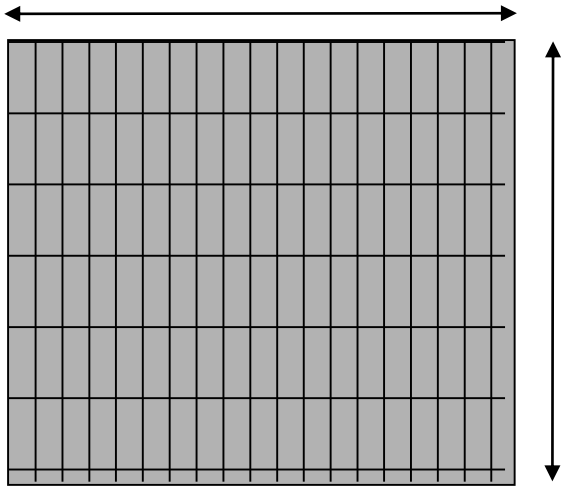


Physical Synthesis Steps

- Floorplanning
- Placement
- Routing

Floorplanning

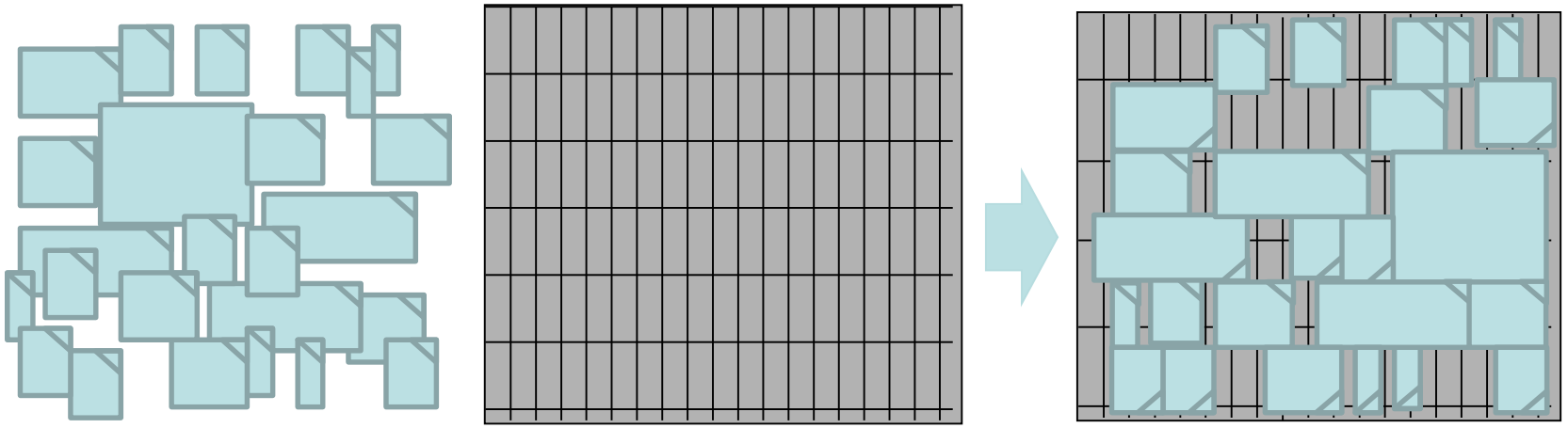
- During the floorplanning step the overall cell is defined, including: cell size, supply network, etc.



Floorplan

Placement

- Placement – exact placement of modules (modules can be standard cells, IPs)
  - The goal is to minimize the total area and interconnect length



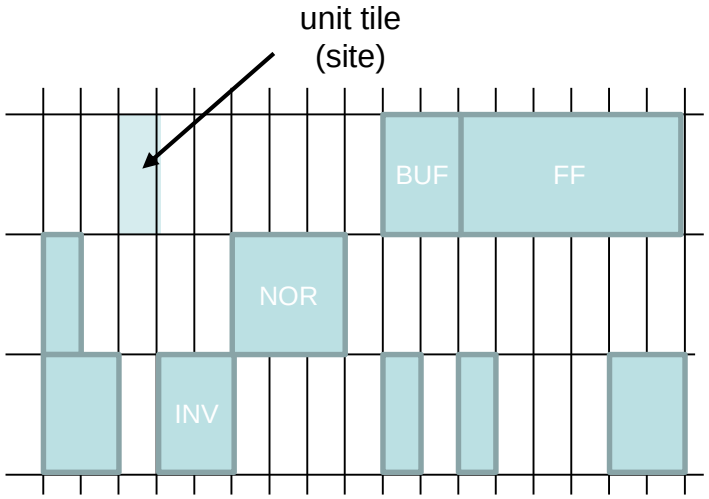
Cells from a circuit

Floorplan

Placed design

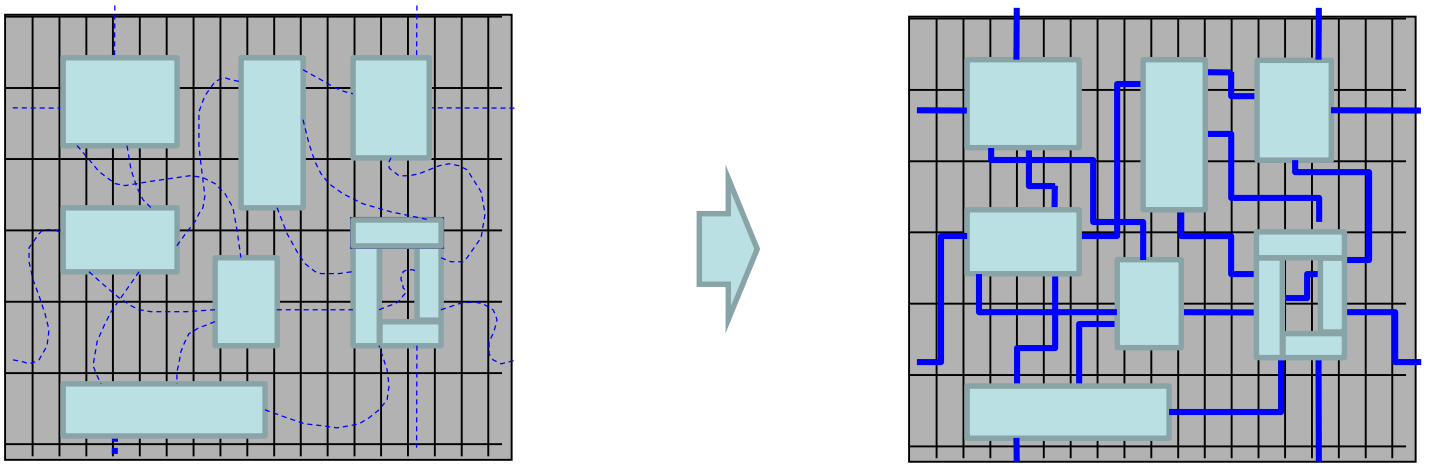
Unit Tile

- Placement uses grid in which cells are placed
- Floorplanning ‘unit tile’ cell to build this grid
  - Unit tile is defined by a library developer
  - All the cells in the library are designed to be multiple to unit tile



Routing

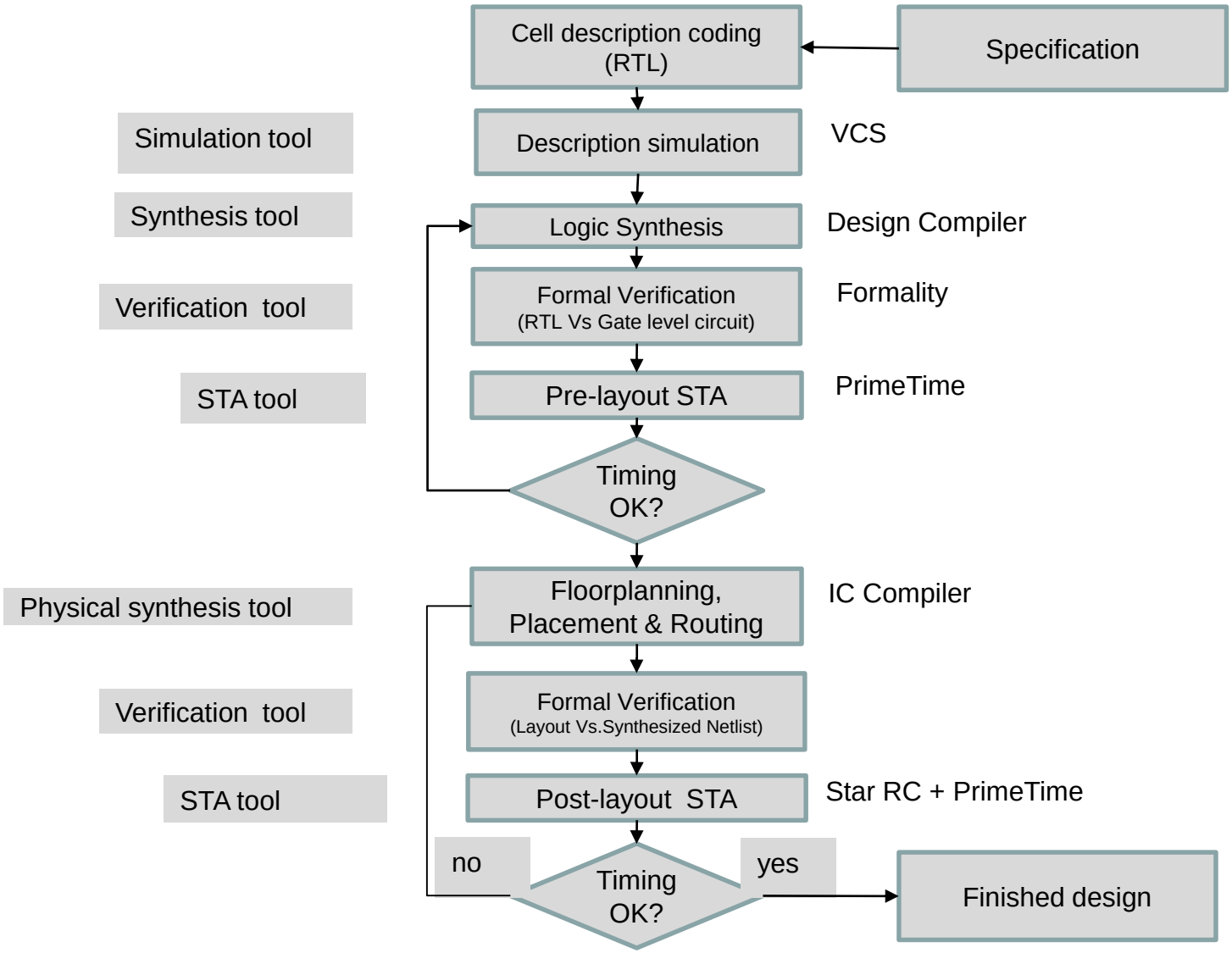
- Routing connects placed cells according to schematic
  - The goal is minimal impact of interconnects on circuit operation



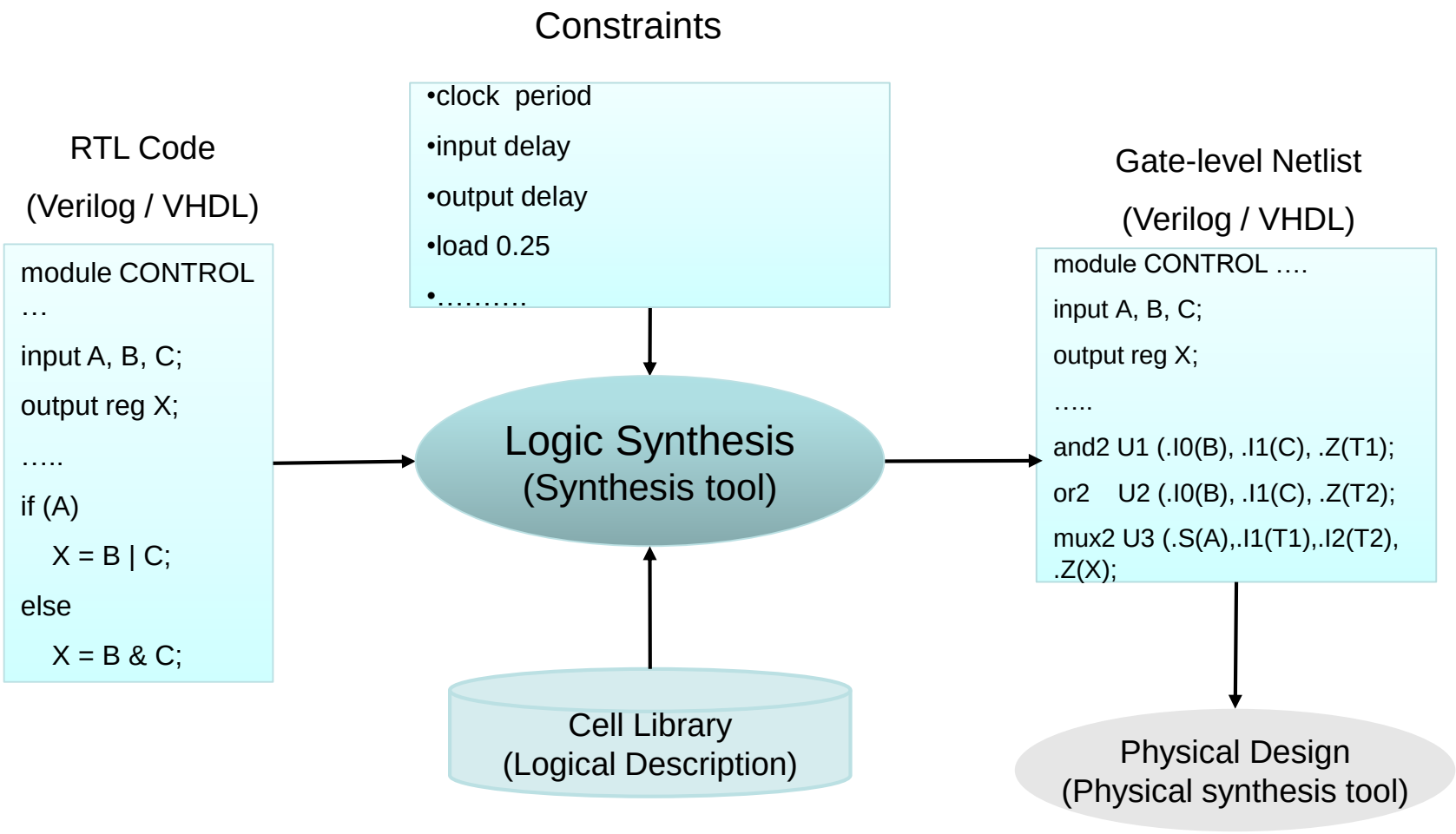
Placed design

Routed design

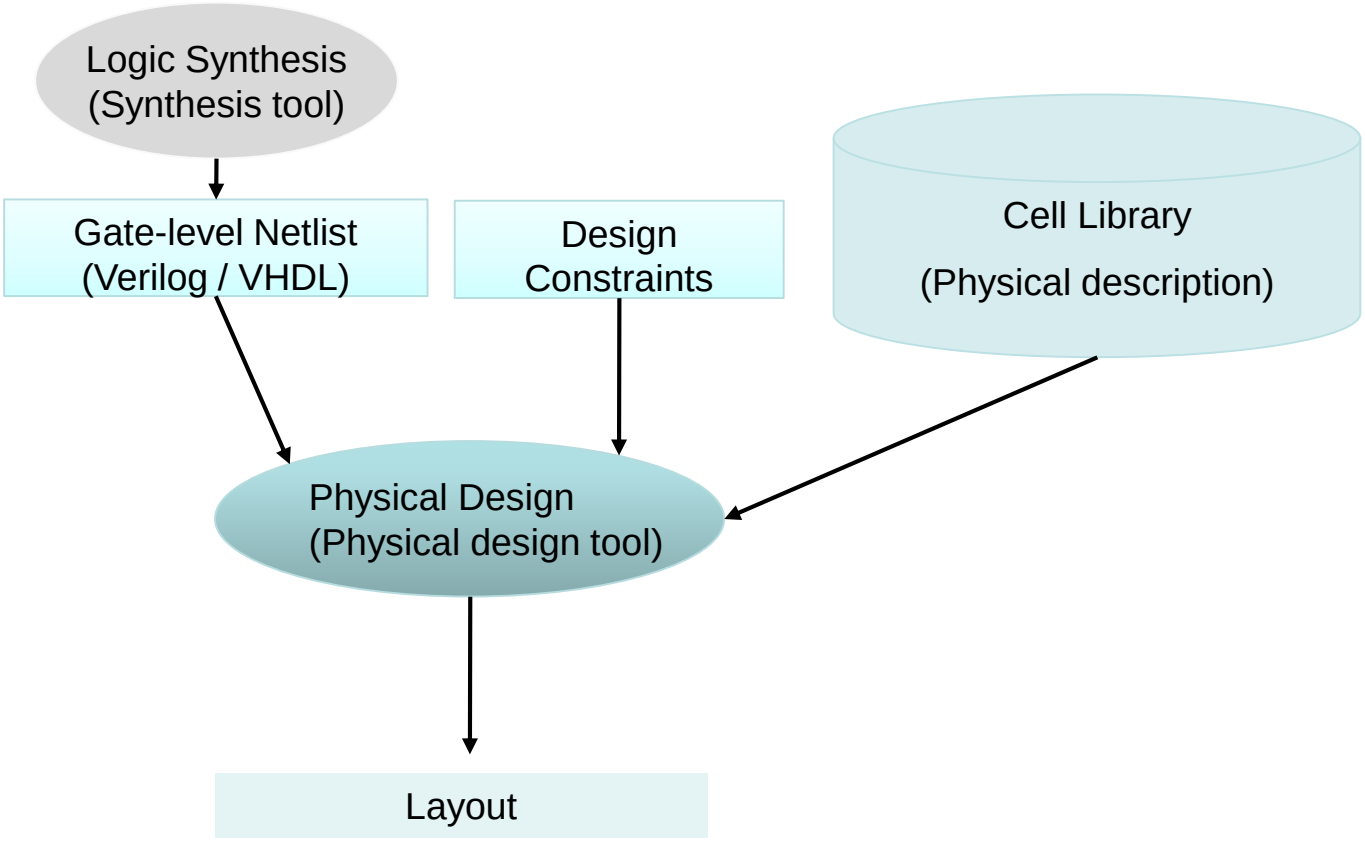
Digital IC Design Flow



Design Environment of Logic Synthesis



Design Environment of Physical Synthesis

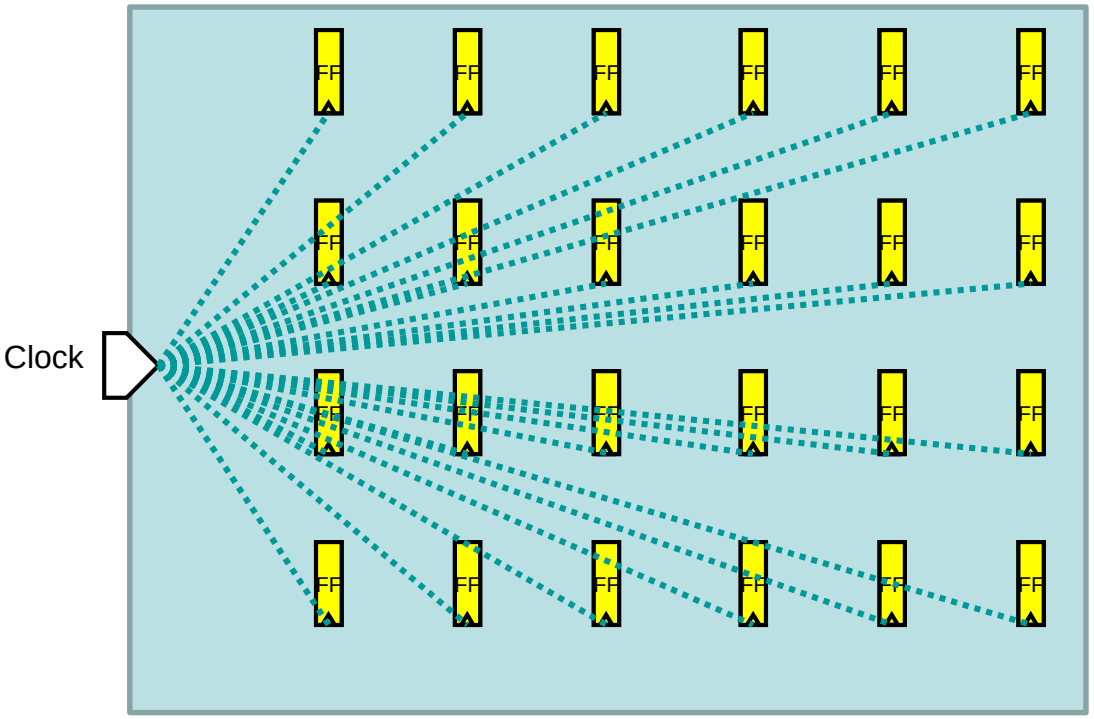


Circuit Optimization During Physical Synthesis

- Physical synthesis not only places and routes the cells of a circuit but also optimizes the cell as required by the designer
- Optimizations
  - Area
  - Interconnect length
  - Power density
  - Clock distribution
  - ...

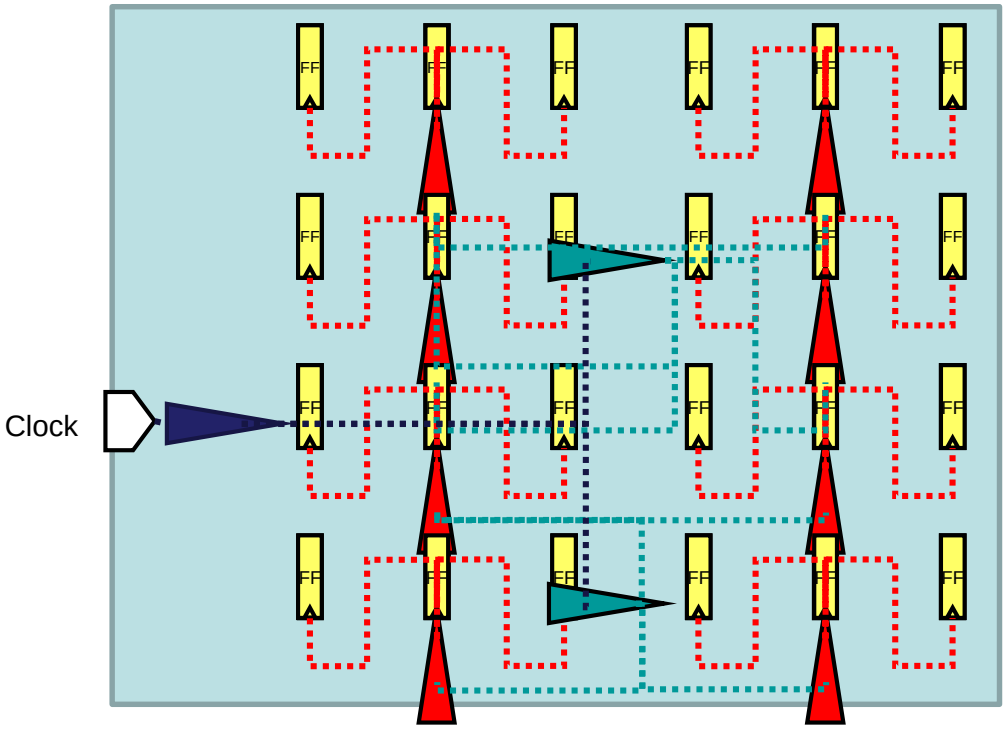
Physical Synthesis Circuit Optimization Example: Clock Delay Problems

- All clock pins are driven by a single clock source



Physical Synthesis Circuit Optimization Example: Clock Tree Synthesis

- A buffer tree is built to balance the loads and minimize the delays



Digital Standard Cell Library

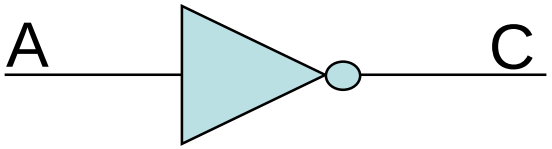
- Digital standard cell library (DSCL) is a set of cells which is used to design large ICs
- Cell number can be minimal, but the larger and more comprehensive is the set the more flexible will be synthesis, and the better will be the resulting circuit operation.

Digital Standard Cell Library: Gates

- Boolean logic is a set of functions defined on binary valued variables
  - Variable values may be defined in any of several ways: {1,0}, {True,False}, {On,Off}, {High,Low}, {2.5V,0V}, {VDD, VSS}
  - A logic function performs transformation on a set of boolean variables and constants

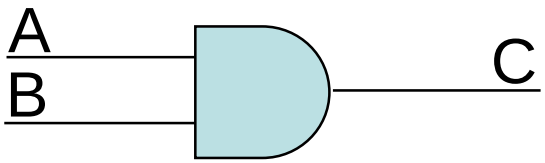
Basic Logic Gates

NOT or INV: NOT(0)=1, NOT(1)=0



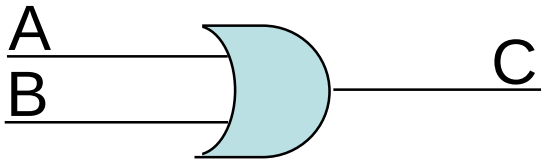
$C=\overline{A}$

AND: AND(1,1)=1, otherwise 0



C=AB

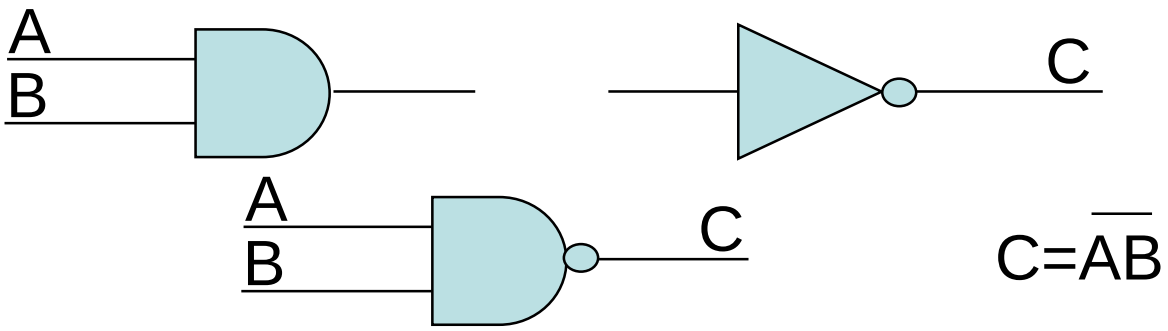
OR: OR(0,0)=0, otherwise 1



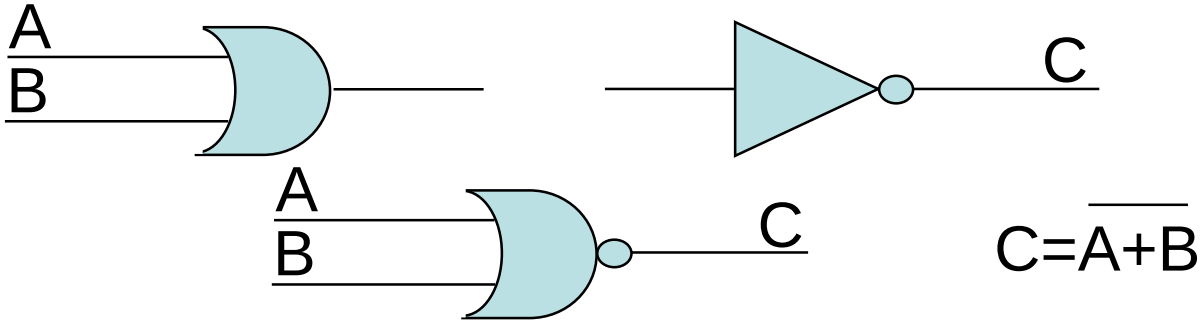
C=A+B

Basic Logic Gates (2)

NAND: NOT(AND)

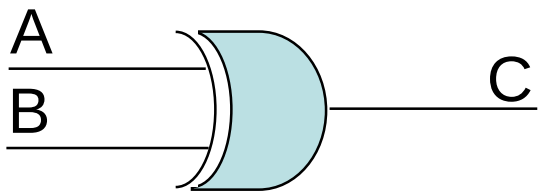


NAND: NOT(AND)



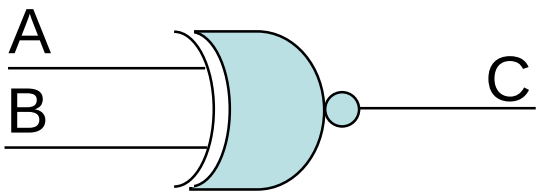
Basic Logic Gates (3)

XOR: XOR(0,1)=XOR(1,0)=1, otherwise 0



$C = A \oplus B$

XNOR: NOT(XOR) (Equivalent Gate)



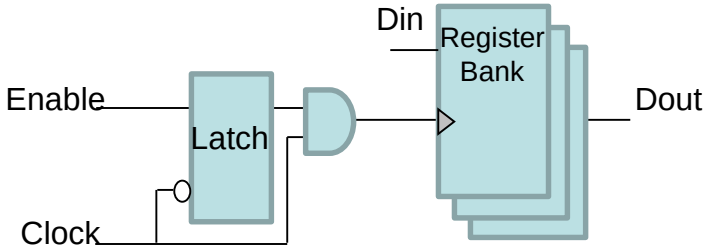
$C = \overline{A \oplus B}$

Standard Cell Specification Example

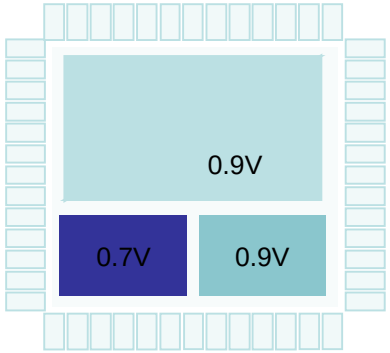
- The SAED_EDK32/28_CORE Digital Standard Cell Library will be built using SAED32/28nm 1P9M 1.05V/1.8V/2.5V design rules.
- The library will be created aimed at optimizing the main characteristics of designed integrated circuits by its help.
- The library will include typical miscellaneous combinational and sequential logic cells for different drive strengths.
- Besides, the library will contain all the cells which are required for different styles of low power (multi-voltage, multi-threshold) designs. Those are: Isolation Cells, Level Shifters, Retention Flip-Flops, Always-on Buffers and Power Gating Cells.
- The presence of all these cells will provide the support of integrated circuits design with different core voltages to minimize dynamic and leakage power.

Low Power Design Techniques Overview

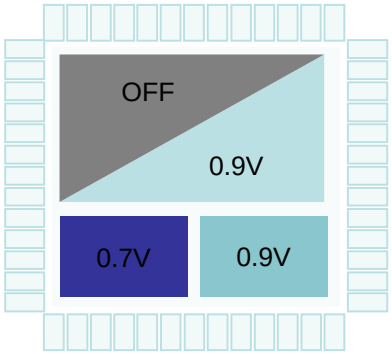
Clock Gating



Static Multi Voltage (MV)



MV with power gating



Standard Cell General Information

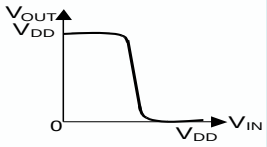
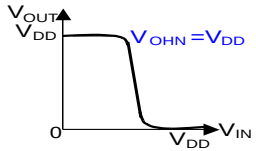
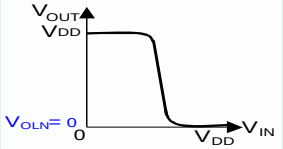
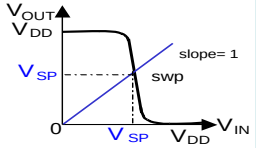
Symbol	State
L (“0”)	LOW Logic Level
H (“1”)	HIGH Logic Level
Z	High-impedance State
LH (“0”→”1”)	LOW to HIGH Transition
HL (“1”→”0”)	HIGH to LOW Transition
X	Either HIGH or LOW Logic Level

Operating Conditions

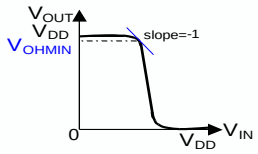
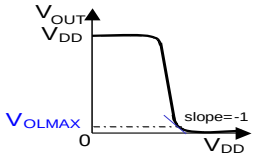
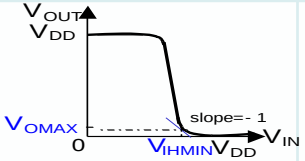
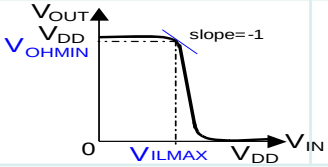
- SAED_EDK32/28_CORE Digital Standard Cell Library specification is given for 1.2V operation. The used process technology will be SAED32/28nm 1P9M 1.05V/1.8V/2.5V, but only the 1P1M option will be used.

Parameter	Min	Typ	Max	Units
Power Supply (VDD) range	0.945	1.05	1.061	V
Operating Temperature range	-40	+25	+125	°C
Operating Frequency (F)	-	300	-	MHz

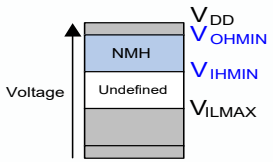
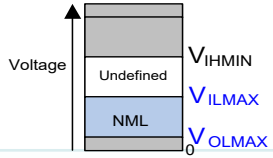
DC Parameters and Measurement Conditions of Digital Cells

N	Parameter	Unit	Symbol	Figure	Definition
1.	Voltage Transfer Characteristic	-	VTC		DC functional dependence between input and output voltages.
2.	Output high level voltage (nominal)	V	$V_{OHN}=V_{DD}$		Output high voltage at nominal condition, usually equals to V_{DD}
3.	Output low level voltage (nominal)	V	$V_{OLN}=0$ ($V_{OLN}=V_{SS}$)		Output low voltage at nominal condition, usually $V_{OLN}=0$
4.	Switching point voltage	V	V_{SP}		Point on VTC where VOUT =VIN

DC Parameters and Measurement Conditions of Digital Cells (2)

N	Parameter	Unit	Symbol	Figure	Definition
5.	Output high level minimum voltage	V	V_{OHMIN}		Highest output voltage at slope= -1
6.	Output low level maximum voltage	V	V_{OLMAX}		Lowest output voltage at slope= -1
7.	Input minimum high voltage	V	V_{IHMIN}		Highest input voltage at slope = -1
8.	Input maximum low voltage	V	V_{ILMAX}		Lowest input voltage at slope = -1

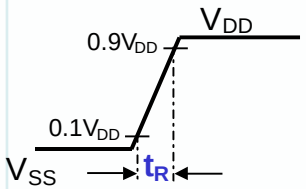
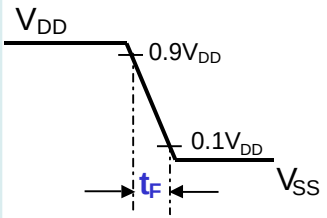
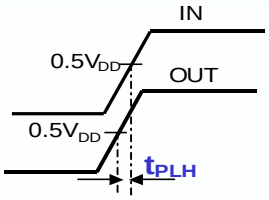
DC Parameters and Measurement Conditions of Digital Cells (3)

N	Parameter	Unit	Symbol	Figure	Definition
9.	High state noise Margin	V	$NMH = V_{OHMIN} - V_{IHMIN}$		The maximum input noise voltage which does not change the output state when its value is subtracted from the input high level voltage
10.	Low state noise margin	V	$NML = V_{ILMAX} - V_{OLMAX}$		The maximum input noise voltage which does not change the output state when added to the input low level voltage
11.	Static leakage current consumption at output on high state	uA	I_{LEAKH}	None	The current consumed when the output is high
12.	Leakage power consumption (dissipation) at output	pW	$P_{LEAKL} = V_{DD} \times I_{LEAK}$	None	The power consumed when the output is high

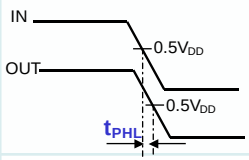
DC Parameters and Measurement Conditions of Digital Cells (4)

N	Parameter	Unit	Symbol	Figure	Definition
13.	Leakage power consumption (dissipation) at output on high state	pW	$P_{LEAKL}=V_{DD}$ x I_{LEAKH}	None	The power consumed when the output is high
14.	Leakage power consumption (dissipation) at output on high state	pW	PLEAKL=VDD x ILEAKL	None	The power consumed when the output is low

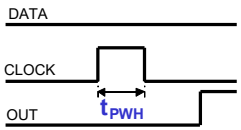
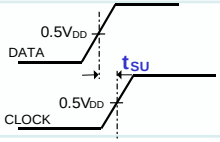
AC Parameters and Measurement Conditions of Digital Cells

N	Parameter	Unit	Symbol	Figure	Definition
1.	Rise transition time	ns	t_R		The time it takes a driving pin to make a transition from kV_{DD} to $(1-k)V_{DD}$ value. Usually k=0.1 (also possible k=0.2, 0.3, etc)
2.	Fall transition time	ns	t_F		The time it takes a driving pin to make a transition from $(1-k)V_{DD}$ to kV_{DD} value. Usually k=0.1 (also possible k=0.2, 0.3, etc)
3.	Propagation delay low-to-high (Rise propagation)	ns	t_{PLH} (t_{PR})		Time difference between the input signal crossing a $0.5V_{DD}$ and the output signal crossing its $0.5V_{DD}$ when the output signal is changing from low to high

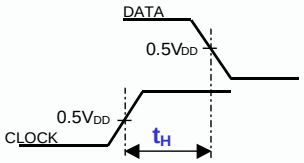
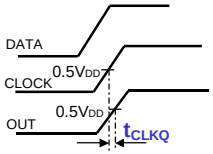
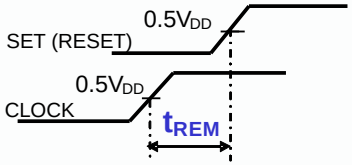
AC Parameters and Measurement Conditions of Digital Cells (2)

N	Parameter	Unit	Symbol	Figure	Definition
4.	Propagation delay high-to-low (Fall propagation)	ns	t_{PHL} (t_{PF})		Time difference between the input signal crossing a 0.5V_{DD} and the output signal crossing its 0.5V_{DD} when the output signal is changing from high to low
5.	Average supply current	uA	$I_{V_{DD}AVG} = \int_0^T I_{V_{DD}}(t)dt$		The power supply current average value for a period (T)
6.	Supply peak current	uA	$I_{VDDPEAK}$= =max(I_{VDD}(t)) t∈[0;T]		The peak value of power supply current within one period (T)
7.	Dynamic power dissipation	pW	P_{DISDYN}= =I_{VDDAVG} x V_{DD}		The average power consumed from the power supply
8.	Power-delay product	nJ	PD=PDISDYN x x max (tPHL,tPLH)		The product of consumed power and the largest propagation delay

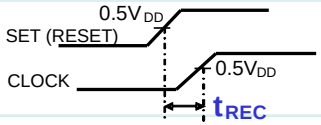
AC Parameters and Measurement Conditions of Digital Cells (3)

N	Parameter	Unit	Symbol	Figure	Definition
9.	Energy-delay product	nJs	ED=PD x x max(t_{PHL},t_{PLH})		The product of PD and the largest propagation delay
10.	Switching fall power	nJ	P_{SWF} = =(C_{LOAD}+C_{OUTF}) x x V_{DD}^2/2		The energy dissipated on a fall transition. (C_{OUTF} is the output fall capacitance)
11.	Minimum clock pulse (only for flip-flops or latches)	ns	t_{PWH} (t_{PWL})		The time interval during which the clock signal is high or low, so that it ensures proper operation of a flip-flop or a latch
12.	Setup time (only for flip-flops or latches)	ns	t_{PWH} (t_{PWL})		The time interval during which the clock signal is high or low, so that it ensures proper operation of a flip-flop or a latch
13.	Setup time (only for flip-flops or latches)	ns	t_{SU}		The minimum period in which the input data to a flip-flop or a latch must be stable before the active edge of the clock occurs

AC Parameters and Measurement Conditions of Digital Cells (4)

N	Parameter	Unit	Symbol	Figure	Definition
14.	Hold time (only for flip-flops or latches)	ns	t_H		The minimum period in which the input data to a flip-flop or a latch must remain stable after the active edge of the clock has occurred
15.	Clock-to-output time (only for flip-flops or latches)	ns	t_{CLKQ}		The amount of time that takes the output signal to change after clock's active edge is applied
16.	Removal time (only for flip-flops or latches with asynchronous Set or Reset).	ns	t_{REM}		The minimum time in which the asynchronous Set or Reset pin to a flip-flop or latch must remain enabled after the active edge of the clock has occurred

AC Parameters and Measurement Conditions of Digital Cells (5)

N	Parameter	Unit	Symbol	Figure	Definition
17.	Recovery time (only for flip-flops and latches with asynchronous Set or Reset)	ns	t_{REC}		The minimum time in which Set or Reset must be held stable after being deasserted before next active edge of the clock occurs
18.	From high to Z-state entry time, (only for tri-state output cells)	ns	t_{HZ}		The amount of time that takes the output to change from high to Z-state after control signal is applied
19.	From low to Z-state entry time, (only for tri-state output cells)	ns	t_{LZ}		The amount of time that takes the output to change from low to Z-state after control signal is applied
20.	From Z to high-state exit time (only for tri-state output cells)	ns	t_{ZH}		The amount of time that takes the output to change from Z to high-state after control signal is applied
21.	From Z to low-state exit time (only for tri-state output cells)	ns	t_{ZL}		The amount of time that takes the output to change from Z to low-state after control signal is applied
22.	Input pin capacitance	pF	C_{IN}		Defines the load of an output pin
23.	Maximum capacitance	pF	C_{MAX}		Defines the maximum total capacitive load that an output pin can drive

Standard Cell List Example

No	Cell Description	Drive Strength	Cell Name
	Inverters. Buffers		
1.	Inverters	1x C_{sl}	INVX1
2.	Inverters	2x C_{sl}	INVX2
3.	Non-inverting Buffer	4x C_{sl}	NBUFFX4
4.	Non-inverting Buffer	8x C_{sl}	NBUFFX8
5.	Tri-state inverting inverting Buffer W/ Low active enable	2x C_{sl}	TIBUFFL1X2
6.	Tri-state inverting inverting Buffer W/ Low active enable	3x C_{sl}	TIBUFF1X3

Standard Cell List Example (2)

No	Cell Description	Drive Strength	Cell Name
	Logic Gates		
7.	AND 2-input	2x C_{sl}	AND2X2
8.	AND 2-input	3x C_{sl}	AND2X3
9.	NAND 2-input	2x C_{sl}	NAND2X2
10.	NAND 2-input	3x C_{sl}	NAND2X3
11.	OR 2-input	3x C_{sl}	OR2X3
12.	OR 2-input	4x C_{sl}	OR2X4
13.	NOR 2-input	2x C_{sl}	NOR2X2
14.	NOR 2-input	3x C_{sl}	NOR2X3

Standard Cell List Example (3)

No	Cell Description	Drive Strength	Cell Name
	Complex Logic Gates		
15.	AND OR 2/1	2x C_{sl}	AO21X2
16.	AND OR 2/1	3x C_{sl}	AO21X3
17.	AND-OR-Invert 2/1	2x C_{sl}	AOI21X2
18.	AND-OR Invert 2/1	3x C_{sl}	AOI21X3
19.	OR AND 2/2	3x C_{sl}	OA22X3
20.	OR AND 2/2/1	2x C_{sl}	OA221X2
21.	OR AND Invert 2/2/1	2x C_{sl}	OAI221X2
22.	OR AND Invert 2/2/1	3x C_{sl}	OAI221X3
23.	OR AND Invert 2/2/2	3x C_{sl}	OAI222X2

Standard Cell List Example (4)

No	Cell Description	Drive Strength	Cell Name
	Multiplexers		
24.	Multiplexer 2 to 1	2x C_{sl}	MUX21X2
25.	Multiplexer 2 to 1	3x C_{sl}	MUX21X3
26.	Multiplexer 4 to 1	2x C_{sl}	MU421X2
27.	Multiplexer 4 to 1	3x C_{sl}	MUX41X3
	Decoders		
28.	Decoder 2 to 4	2x C_{sl}	DEC24X2
29.	Decoder 2 to 4	3x C_{sl}	DEC24X3
	Adders and Subtractors		
30.	Half Adder 1 bit	2x C_{sl}	HADDX2
31.	Half Adder 1 bit	3x C_{sl}	HADDX2
32.	Full Adder 1 bit	2x C_{sl}	FADDX2

Standard Cell List Example (5)

No	Cell Description	Drive Strength	Cell Name
	D Flip-Flops		
32.	Pos edge D Flip-Flop	2x C_{sl}	DFFX2
33.	Pos edge D Flip-Flop	4x C_{sl}	DFFX4
34.	Pos edge D Flip-Flop, w/ Async low active Set	2x C_{sl}	DFFASBX2
35.	Pos edge D Flip-Flop, w/ Async low active Set	4x C_{sl}	DFFASBX4
36.	Neg edge D Flip-Flop	2x C_{sl}	DFFNX2
37.	Neg edge D Flip-Flop	4x C_{sl}	DFFNX
38.	Neg edge D Flip-Flop, w/ Async low active Set	2x C_{sl}	DFFNASBX2
39.	Neg edge D Flip-Flop, w/ Async low active Set	4x C_{sl}	DFFNASBX4
	Scan D Flip-Flops		
40.	Scan Pos edge D Flip-Flop	2x C_{sl}	SDFFX2
41.	Scan Pos edge D Flip-Flop	4x C_{sl}	SDFFX4
42.	Scan Pos edge D Flip-Flop w/ Async low active Set	2x C_{sl}	SDFFASBX2
43.	Scan Pos edge D Flip-Flop w/ Async low active Set	4x C_{sl}	SDFFASBX4
44	Scan Neg edge D Flip-Flop	2x C_{sl}	SDFFNX2
45.	Scan Neg edge D Flip-Flop	4x C_{sl}	SDFFNX4
46.	Scan Neg edge D Flip-Flop w/ Async low active Set	2x C_{sl}	SDFFNASBX2
47.	Scan Neg edge D Flip-Flop w/ Async low active Set	4x C_{sl}	SDFFNASBX4

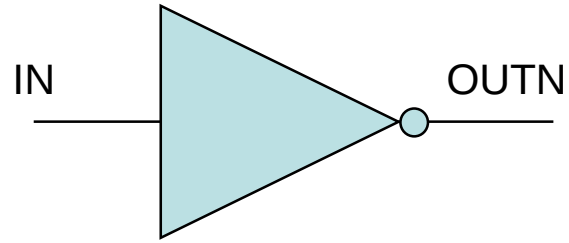
Standard Cell List Example (6)

No	Cell Description	Drive Strength	Cell Name
	Latches		
48.	RS NAND Latch	2x C_{sl}	LNANDX2
49.	RS NAND Latch	4x C_{sl}	LNANDX4
	Delay Lines		
50.	Non-inverting Delay Line, 0.5 ns	2x C_{sl}	DELLN1D1
51.	Non-inverting Delay Line, 0.75 ns	2x C_{sl}	DELLN1D2
	Pass Gates		
52.	Pass Gate	2x C_{sl}	PGX2
53.	Pass Gate	3x C_{sl}	PGX3
Bi-directional Switches			
54.	Bi-directional Switch w/ High-active Enable	2x C_{sl}	BSHEX2
55.	Bi-directional Switch w/ High-active Enable	3x C_{sl}	BSHEX3
	Isolation Cells		
56.	Hold 1 Isolation Cell(Logic AND)	2x C_{sl}	ISOLANDX2
57.	Hold 1 Isolation Cell(Logic AND)	4x C_{sl}	ISOLANDX4
	Level Shifters		
58.	Low to High Level Shifter	2x C_{sl}	LSUPX2
59.	High to High Level Shifter	4x C_{sl}	LSDNX4
60.	High to High Level Shifter	16x C_{sl}	LSDNX16

Standard Cell List Example (7)

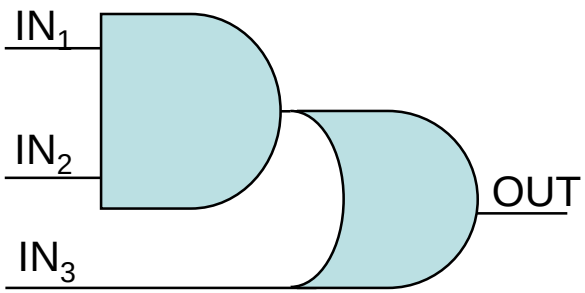
No	Cell Description	Drive Strength	Cell Name
	Retention Flip-Flops		
61.	Pos edge Retention D Flip-Flop	2x C_{sl}	DRFFX2
62.	Pos edge Retention D Flip-Flop	4x C_{sl}	DRFFX4
63.	Pos edge Retention D Flip-Flop, w/ Async low active Reset	2x C_{sl}	DRFFARBX2
64.	Pos edge Retention D Flip-Flop, w/ Async low active Reset	4x C_{sl}	DRFFARBX4
	Power Gating Cells		
65.	Header Cell	4x C_{sl}	HEADX4
66.	Footer Cell	4x C_{sl}	FOOTX4
	Always on Cells		
67.	Always on Non-inverting Buffer	2x C_{sl}	AOBUFX2
68.	Always on Non-inverting Buffer	4x C_{sl}	AOBUFX4
69.	Always on Pos edge D Flip-Flop, w/ Async low active Reset	2x C_{sl}	AODFFASBX2
70.	Always on Pos edge D Flip-Flop, w/ Async low active Reset	4x C_{sl}	AODFFASBX2
	Additional Cells		
71.	Tie High		TIEH
72.	Tie Low		TIEL
73.	Antenna Diode		ANTENNA
74.	Decoupling Capacitance		DCAP
	Fillers		
75.	Filler Cell 1grid width		FILL1
76.	Filler Cell 2grid width		FILL2
77.	Double height filler Cell 1grid width		DHFILL1
78.	Double height filler Cell 4grid width		DHFILL4

Standard Cell Example: Inverter



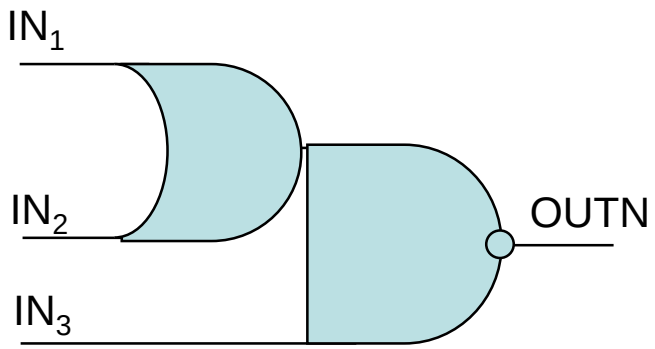
IN	OUTN
0	1
1	0

Standard Cell Example: AND-OR



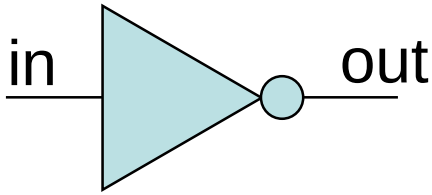
IN_1	IN_2	IN_3	OUT
1	1	X	1
X	X	1	1
0	X	0	0
X	0	0	0

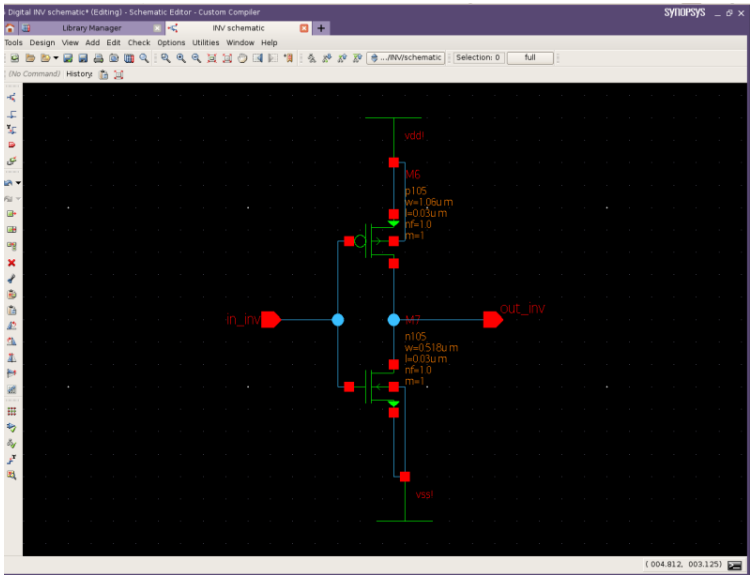
Standard Cell Example: OR-AND-INVERT

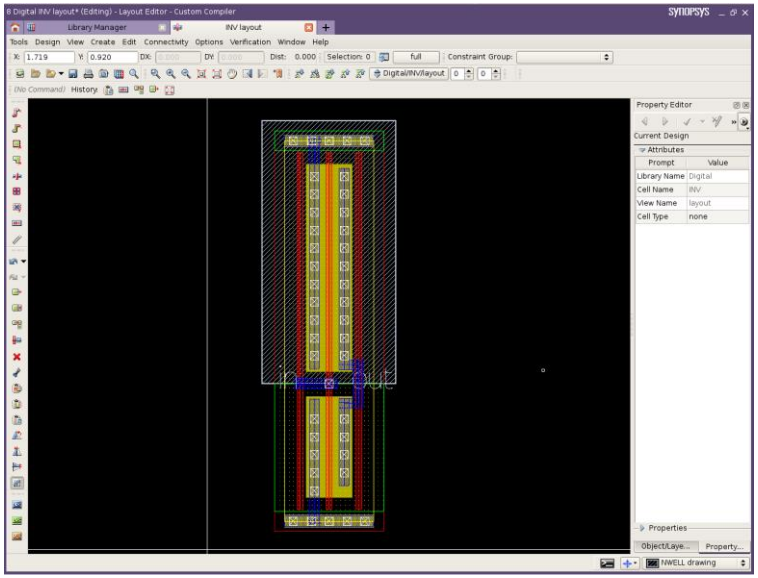


IN_1	IN_2	IN_3	OUTN
0	0	X	1
X	X	0	1
1	X	1	0
X	1	1	0

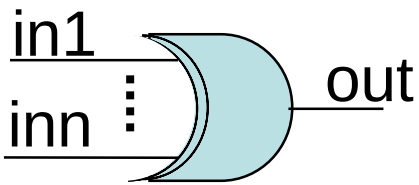
Standard Cell Example: Inverter

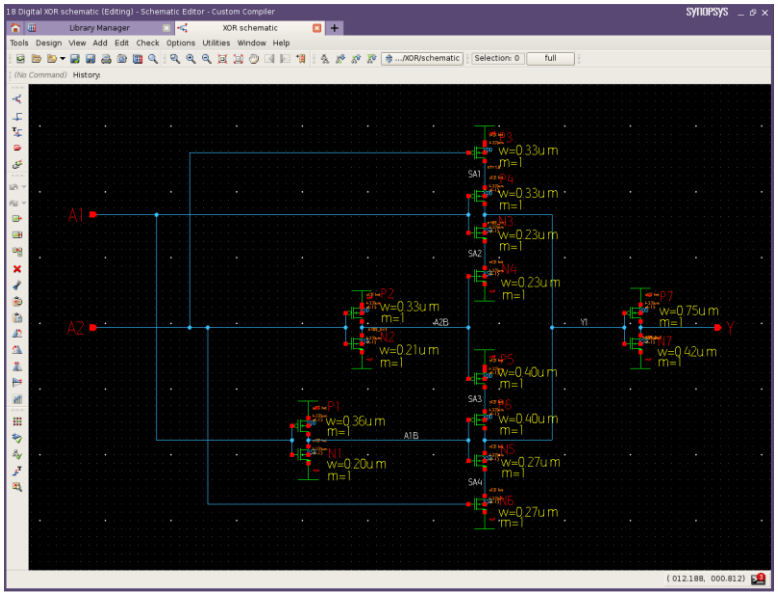


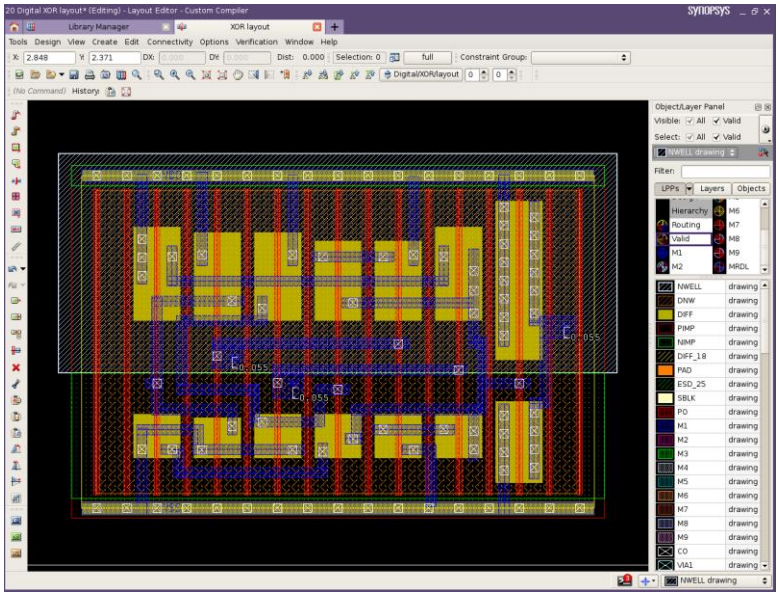




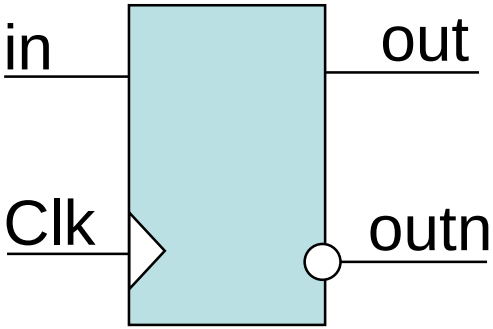
Standard Cell Example: XOR

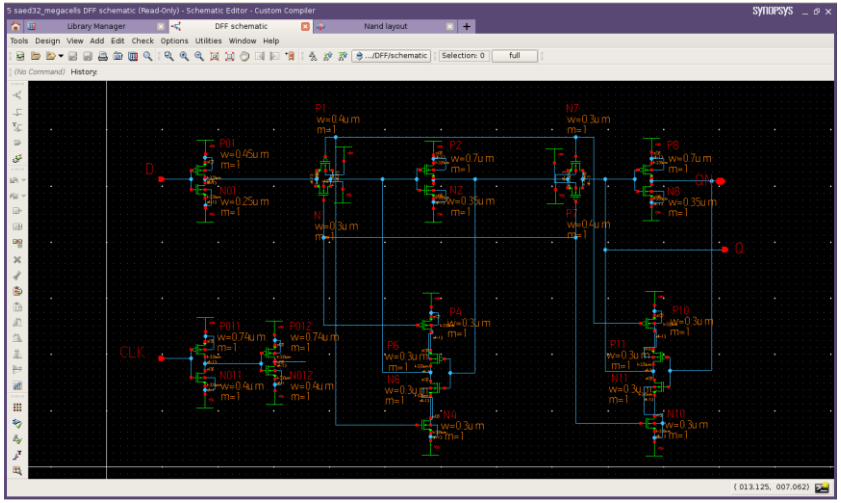


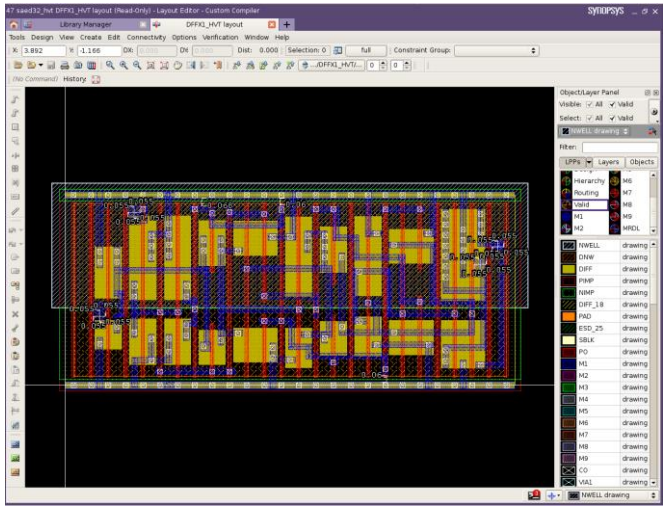




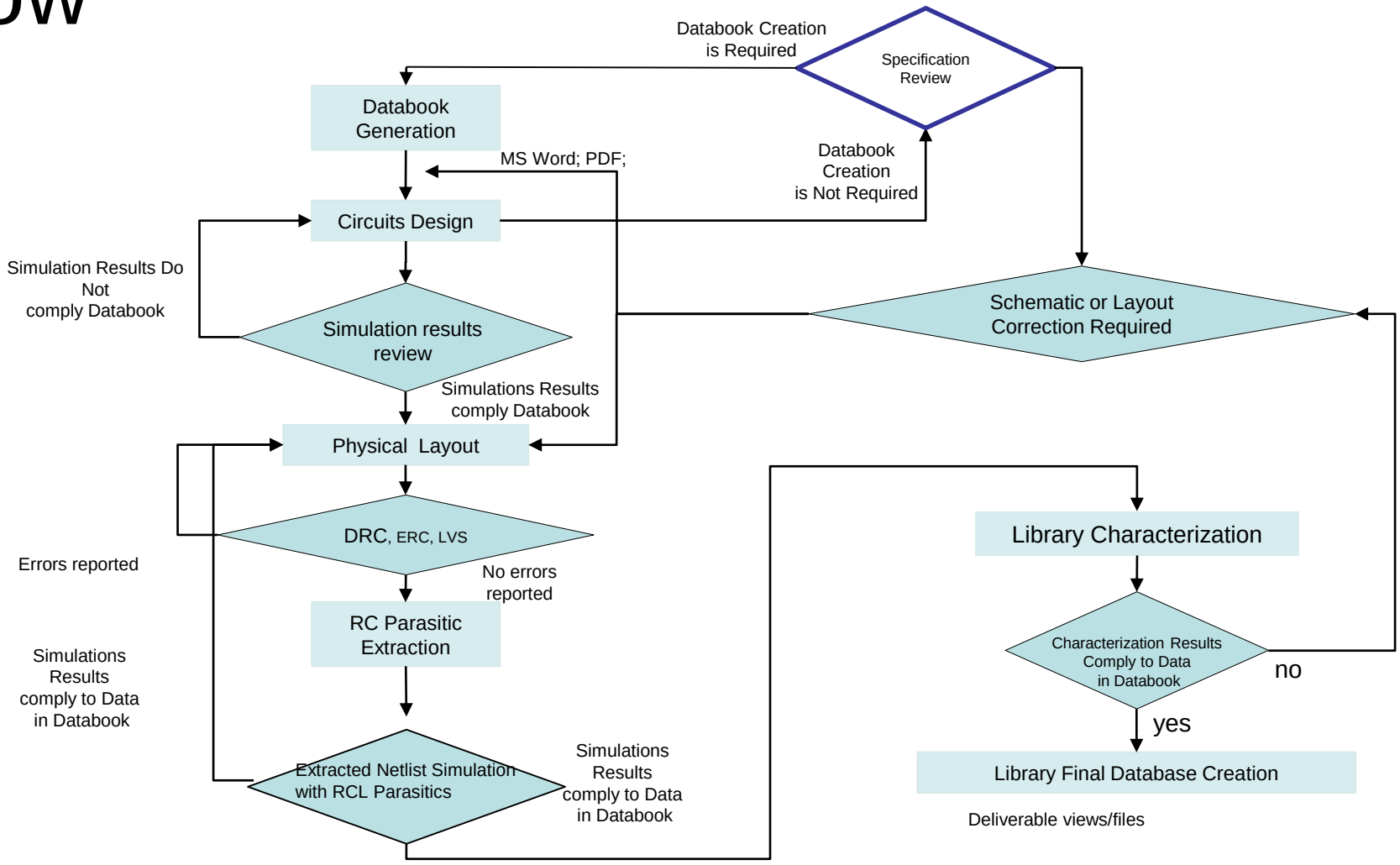
Standard Cell Example: D Flip-Flop







Digital Standard Cell Library Design Flow

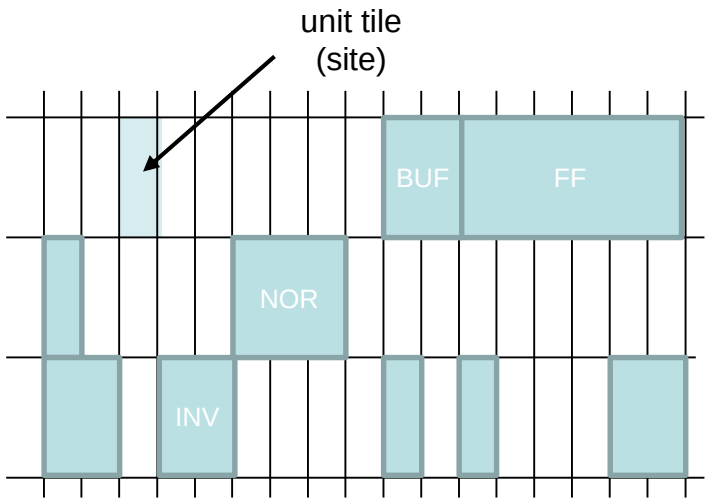


Standard Cell Physical Structure

- Standard cell layout structure is fully defined by the purpose for which they are designed: use in physical synthesis tool
- Because minimizing routing area is more important to achieving a small die size than minimizing the size of the cells, it is best to design cells so that they are best suited for the place and route tools in use.

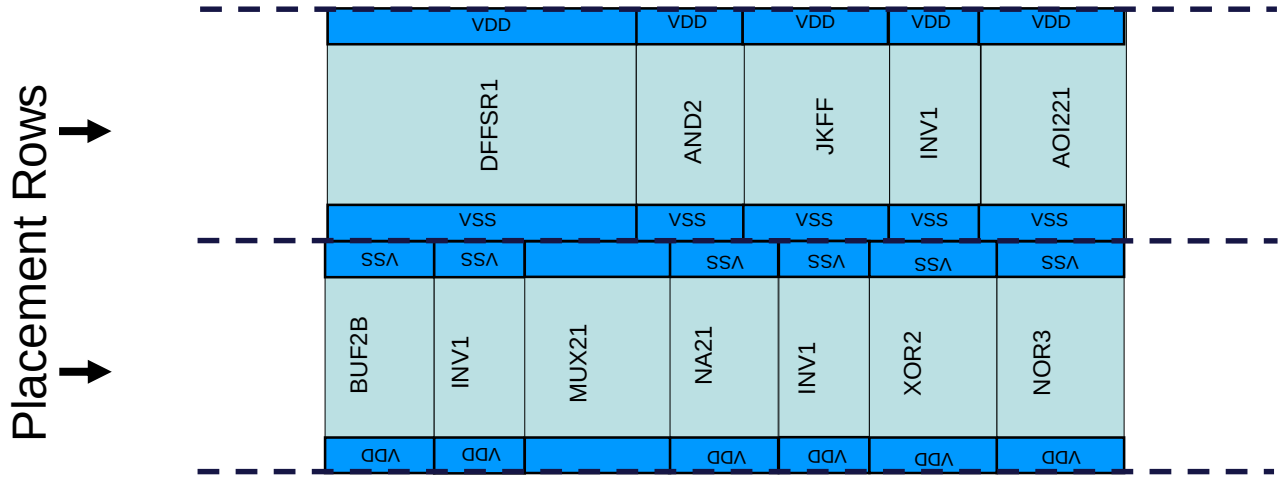
Standard Cell Physical Structure: Unit Tile

- Placement uses vertical and horizontal grid in which cells are placed
- All the cells in the library are designed to be multiple to unit tile

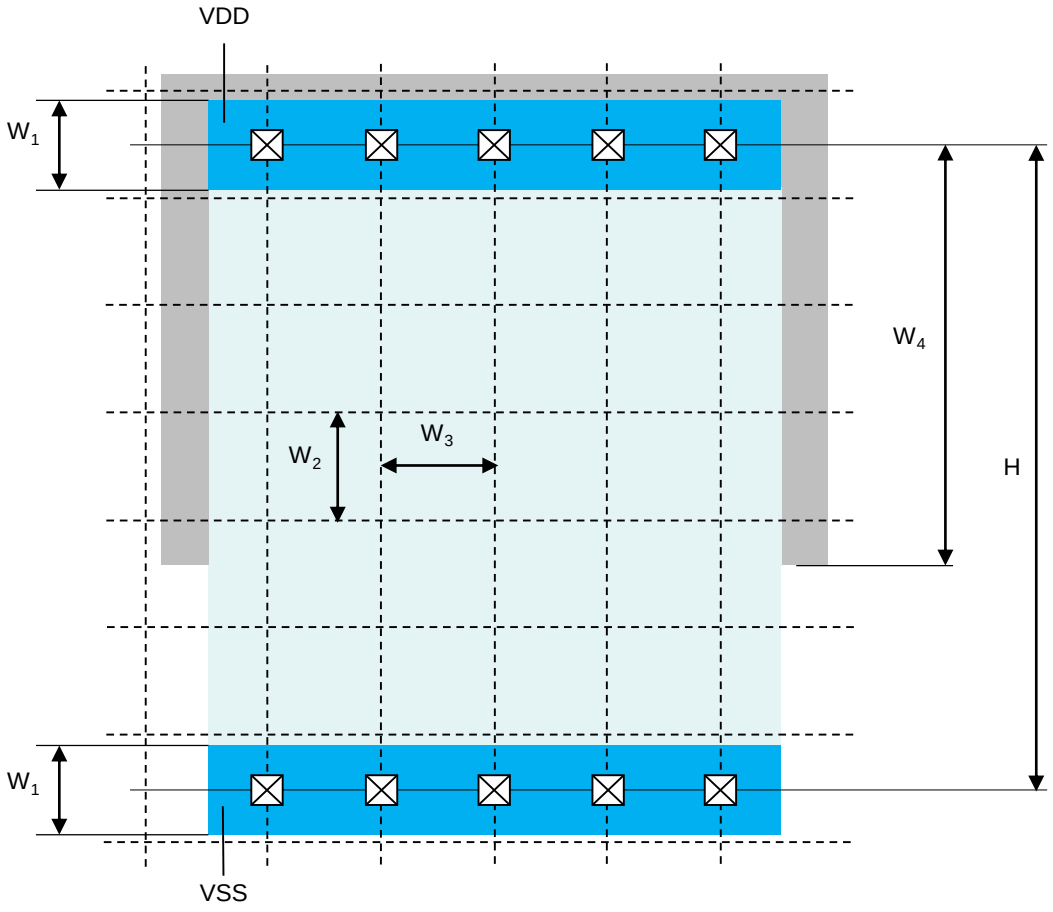


Standard Cell Physical Structure

- Cells are placed in rows, next to each other
- One cells structure continue previous one
- Cells on neighbor rows are flipped so that they can share same supply



Standard Cell Physical Structure (2)



Parameter	Symbol
Cell height	H
Power rail width	W_1
Vertical grid	W_2
Horizontal grid	W_3
N-Well height	W_4

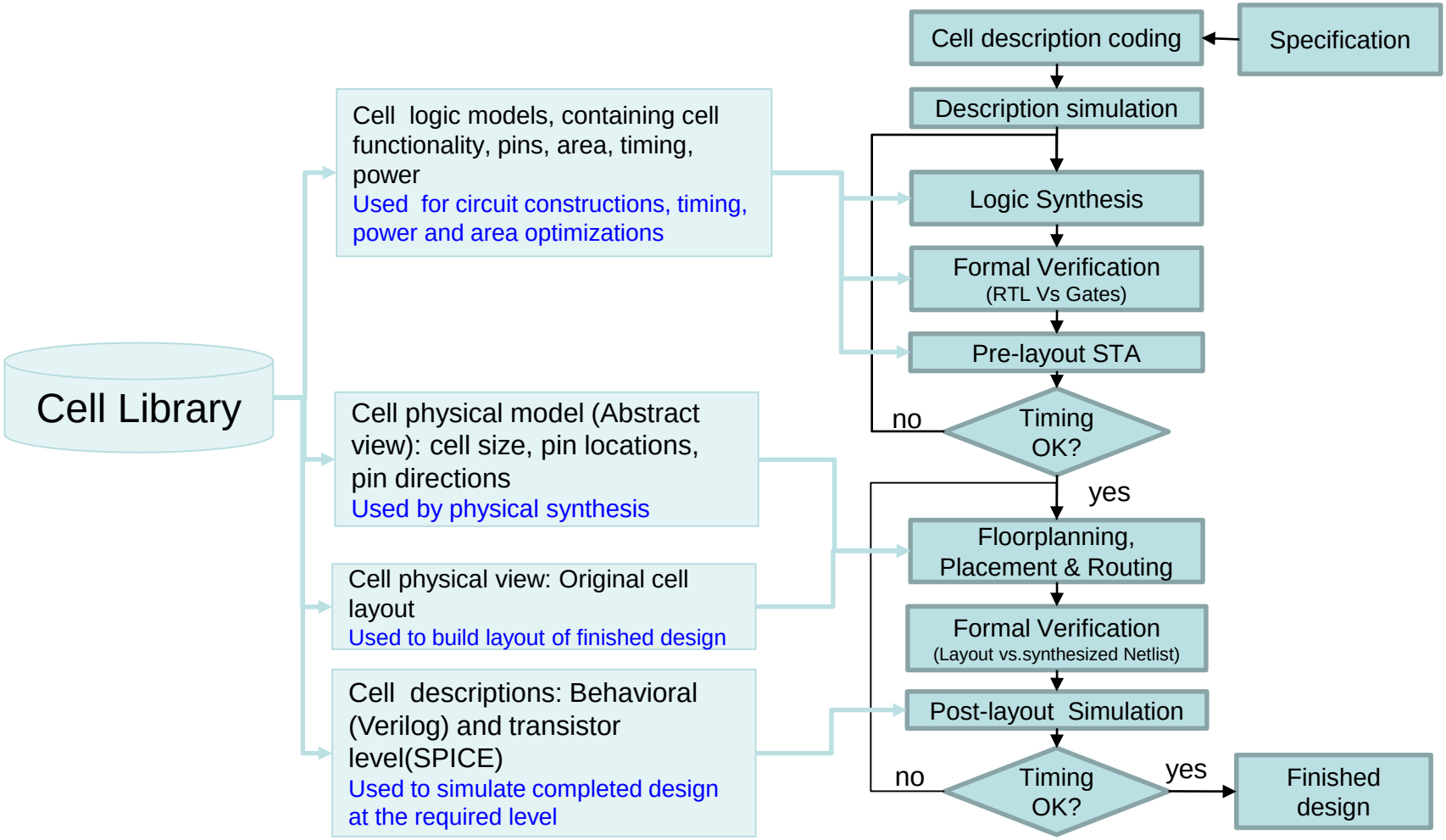
Standard Cell Physical Structure (3)

- All vertical sizes are fixed in a cell as one cell should be continuation of the other one
  - Cell height
  - N-Well height
  - Bus width
- Horizontal size is constrained by vertical grid
  - Cells should be multiple of grid steps
- Pin placement is important for routing
  - Pins should be placed on grid vertices
  - Distance between pins should enable connecting to both without DRC errors

Deliverables

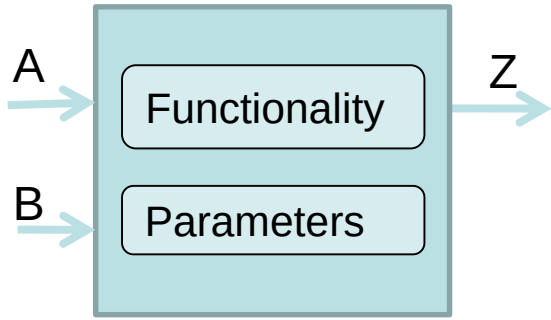
- Two types of deliverables
  - Data: views, files
    - Needed by the design flow or process in which cells are to be used
  - Documentation, reports, etc.
    - Needed by library users to be introduced to library

Necessary Data For Digital Design Flow



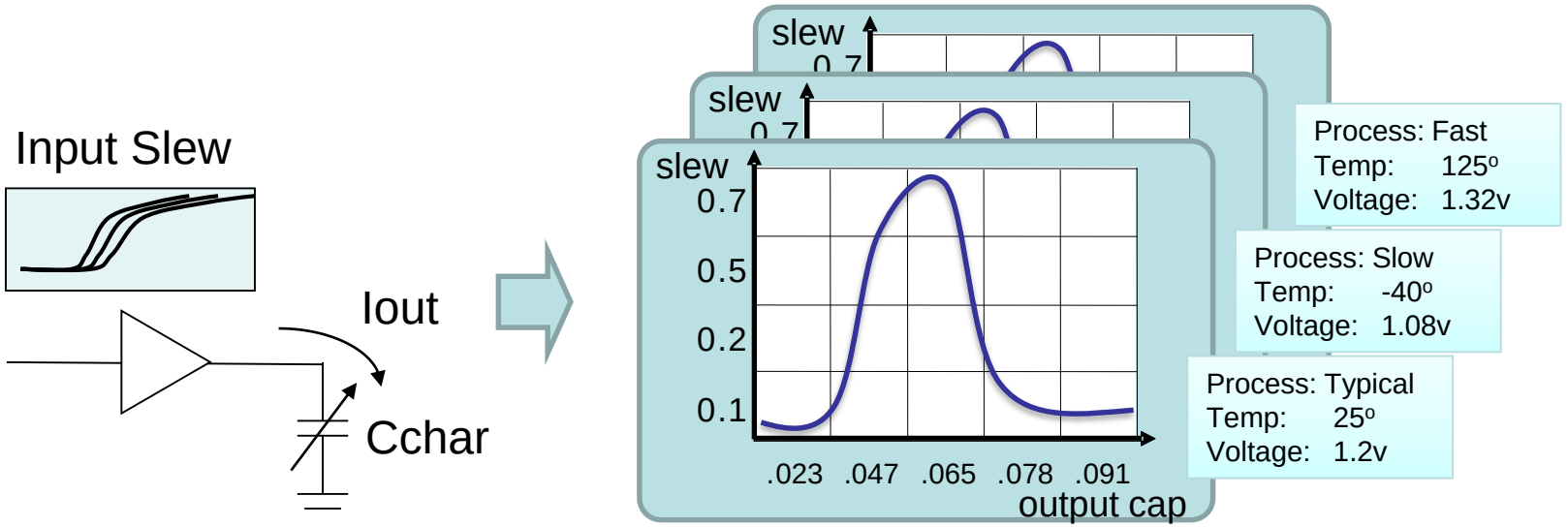
Cell Logic Model

- Cell logic model is generated by characterization process
- Cell model contains
  - Cell name, pins, pin directions
  - Functionality
  - Timing parameters
  - Power parameters
  - Other parameters if needed by EDA tool
    - Pin capacitances
    - etc.

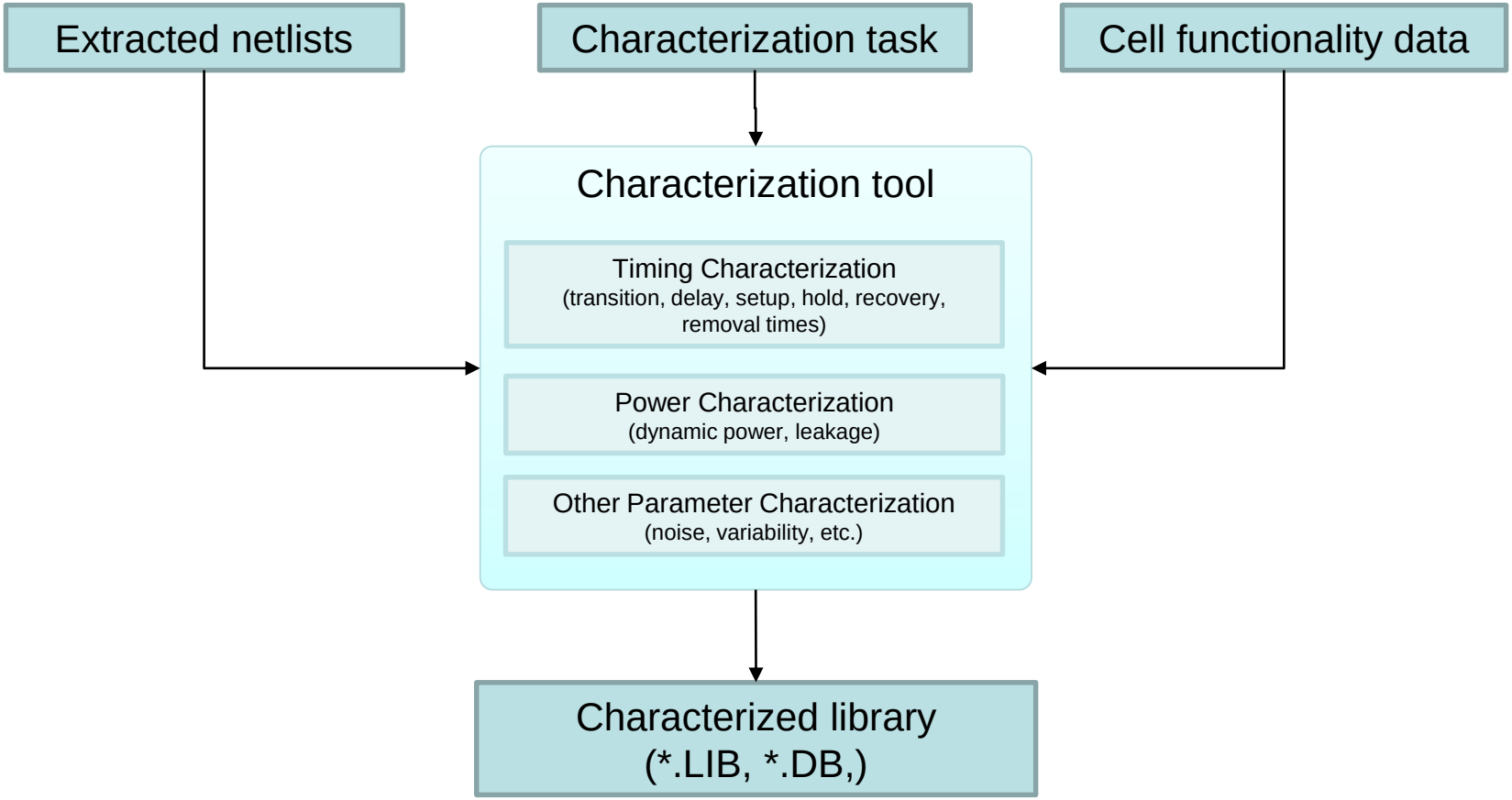


Characterization Goal

- Characterization computes cell parameter (e.g. delay, output current) depending on input variables: output load, input slew, etc.
- Characterization is preformed for various combinations of operating conditions: process, voltage, temperature (also called PVT corners).



Characterization Flow



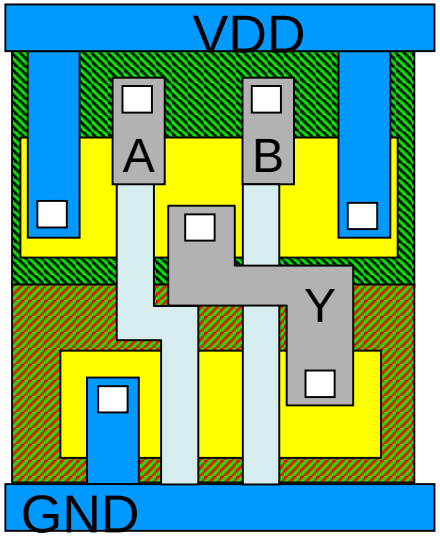
Cell Library Logic Model File

- Synopsys Liberty Format (.lib)
  - Library (.lib) is a text file
  - Content:
    - Cell Function
    - Delays
    - Rise/Fall Times
    - Cell Area
    - Pin Directions
    - Pin Capacitances
    - etc.

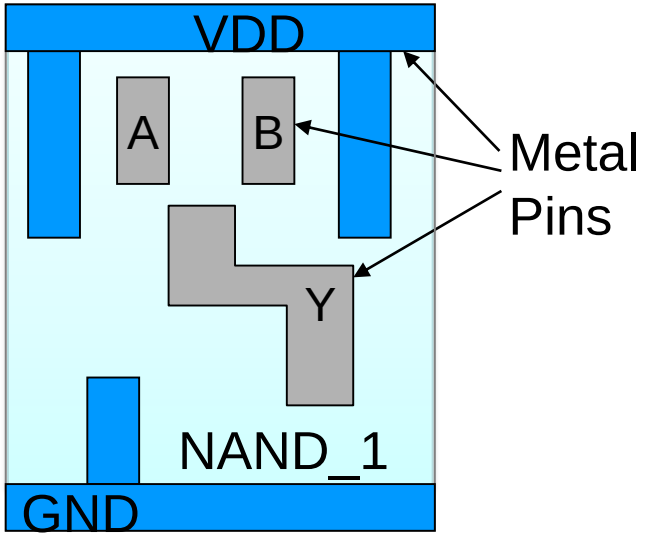
```
library (Digital_Std_Lib) {
technology (cmos);
delay_model : table_lookup;
cell(AND2) {
  area : 2;
  pin(A) {
    direction : input;
  }
  pin(B) {
    direction : input;
  }
  pin(Z) {
    direction : output;
    function : "A*B";
     timing() {
          related_pin : “A" ;
          timing_type : "combinational" ;
          cell_rise(…) {  values("1.0020, 1.1280"}
          rise_transition(…) {  values("0.2069, 0.3315"}
          cell_fall(…) {  values("1.0720, 1.2060"); }
          fall_transition(...) {  values("0.2187, 0.3333"); }
    }
    }
 } /* end of cell */
} /* end of library*/
```

FRAM (Abstract) View of Cell

- Abstract views contain only minimal data needed for placement and routing.



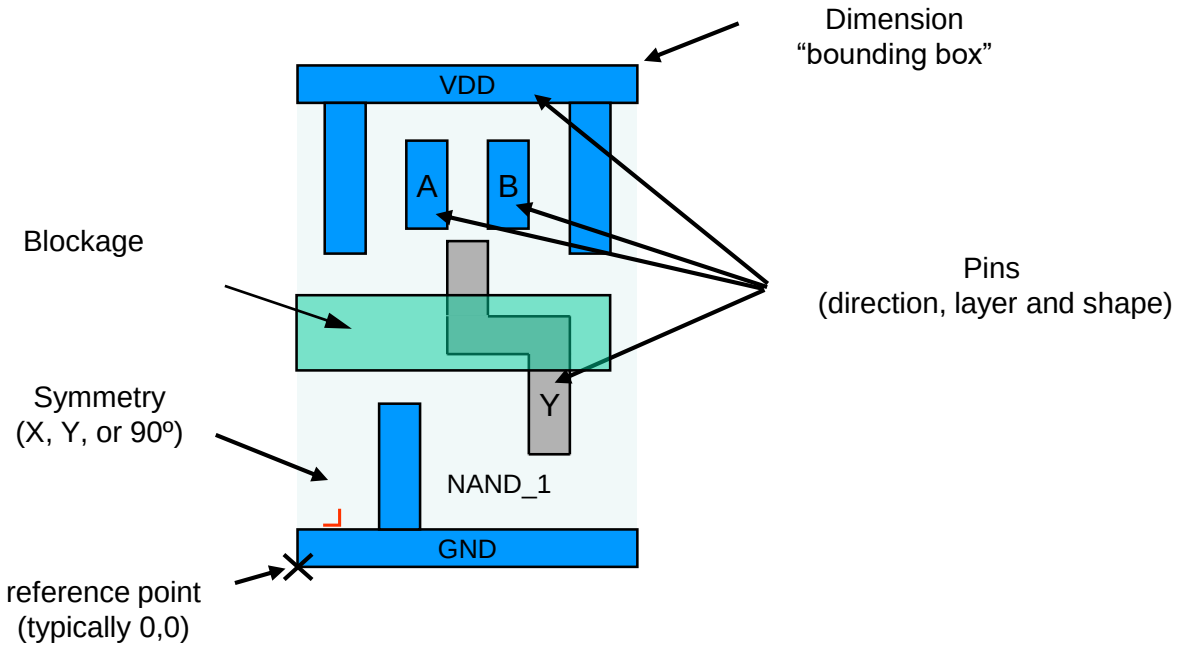
Layout View



Abstract View

FRAM (Abstract) View Content

- Abstract view contain physical information of standard and macro cells necessary for placement



Library Exchange Format (LEF)

- LEF file is the text-based format containing cell physical abstract model contents
  - Cell geometries
  - Pin geometries
  - Blockages
  - Pin antenna information
  - …

```
MACRO single_port_bbb
CLASS BLOCK ;
FOREIGN single_port_bbb ;
ORIGIN 0 0 ;
SIZE 774 BY 547 ;
SYMMETRY X Y R90 ;
PIN OUT
DIRECTION INPUT ;
USE SIGNAL ;
PORT
LAYER M3 ;
RECT 420.180 625.650 420.960 625.810 ;
END
END OUT
OBS
LAYER M1 ;
RECT 0.000 0.000 774.000 547.000 ;
END
END single_port_bbb
```

The Benefits of Text-Based and Binary Formats

Text-based format

- Human readable
- Interoperable
- Can be modified
- Butch processing is possible (using scripts)

Binary format

- Programs do not waste time on conversion to internal representation
- Usually smaller in size than text equivalent

Digital Standard Cell Library Deliverables

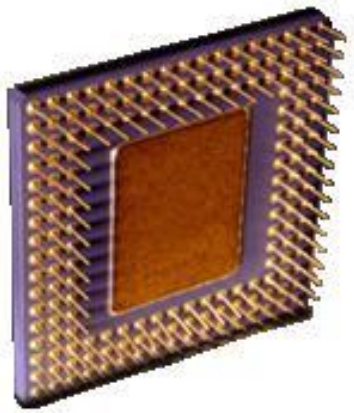
- Logic libraries: Timing, Power
- Footprint (LEF/FRAM) Physical Libraries
- Library GDSII Layouts
- SPICE netlists
- Verilog/VHDL simulation models
- Databook in PDF format
- Datasheets containing timing tables (HTML)
- Layouts' Design Rule Check (DRC) Reports
- Layout versus Schematics (LVS) Check reports
- Simulation results

IC Design Flow: Participants

- Digital design engineers
- Test engineers
- Layout engineers
- CAD engineers
- Semiconductor manufacturers

IC Fabrication

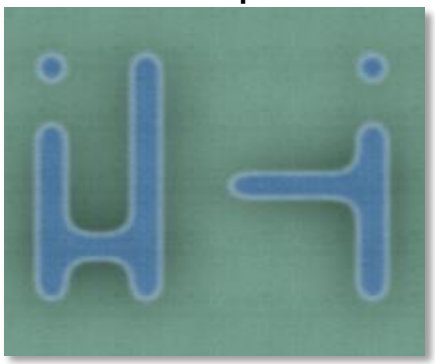
- Semiconductor fabrication performs the following works:
  - Reruns the design
  - Generates masks
  - Performs photolithography process
  - IC Packaging
  - IC Testing
  - Delivers ICs



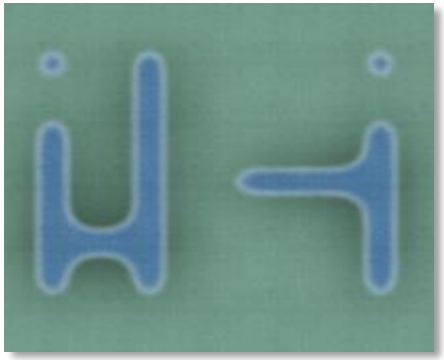
Structure Deration in Deep Submicron Processes

Layout

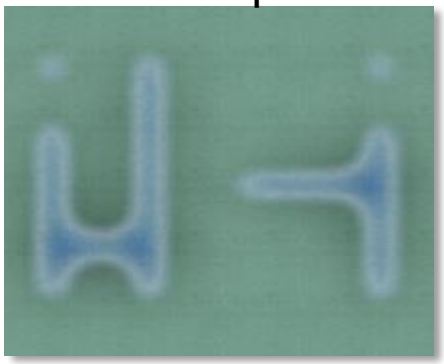
0.25µ



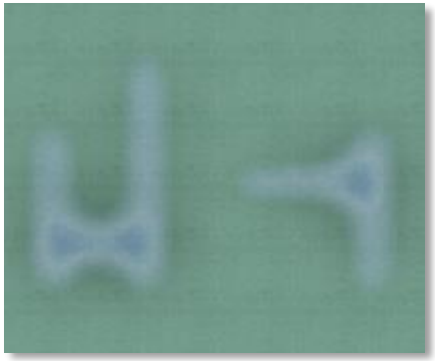
0.18µ



0.15µ

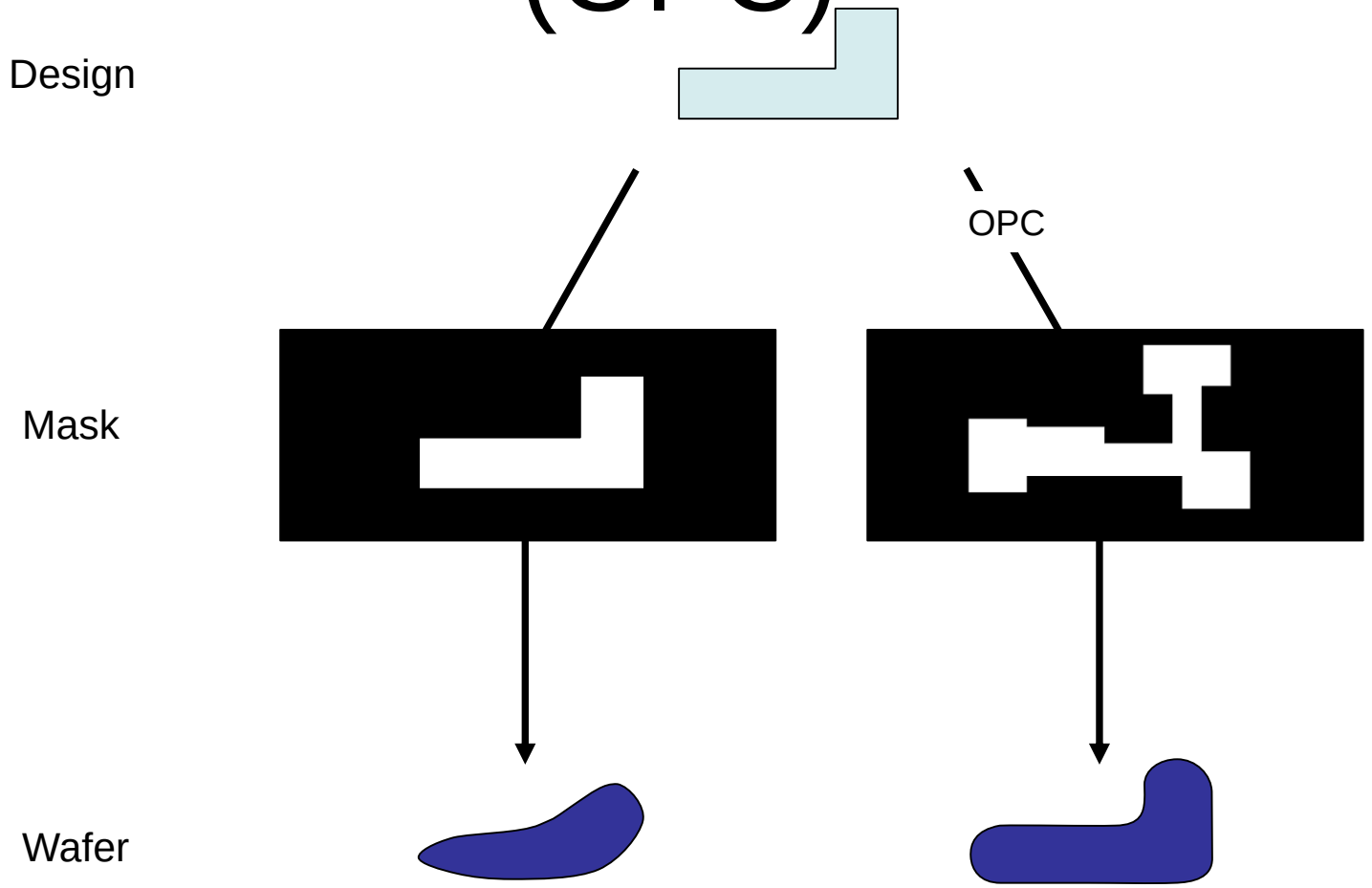


130nm



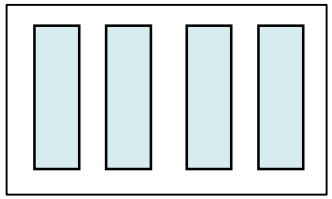
90nm

Optical Proximity Correction (OPC)



Double Patterning

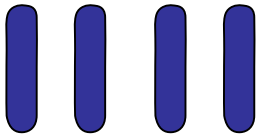
Design



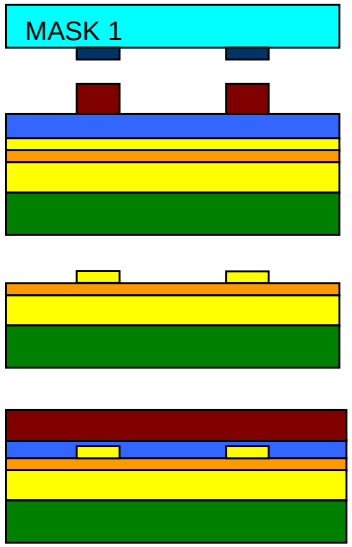
Mask 1

Mask 2

Wafer



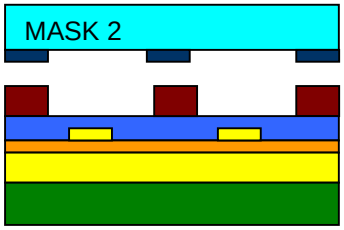
Double Patterning Lithography



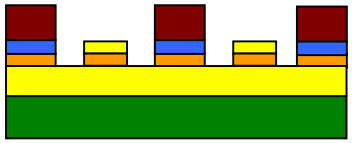
Pattern Photoresist with 1st pattern

Etch Pattern in poly (HM) Strip Resist & BARC

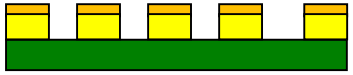
Coat 2nd Barc & Resist Layers



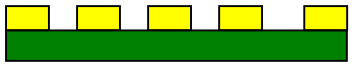
Pattern Photoresist with 2nd pattern



Open BARC Etch & Etch 2nd Oxide HM



Strip Resist & BARC Etch HM Pattern Into Poly



Strip HM

Photoresist
BARC (bottom anti-reflection coating)
Oxide (HM)

Poly
Si Substrate

LELE Double Patterning (Poly Lines – Dual Hardmask)
source: KLA Tencor
URL: http://www.sematech.org/meetings/archives/litho/8376/pres/O-DS-01_Robertson_KLA-T.pdf