EE-429

Fundamentals of VLSI Design

A quick tour around Integrated Circuits

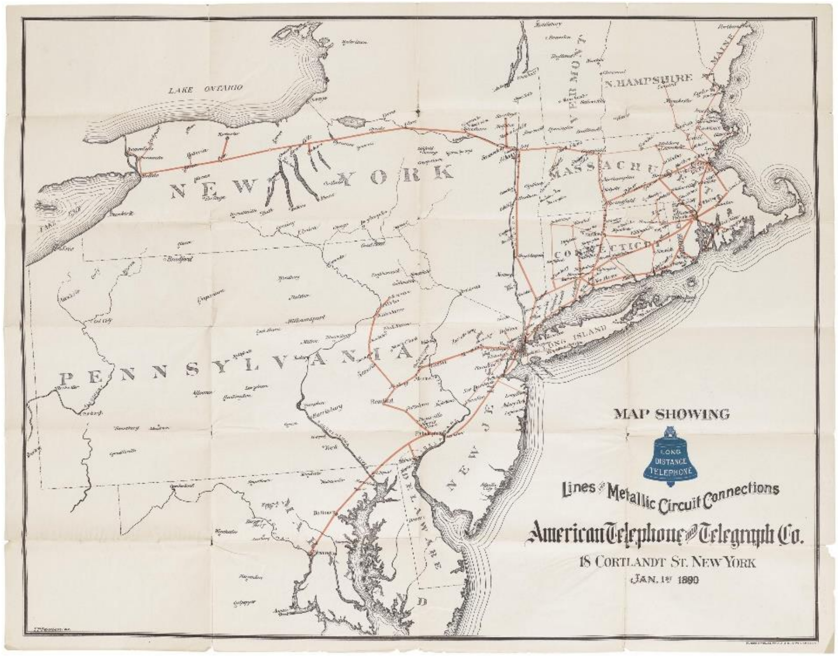
Andreas Burg, Alexandre Levisse

Acknowledgement: Prof. Adam Teman (BIU)

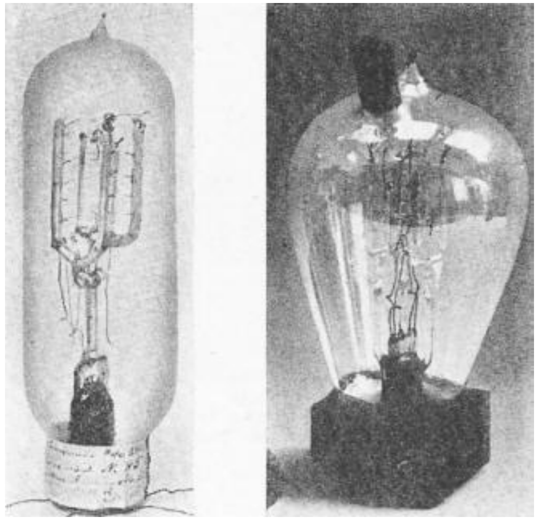
How we got to the Transistor

- **The original problem from which it all started:**
  - Telephone connections over long distances were difficult as signals were heavily attenuated
  - Need for an “amplifier” to connect US East-and-West coast with telephone lines

Analog telephones



Connected by long cables with large attenuation

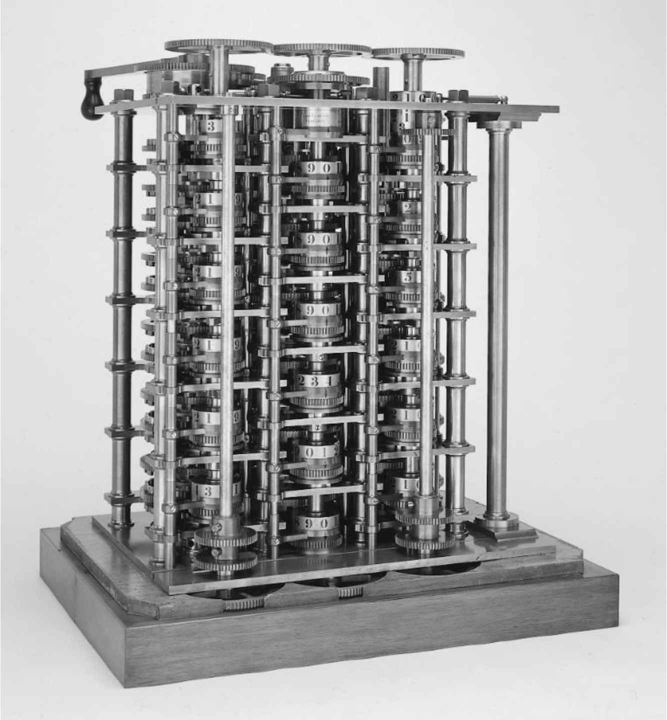


Vacuum tubes allowed to build amplifiers

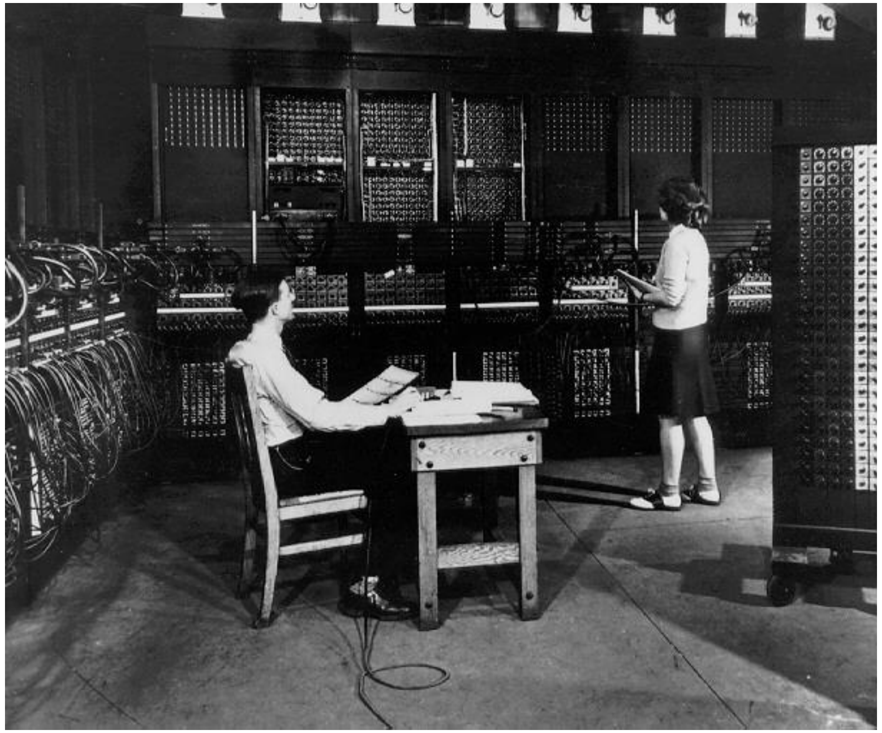
How it all started – The early days (pre-silicon)

Automation → Computation

Jaquet-Droz automata: 1768
Mechanical system carrying out a sequence of movements
Shown in Neuchatel Art and History Museum (still functional)



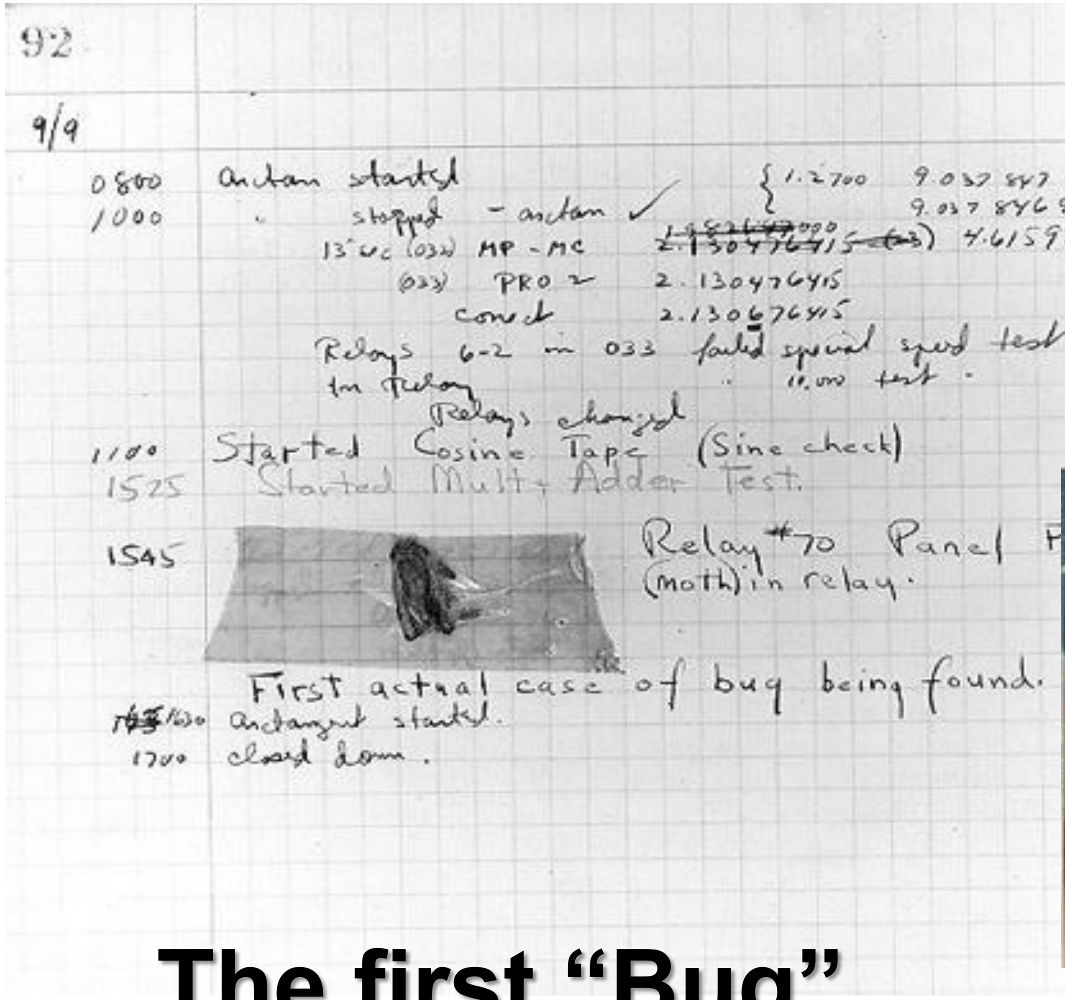
Babbage Difference Engine: 1834*
Punch card, steam-driven machine that can run a program
*Physically built only in 1991



ENIAC (1946)

First Universal **Electronic** Computer. 18,000 electronic valves, weighed 30 Tons and consumed 25kW: 100,000 calculations a second.

When ENIAC did not work



92

9/9

0800 Antan started
1000 " stopped - antan ✓ {1.2700 9.037 847 025
9.037 846 995 correct
13" uc (032) MP - MC ~~1.982147000~~ ~~2.130476415~~ (~~-3~~) 4.615925059(-2)
(033) PRO 2 2.130476415
correct 2.130676415
Relays 6-2 in 033 failed special speed test
In Relay " 11.000 test .
Relays changed
1100 Started Cosine Tape (Sine check)
1525 Started Mult+ Adder Test.
1545 Relay #70 Panel F
(moth) in relay.
First actual case of bug being found.
1630 Antangent started.
1700 closed down.

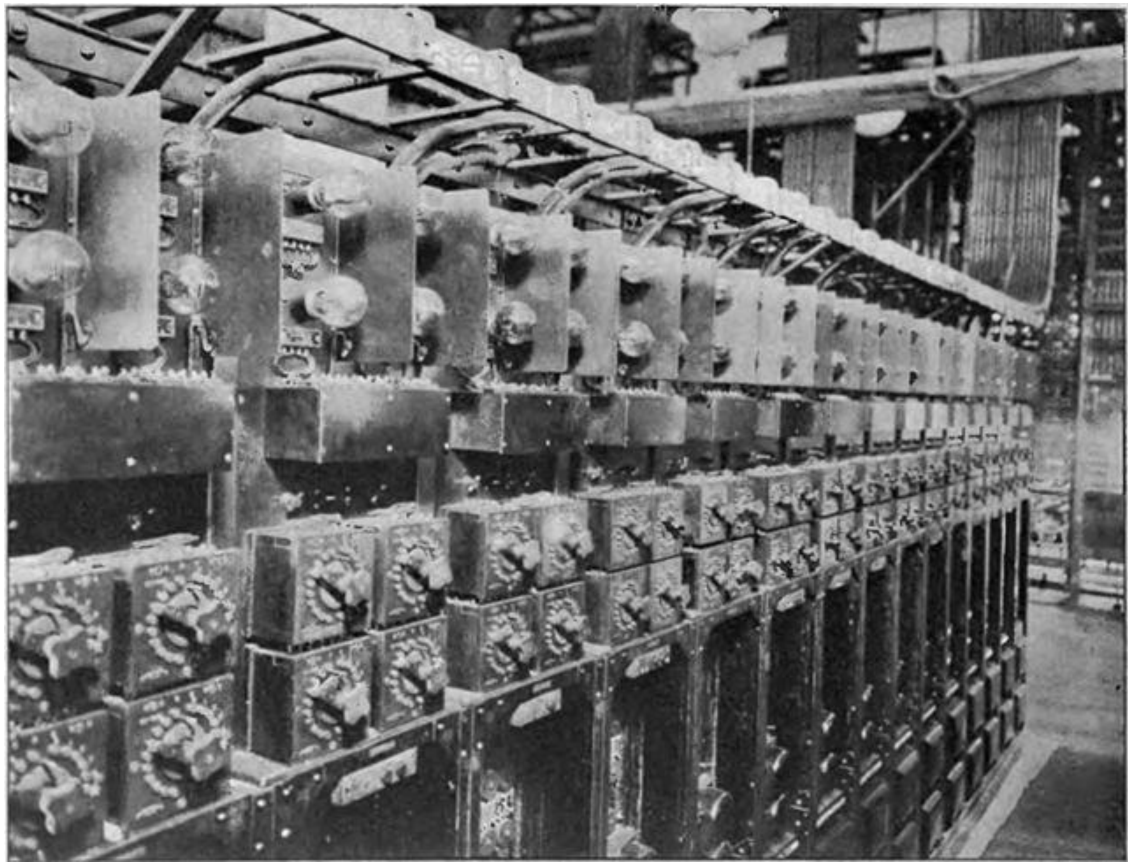
The first “Bug”

Grace Brewster Murray Hopper

Inventor of the infamous Bug!

How we got to the Transistor

- **Telephone repeaters had 1000s of vacuum tubes, producing enormous heat**
  - American Telephone and Telegraph (AT&T) set out to solve this problem and build a more efficient amplifier
  - AT&T research:
    Bell Labs

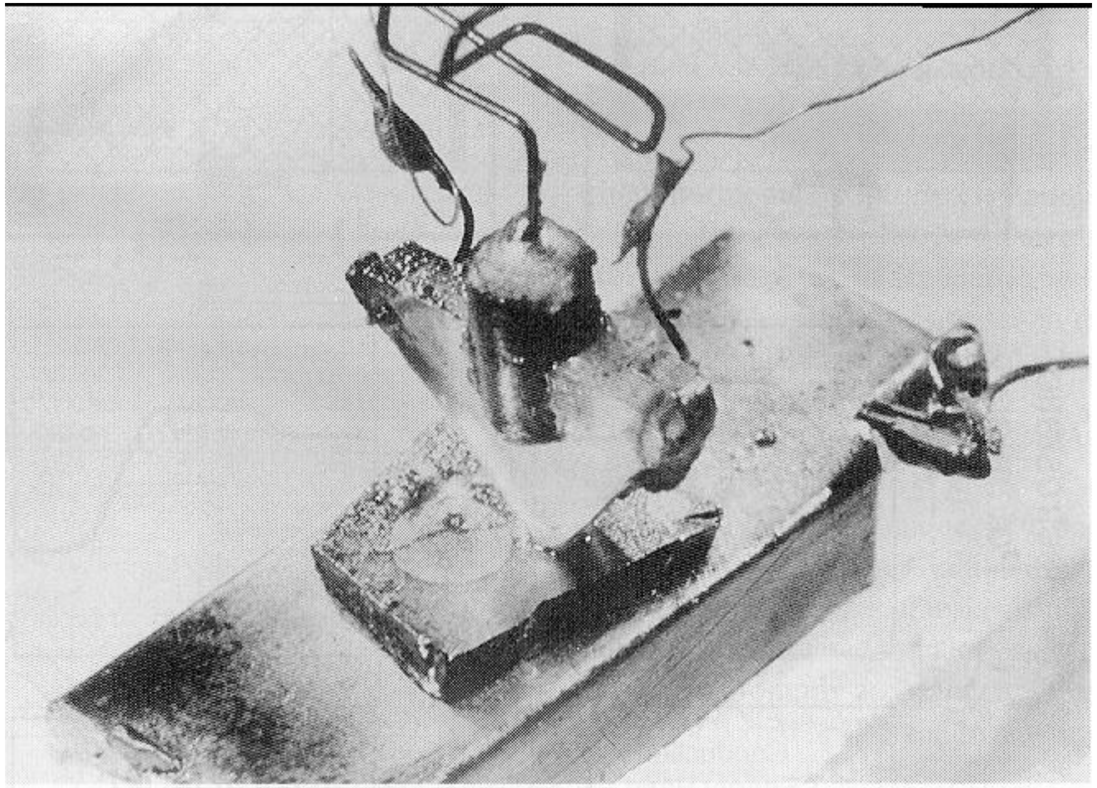


How it all started – The era of the Transistor

- **1948: Point Contact Transistor invented at Bell Labs**
  - William Shockley had devised the theory in 1947
  - John Bardeen and Walter Brattain built the first prototype
  - Made from Germanium

- Served as the basis for the BJT which followed soon after and showed much better behaviour

- **First transistors commercialized** by Texas Instruments **in 1954** for portable radios

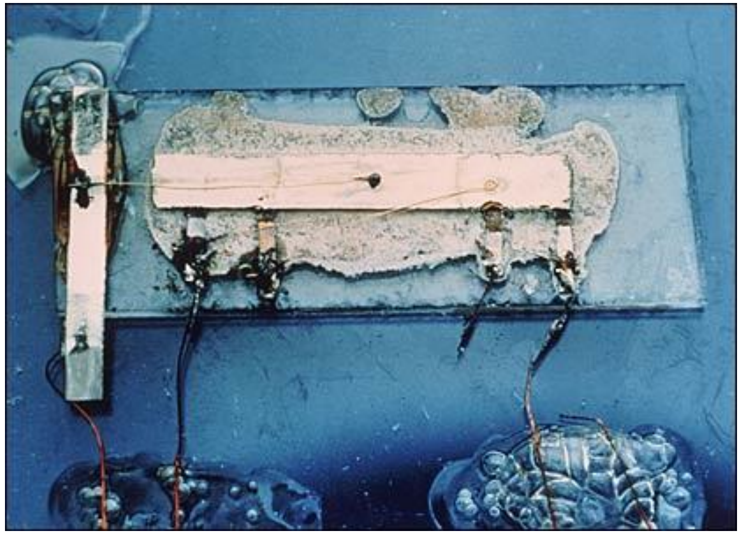


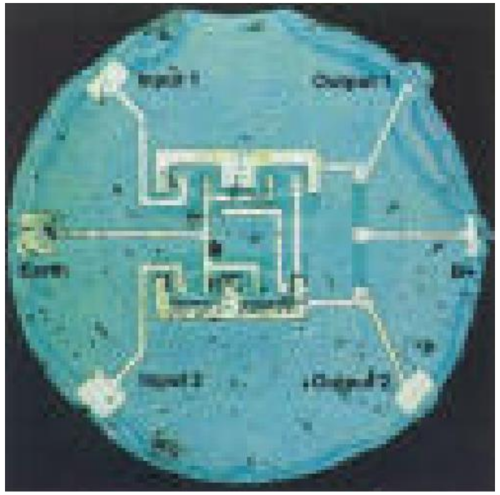
Planar Process: the Key to Large Scale Integration

- **1958: Integrated Circuit invented by Jack Kilby at Texas Instruments and Robert Noyce** (co-founded Fairchild and later Intel)

- **Integrated circuits combine multiple devices on a single substrate**
  - First ICs were about 50 USD in quantity

- **All devices are manufactured in parallel**
  - In principal, the fabrication effort is independent of the number of devices
  - Key to enable scaling

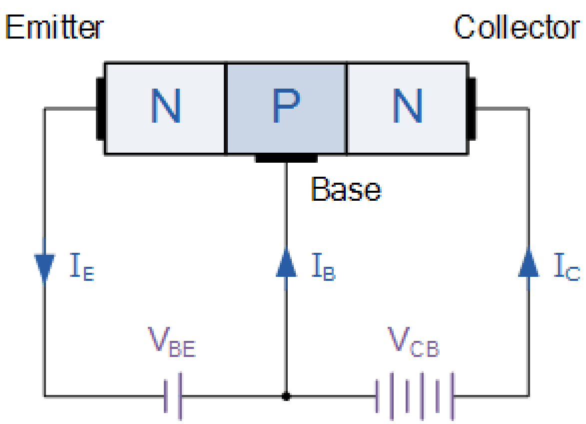


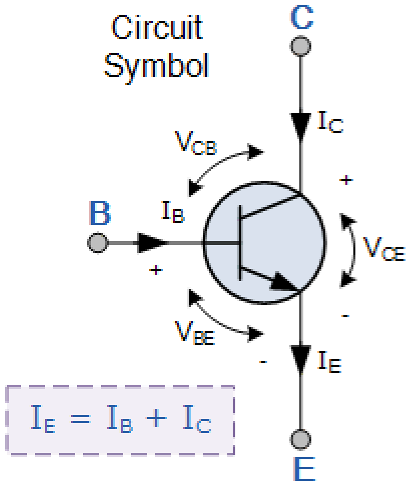


Fairchild 1961 (Flip Flop)

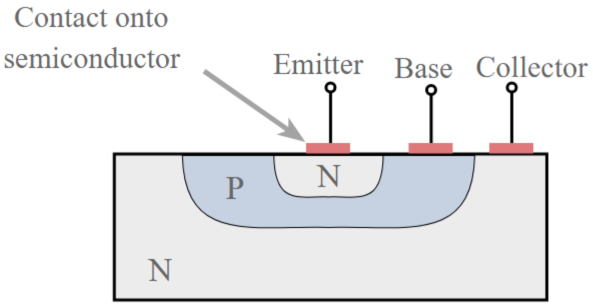
The First Digital Integrated Circuits

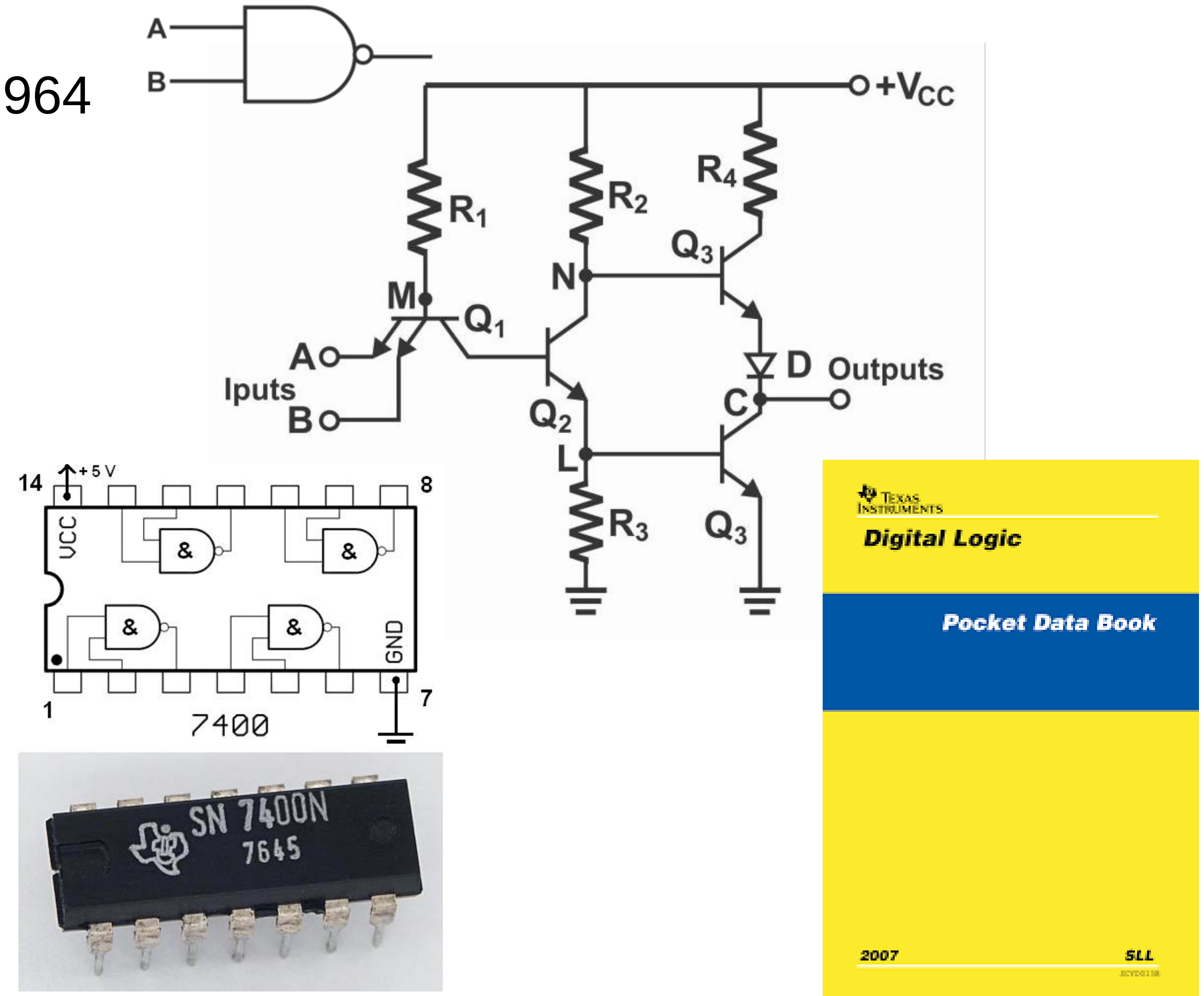
- **The first integrated circuits were based on Bipolar Transistors (BJTs)**
  - BJTs: current controlled devices
  - Transistor-Transistor Logic (TTL) introduced in 1964





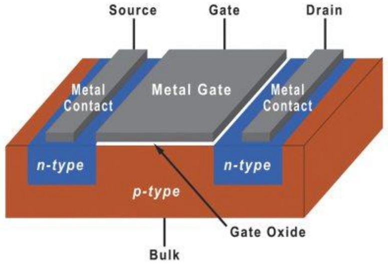
$I_E = I_B + I_C$



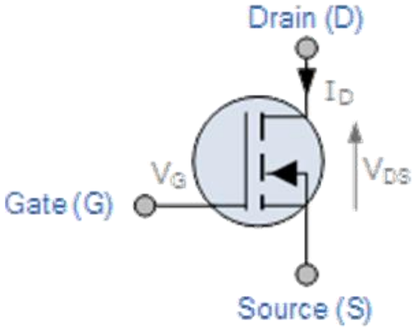


Transition to (C)MOS – Invented in 1963

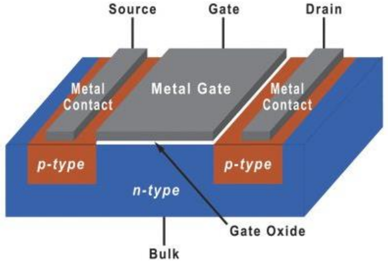
- **Issue of TTL logic: power consumption even without any activity**
- **Solution: CMOS based on voltage controlled Field Effect Transistors**
  - No static power consumption



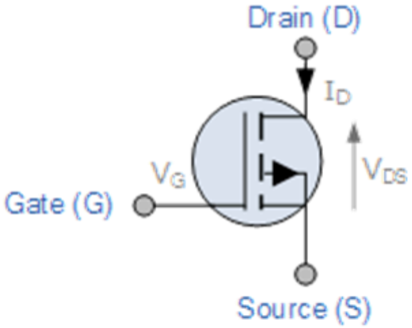
(a)



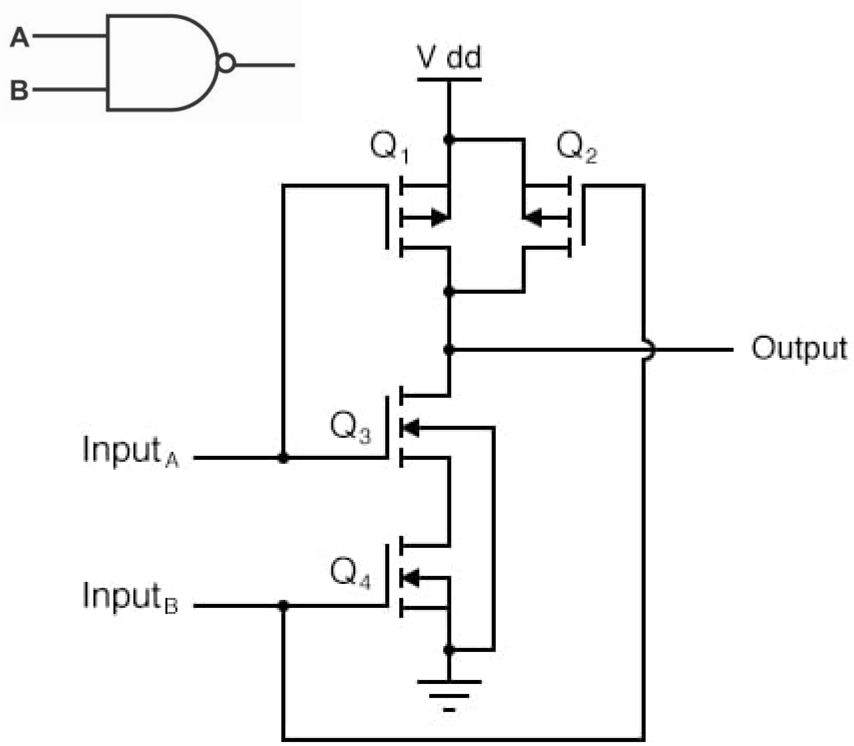
N-channel



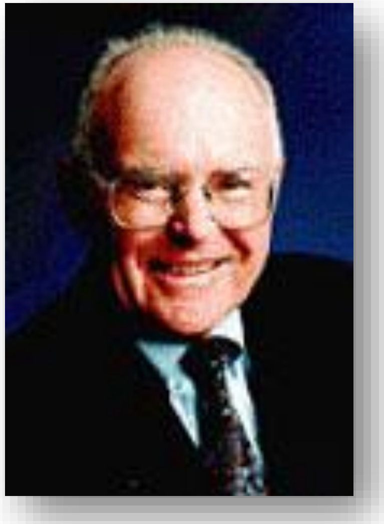
(b)



P-channel



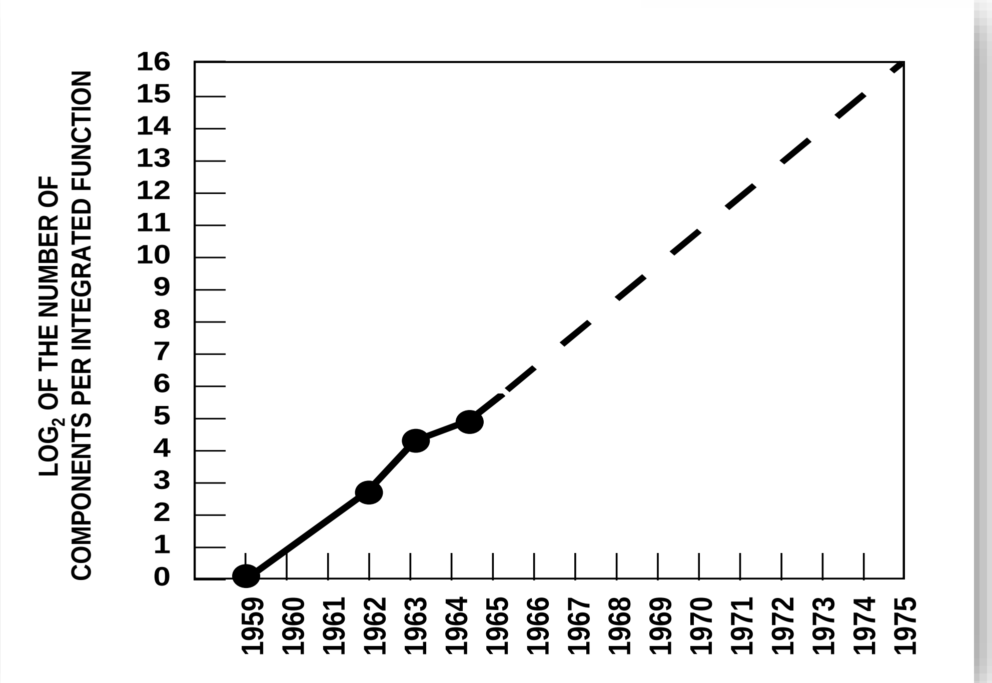
Moore's Law (1965)



- Gordon Moore (INTEL CEO) writes his famous paper entitled: **"Cramming more Components onto Integrated Circuits"**
- He predicting that "*by 1975, the number of components per integrated circuit for minimum cost will be 65,000*"

- **"The number of transistors on a chip doubles every 18 to 24 months."**

- **Remarkable:** his **prediction** is **based on** only **4 data points**



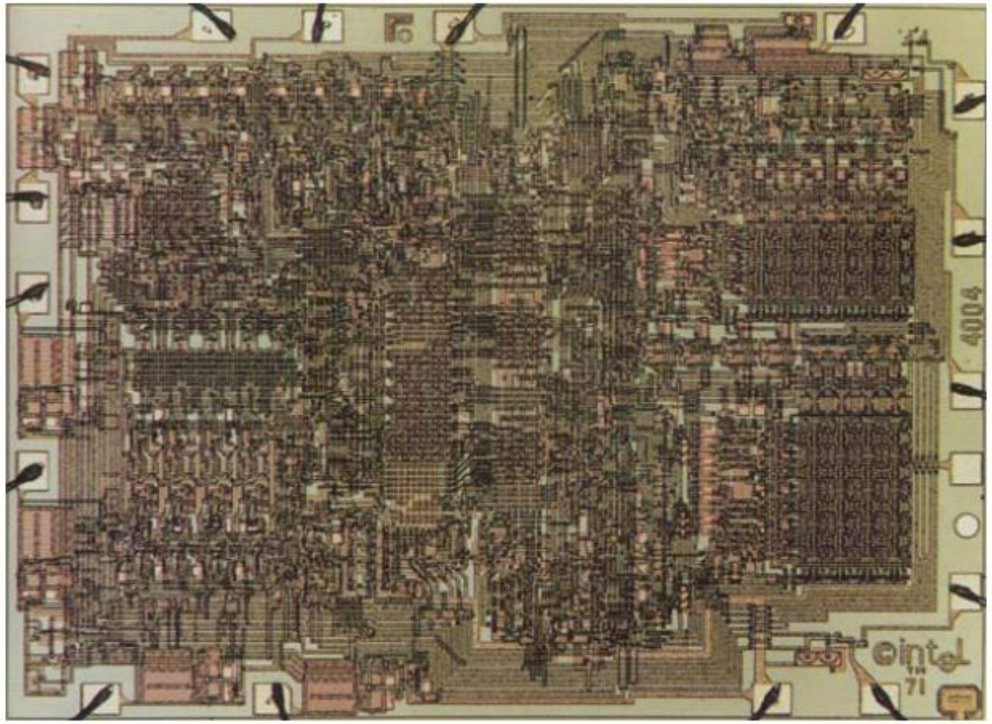
Electronics, April 19, 1965.

(C)MOS Enabled Larger Number of Transistors

- **1968: Noyce and Moore left Fairchild to start INTEL**
- **1971: Intel introduces the first microprocessor, the 4004 (originally designed as a special circuit for a customer)**

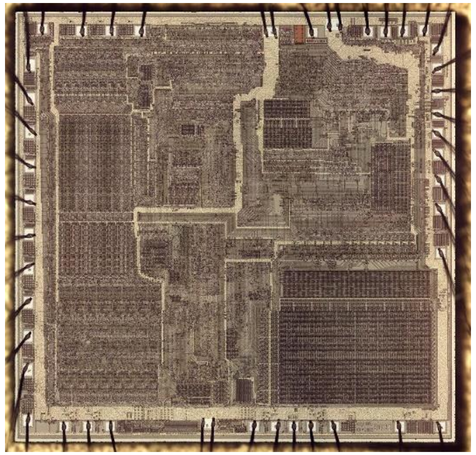
4-bit Microprocessor
740 kHz clock
92’000 instructions / second

2’300 transistors
10us p-MOS process

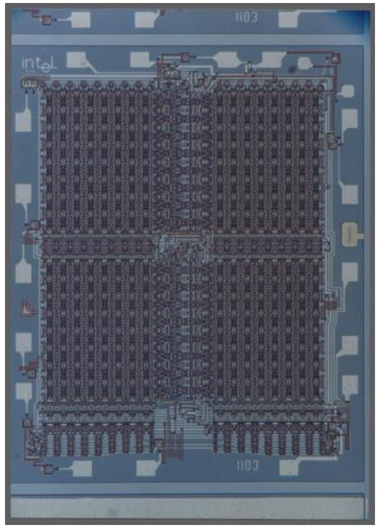


From this point on technology progressed rapidly

- **The Transistor Era**
  - 1960 – First MOSFET Fabricated
  - 1962 – TTL Invented
  - **1963 – CMOS Invented (solve TTL Power issue)**
  - 1964 – 1-inch silicon wafers introduced
  - **1965 – Moore's Law** ➔ **TECHNOLOGY DRIVER**
  - 1967 – Floating Gate invented
  - **1970 – First commercial DRAM (1Kbit) by INTEL**
  - **1971 – First Microprocessor from INTEL**
  - 1978 – Intel 8086/8088
  - 1981 – IBM PC is introduced



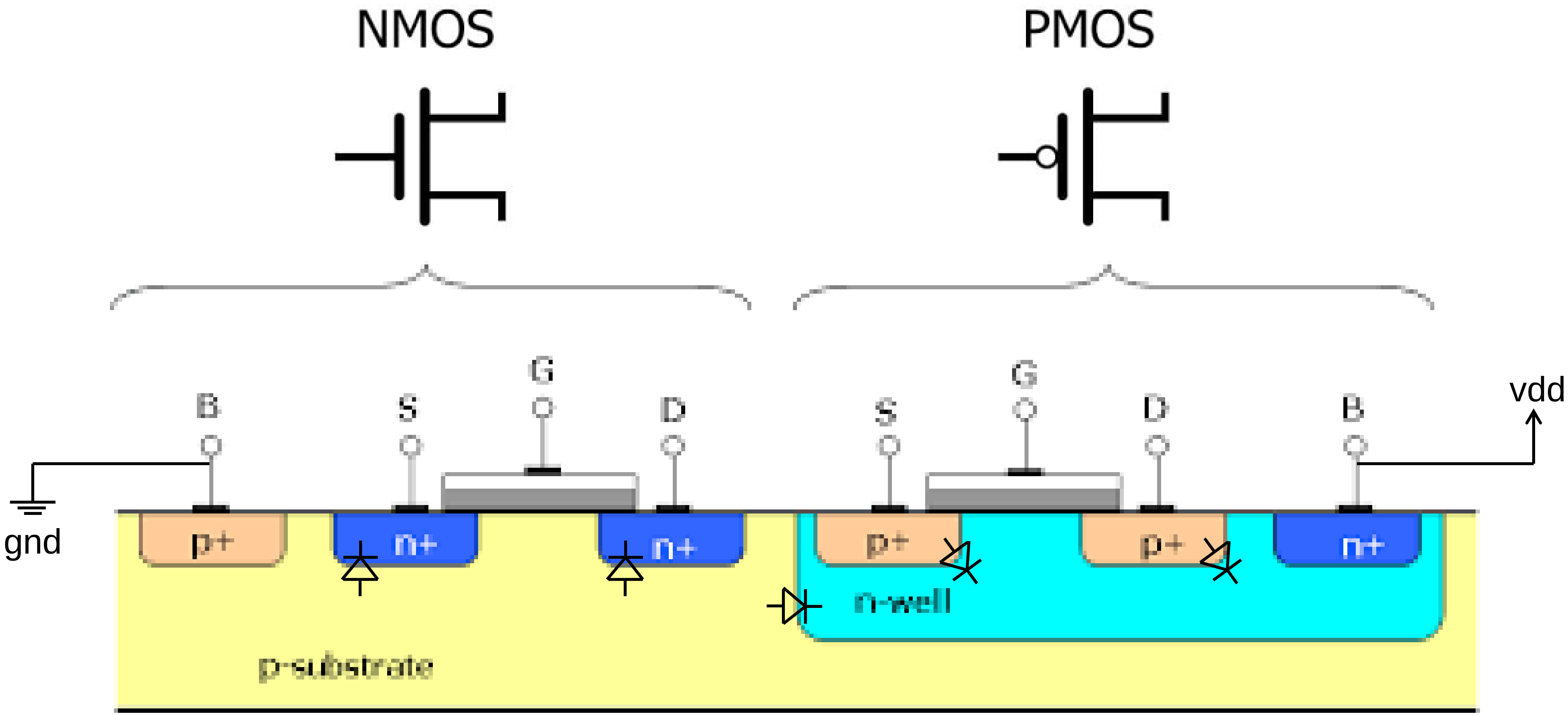
16-bit Microprocessor
5-10 MHz clock
29'000 transistors
3us CMOS process



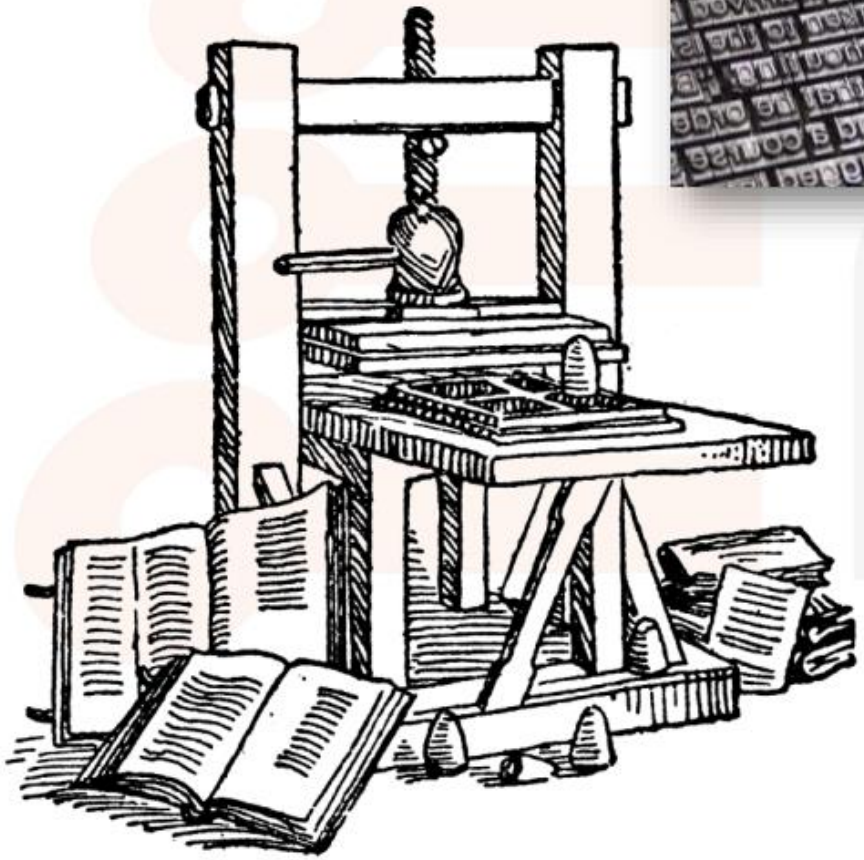
1 kbit DRAM
300ns access time

Planar CMOS Transistor

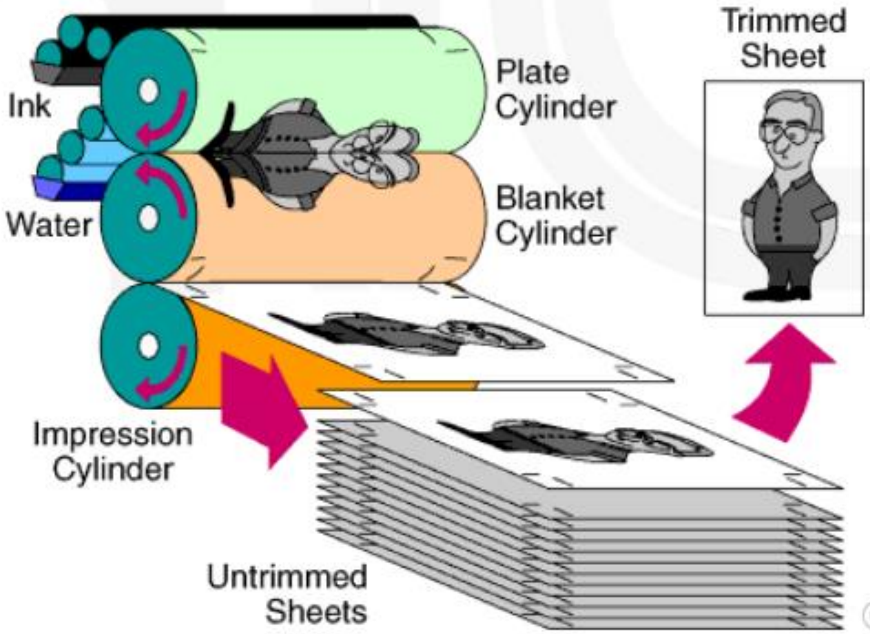
- **CMOS requires two types of transistors: NMOS and PMOS**



Printing Transistors



Source: Capital Poly

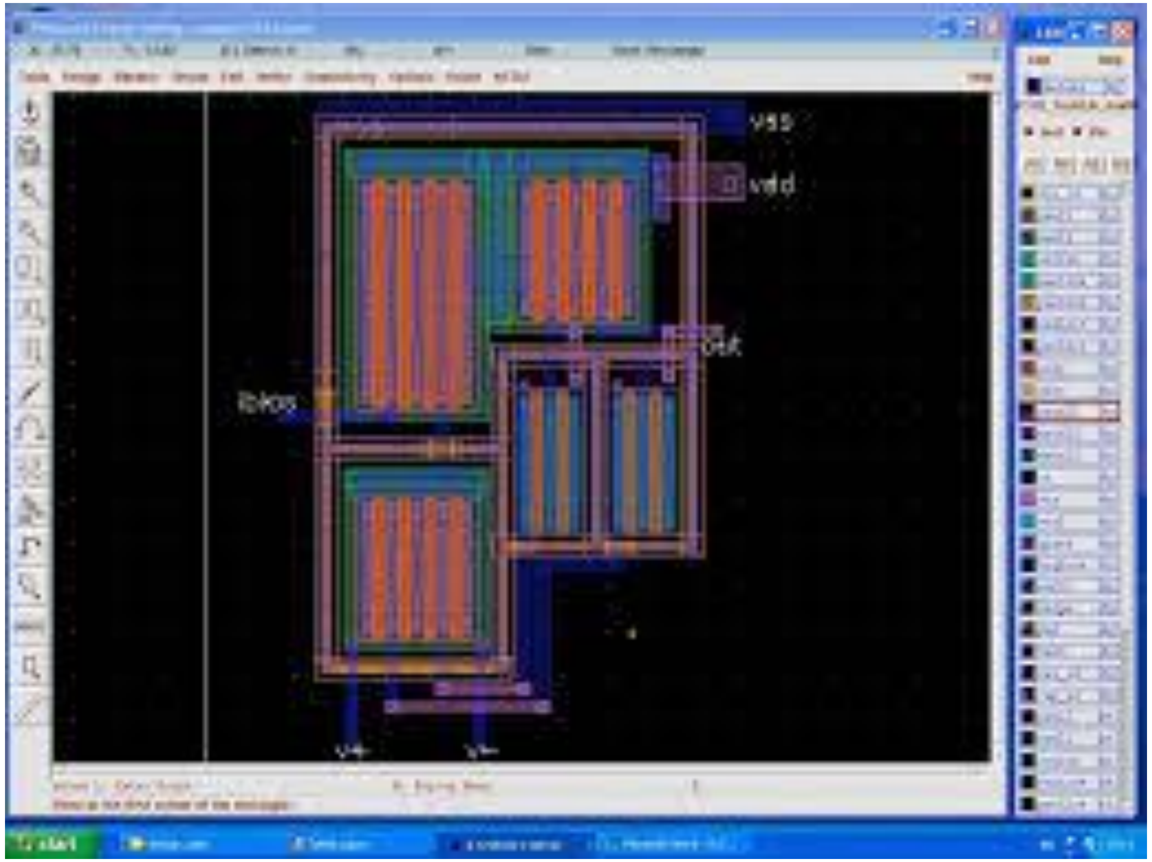


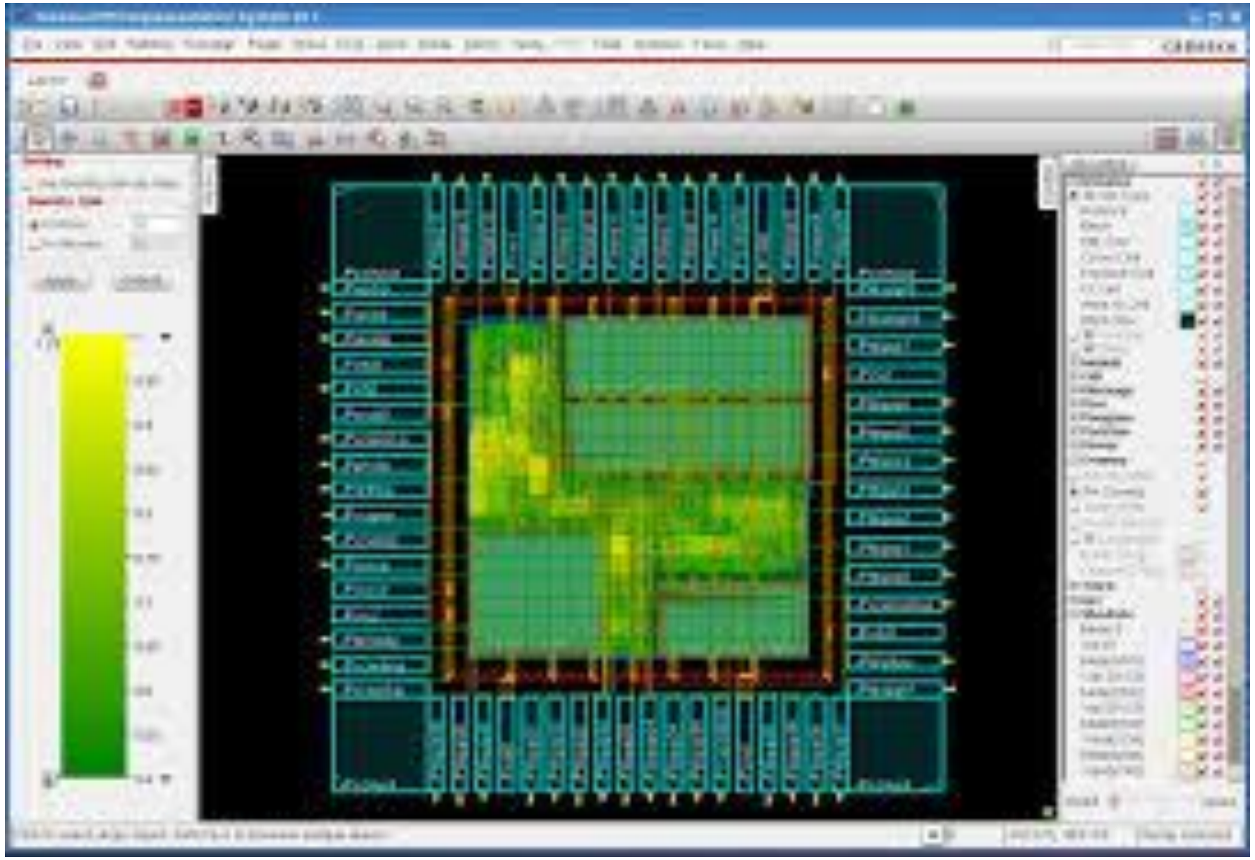
Layout/Mask preparation in the 1960s

Mask carved out of Rubylith by hand

Layout Masks Today

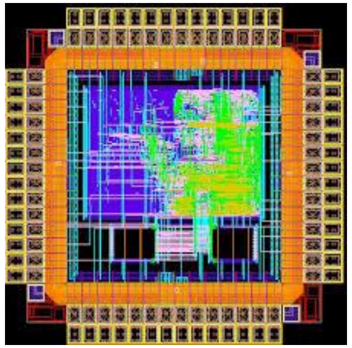
- **Designed with modern EDA tools on the computer**



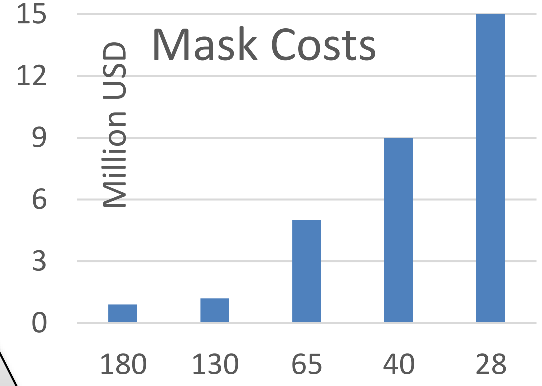


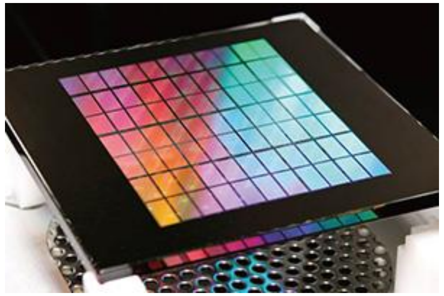
Mask Production

- **Photo masks are produced with eBeam process**
  - Masks are incredibly expensive (millions), but they are the basis for millions of chips



Layout with multiple layers



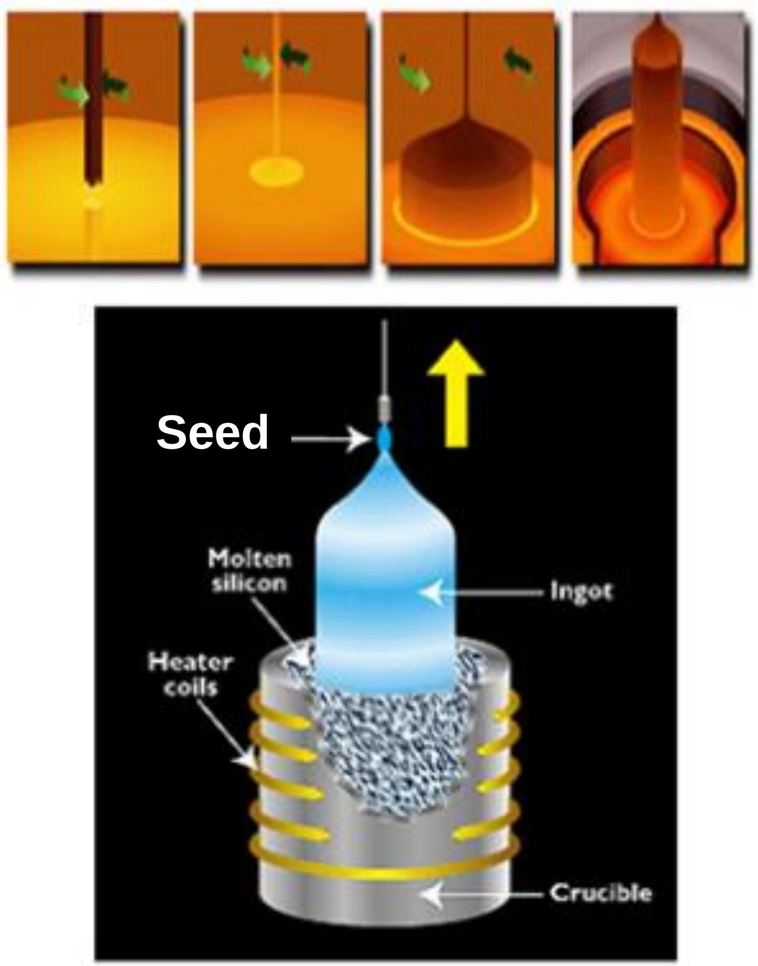


Multiple masks:

- Complex processes need more masks

From “Sand” to Wafers

- **Chips are made from Silicon**, just as Sand, but **with a purity of 99.999999999%**



Czochralski process.

Weights hundreds of kilos and costs 20k USD

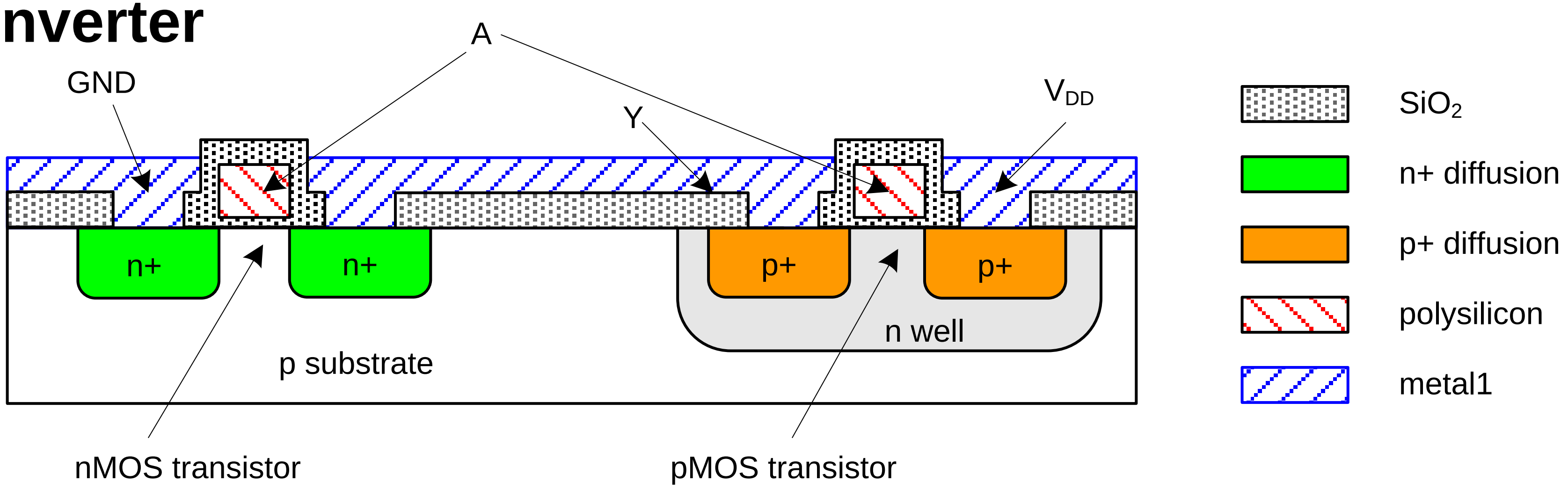
Wafers: size has grown over time.
TODAY’s standard: 12” (300mm)

Manufacturing Example

- **CMOS integrated circuits require active devices and interconnect**

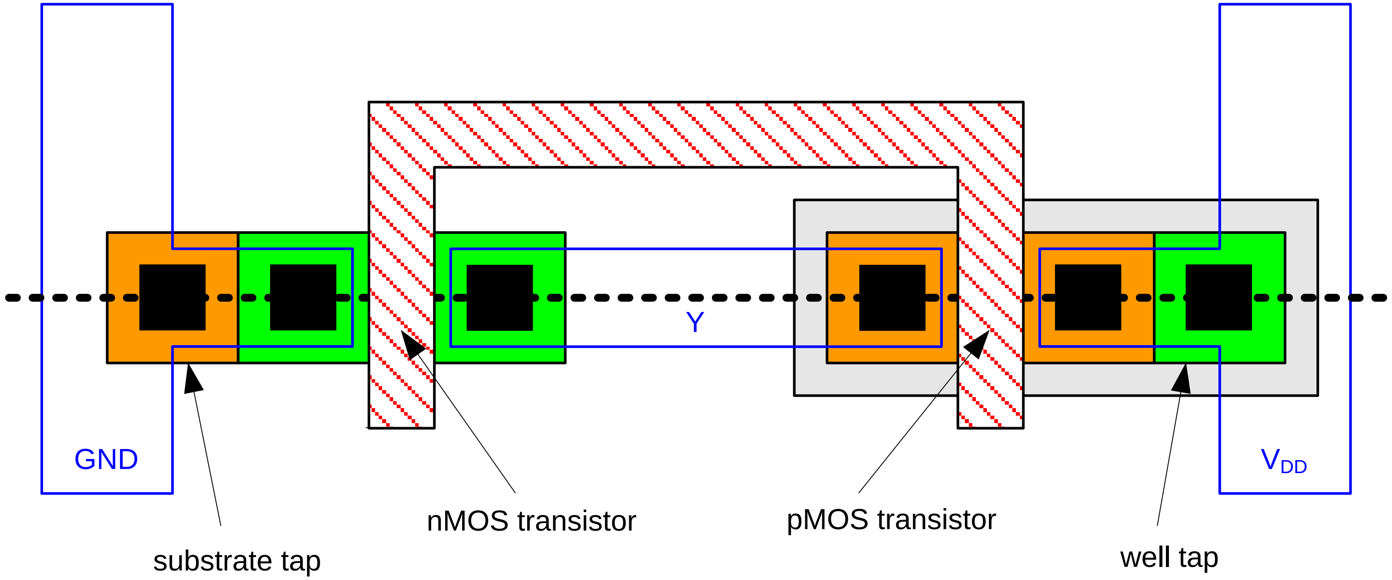
- **Active devices: nMOS and pMOS transistors**
  - Made from doped silicon regions
- **Interconnect: wires that connect the devices**
  - Made from different metal layers and vias that connect these layers

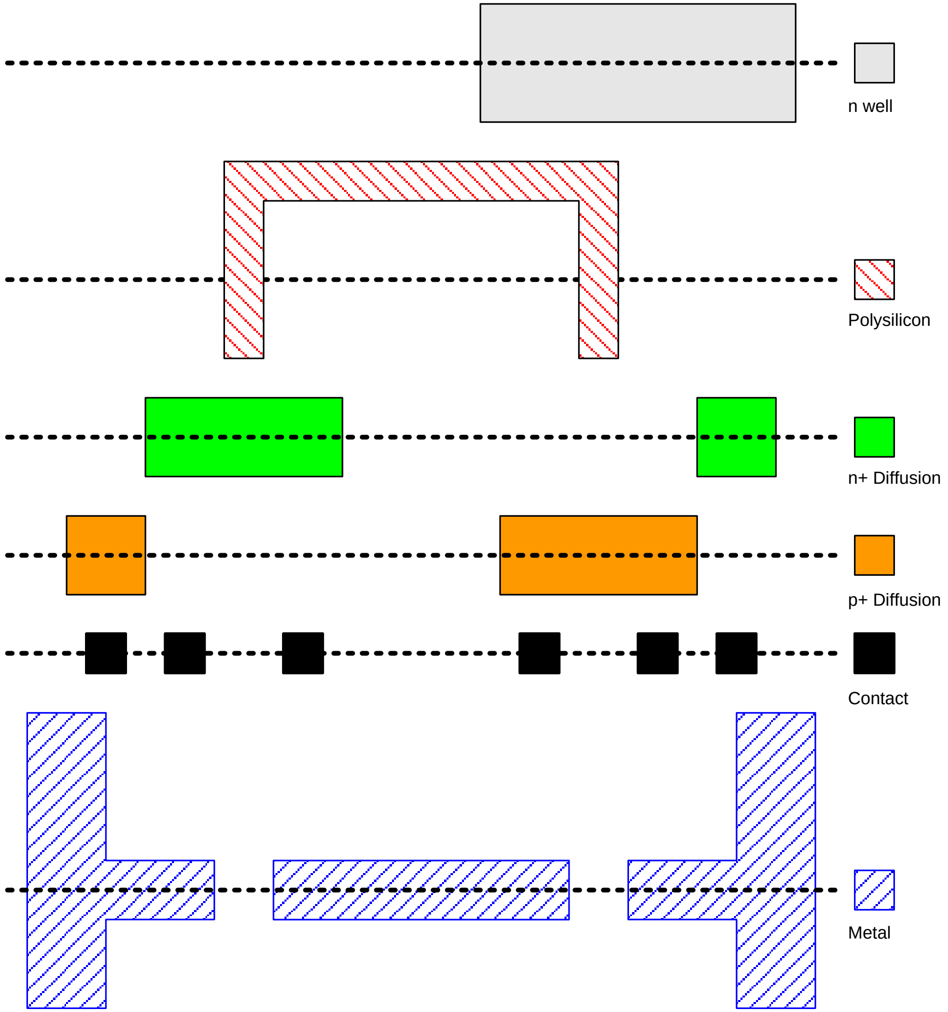
- **Example: CMOS inverter**
  - Cross section:



Mask Sets for an Inverter

- **Transistors and wires are defined by *masks***
  - Cross-section taken along dashed line

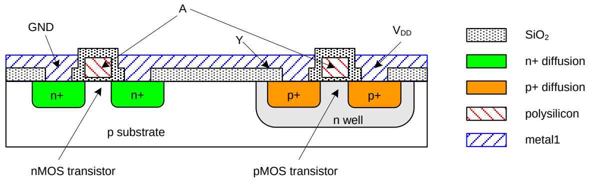




- **Modern technologies require dozens of different masks**

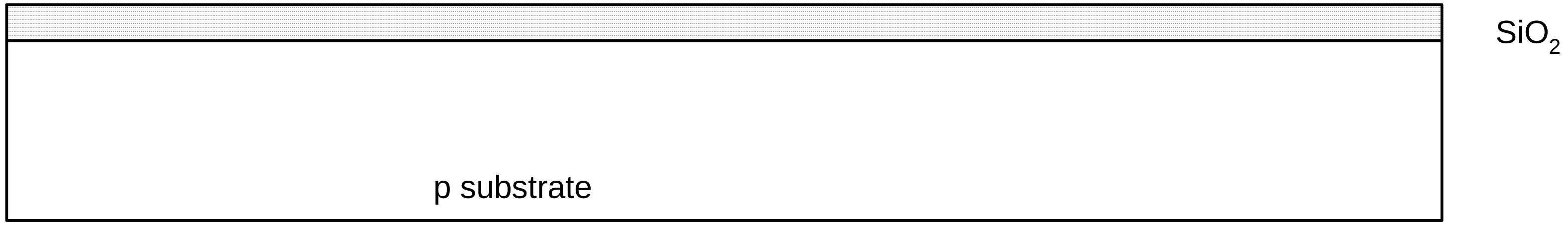
Basic Fabrication Steps

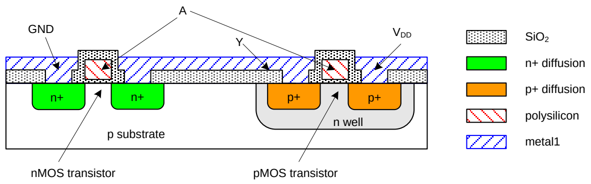
- **Start with a blank wafer**



Basic Fabrication Steps

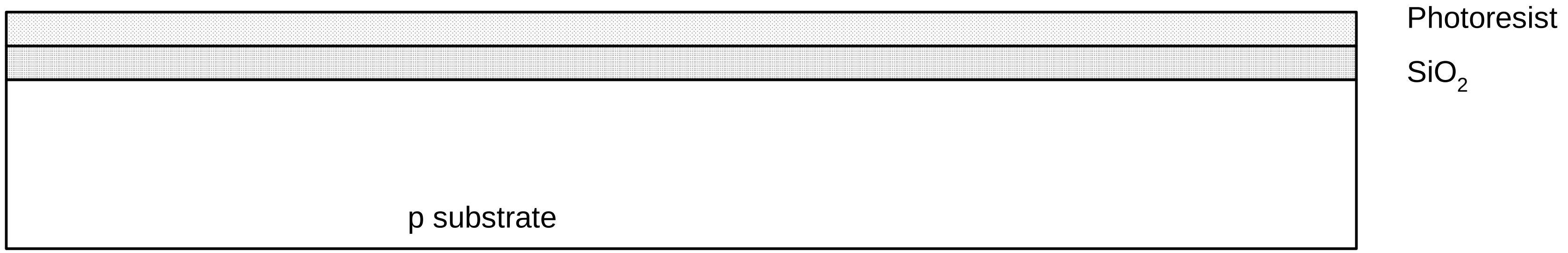
- Start with a blank wafer
- **Grow SiO_2 on top of Si wafer**

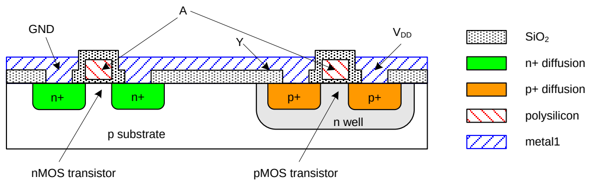




Basic Fabrication Steps: N-Well

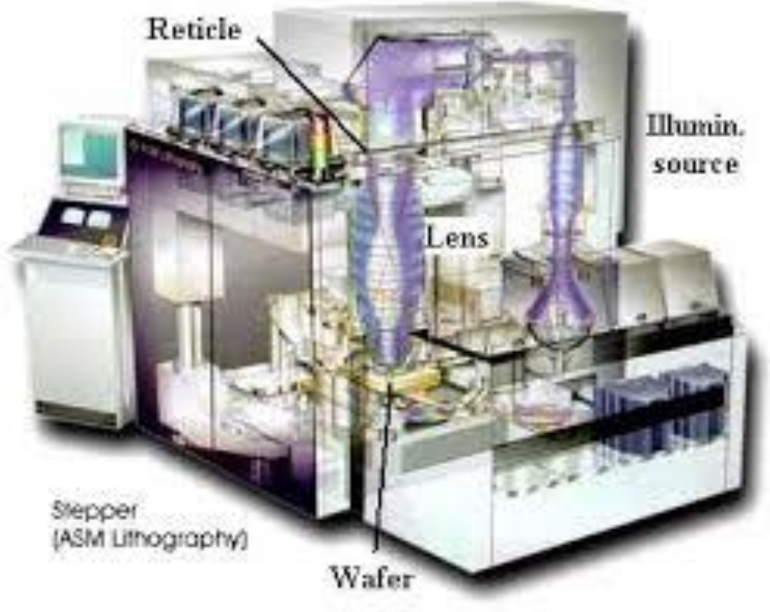
- Start with a blank wafer
- Grow SiO_2 on top of Si wafer
- **Spin on photoresist** (light sensitive organic polymer)
    - Hardens or softens when exposed to light

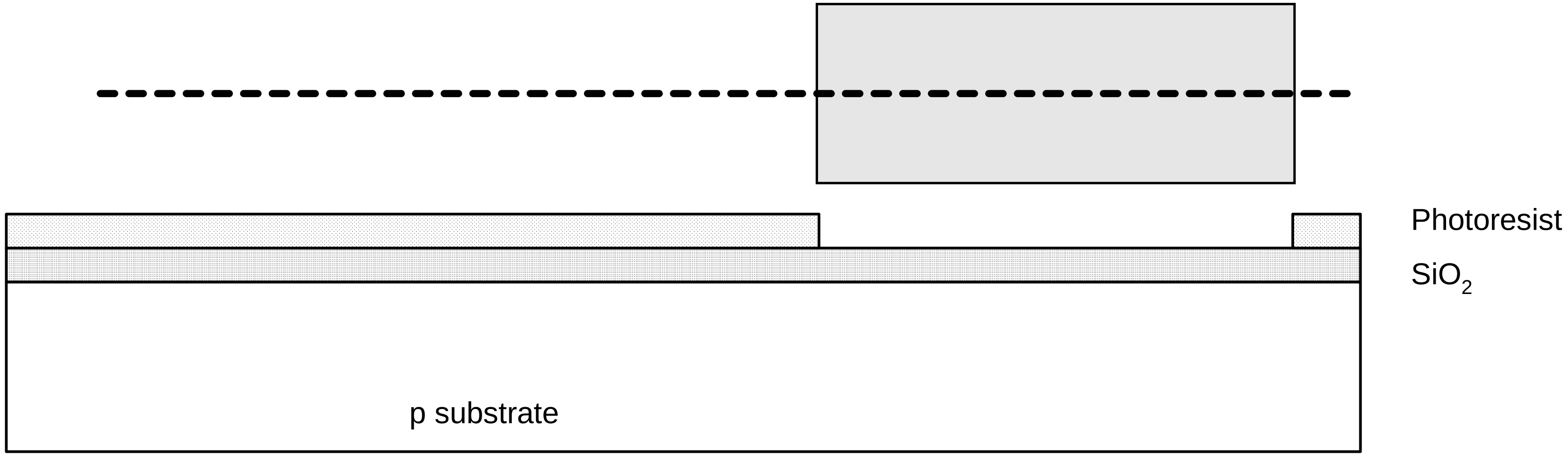


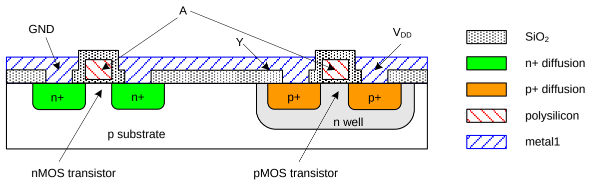


Basic Fabrication Steps: N-Well

- Start with a blank wafer
- Grow SiO_2 on top of Si wafer
- Spin on photoresist (light sensitive organic polymer)
  - Hardens or softens when exposed to light
- **Expose photoresist through n-well mask**
  - Stepper: replicate image across wafer (limited exposure area)
- **Strip off exposed photoresist**

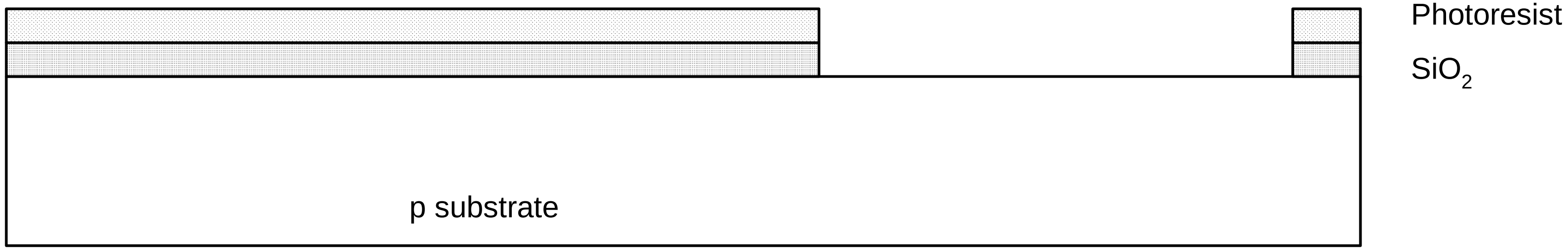


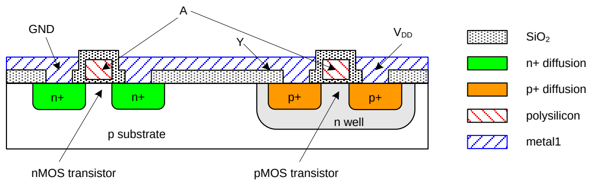




Basic Fabrication Steps: N-Well

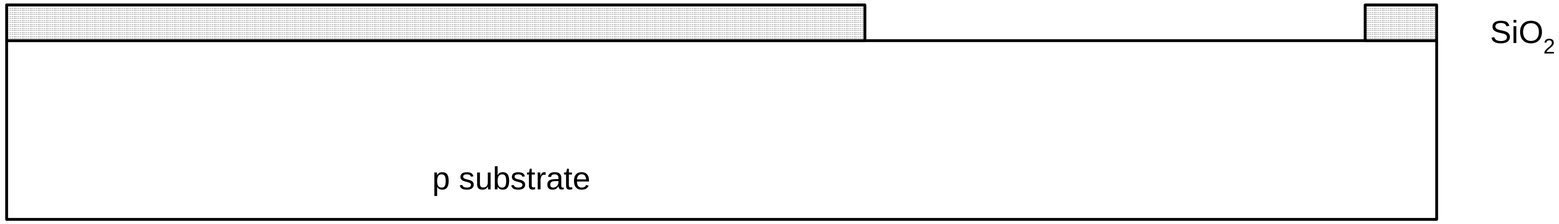
- **Etch oxide with hydrofluoric acid (HF)**
    - Only attacks oxide where resist has been removed

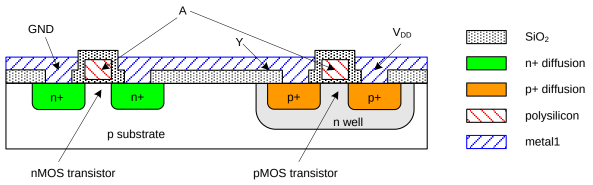




Basic Fabrication Steps: N-Well

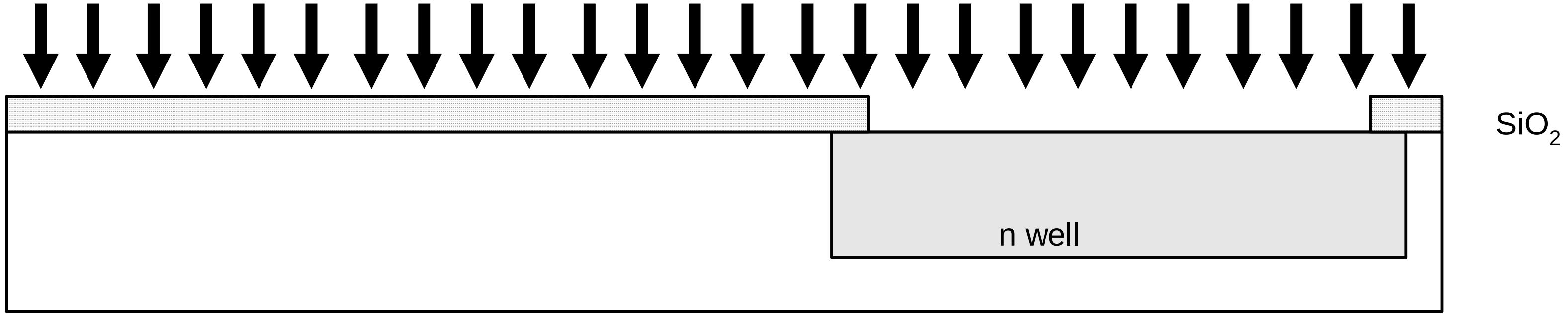
- Etch oxide with hydrofluoric acid (HF)
    - Only attacks oxide where resist has been removed
- **Strip off remaining photoresist** with mixture of acids called piranah

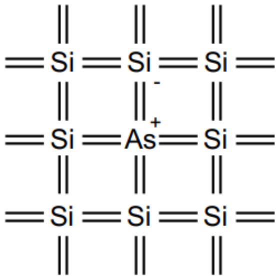




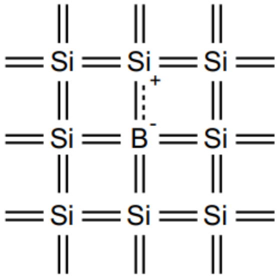
Basic Fabrication Steps: N-Well

- Etch oxide with hydrofluoric acid (HF)
  - Only attacks oxide where resist has been removed
- Strip off remaining photoresist with mixture of acids called piranah
- **n-well is formed with diffusion or ion implantation**
  - Diffusion: Heat wafer in furnace with arsenic gas until As atoms diffuse into exposed Si
  - **Ion Implanatation:** Blast wafer with beam of As ions (blocked by SiO_2, only enter exposed Si)

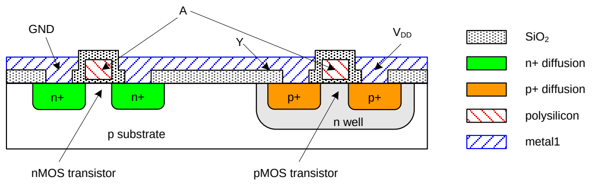




Group V element to create n-type

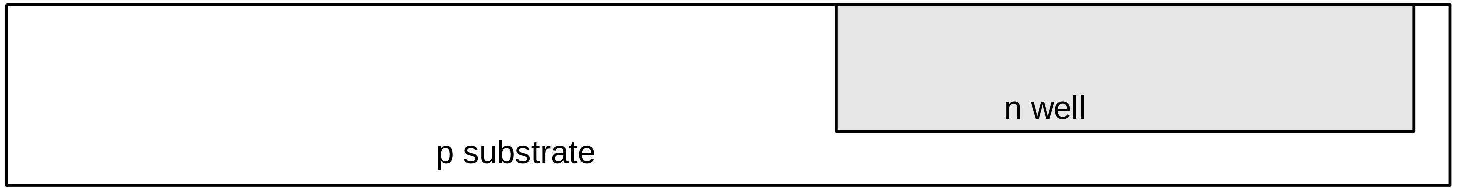


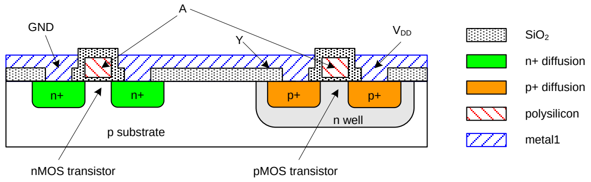
Group III element to create p-type



Basic Fabrication Steps: N-Well

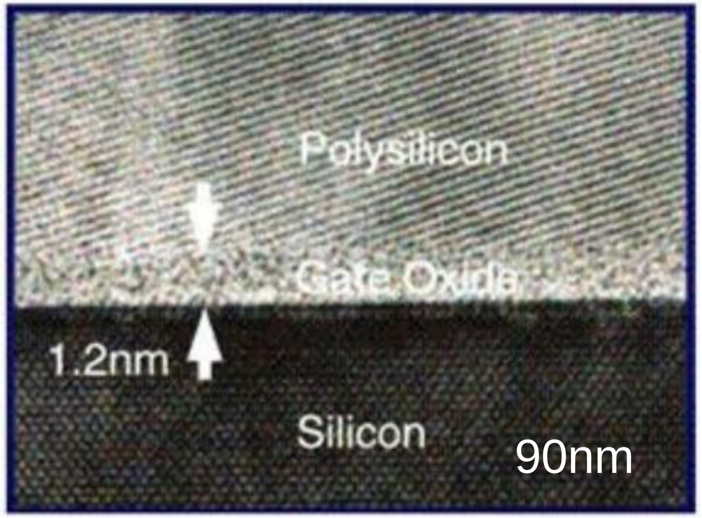
- Etch oxide with hydrofluoric acid (HF)
  - Only attacks oxide where resist has been removed
- Strip off remaining photoresist with mixture of acids called piranah
- n-well is formed with diffusion or ion implantation
  - Diffusion: Heat wafer in furnace with arsenic gas until As atoms diffuse into exposed Si
  - Ion Implanatation: Blast wafer with beam of As ions (blocked by SiO_2, only enter exposed Si)
- **Strip off the remaining oxide using hydrofluoric acid**
  - Back to bare wafer with n-well

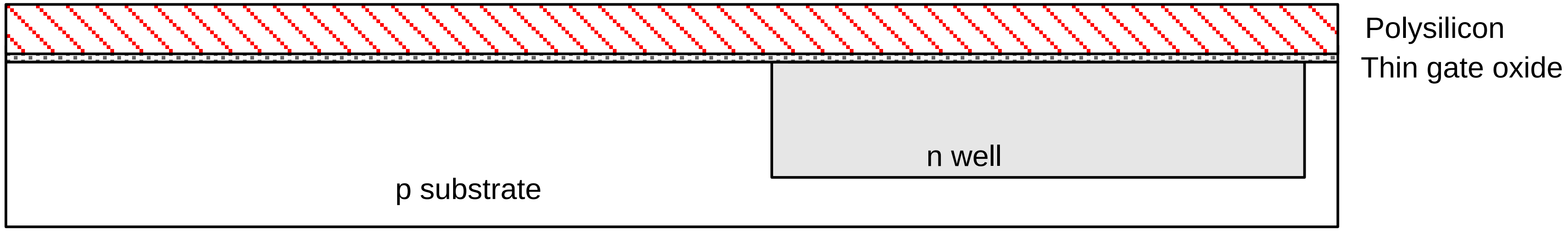


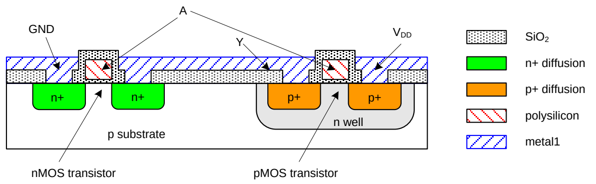


Basic Fabrication Steps: Gate Oxide & Polysilicon (Gate)

- **Deposit very thin layer of gate oxide**
  - 45nm technology has a 1.2nm thick layer (about 5 atoms!)
- **Chemical Vapor Deposition (CVD) of silicon layer**
  - Heavily doped to be good conductor

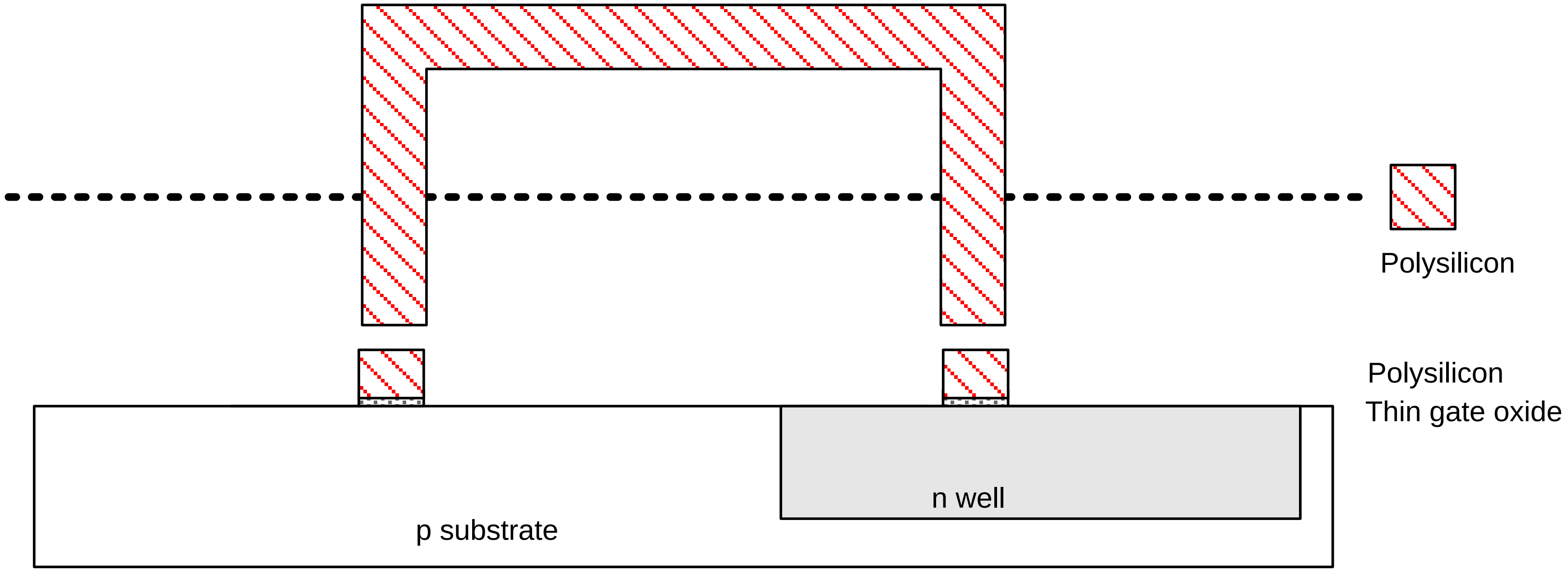


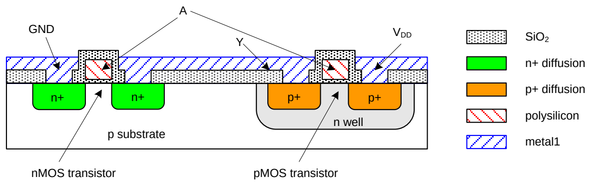




Basic Fabrication Steps: Polysilicon (Gates)

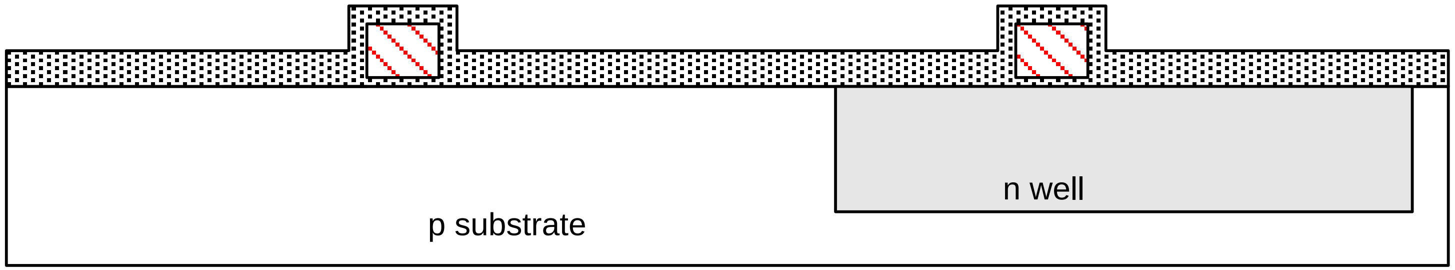
- Deposit very thin layer of gate oxide
  - 45nm technology has a 1.2nm thick layer (about 5 atoms!)
- Chemical Vapor Deposition (CVD) of silicon layer
  - Heavily doped to be good conductor
- **Use same lithography process to pattern polysilicon**

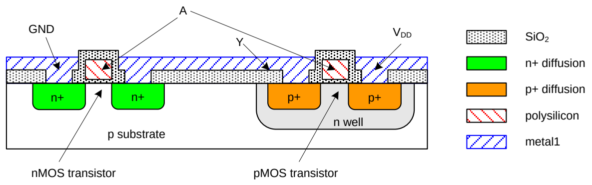




Basic Fabrication Steps: N-Diffusion

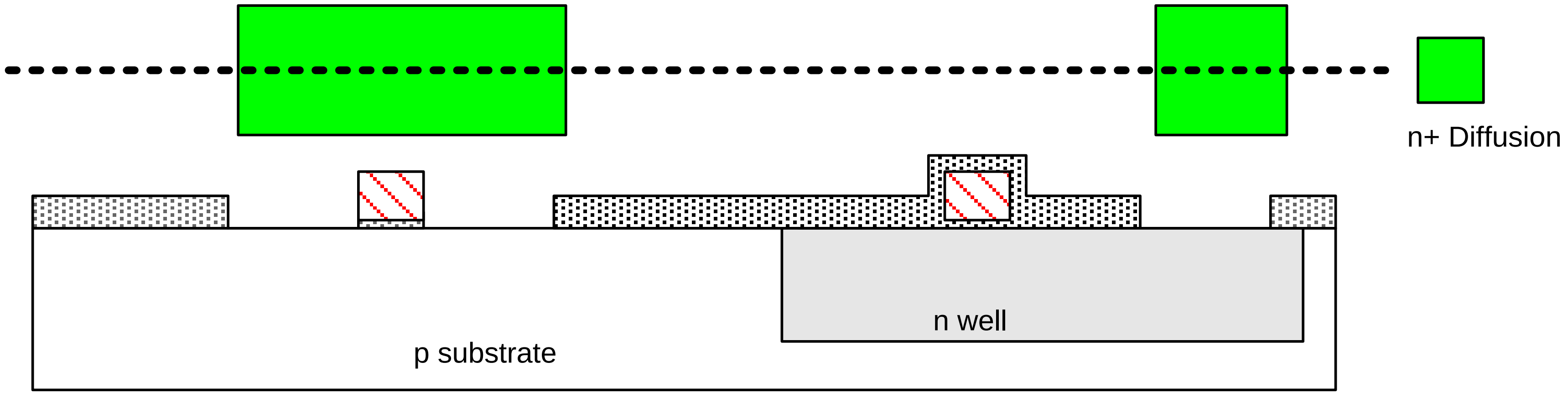
- **N-diffusion forms nMOS source, drain, and n-well contact**
- **Use masking to expose where n+ dopants should be diffused or implanted**

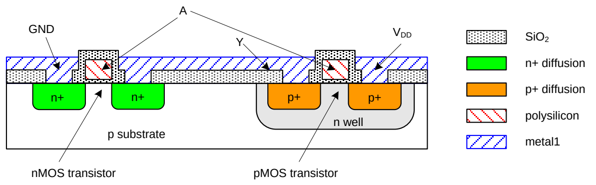




Basic Fabrication Steps: N-Diffusion

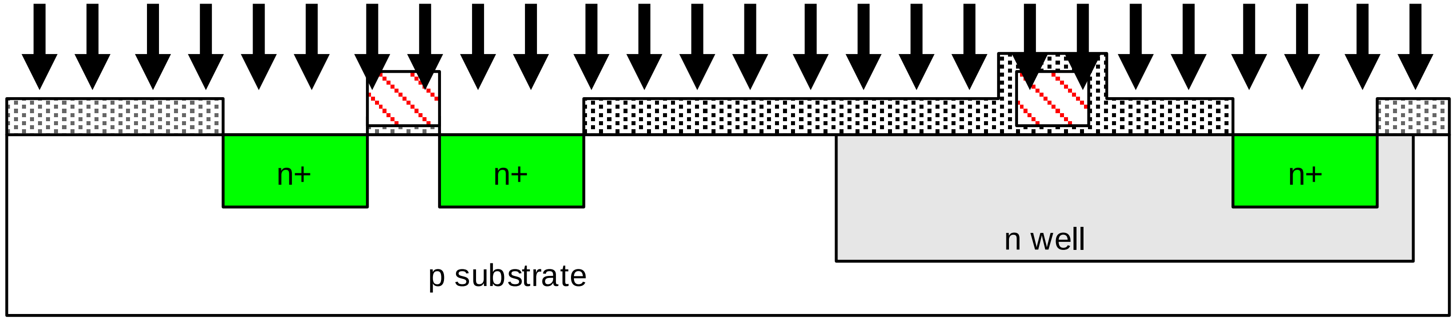
- **N-diffusion forms nMOS source, drain, and n-well contact**
- Use masking to expose where n+ dopants should be diffused or implanted
  - **Expose where n+ dopants should be diffused or implanted**
- **Self-aligned process for forming the diffusion regions**
  - Self-aligned=use oxide as additional “mask” to protect the gate regions from implants

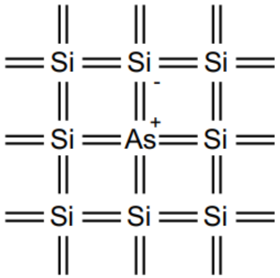




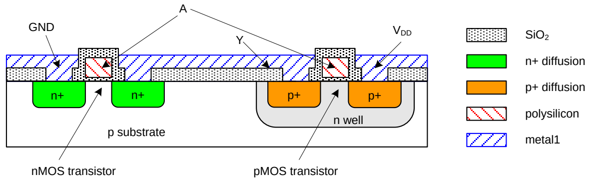
Basic Fabrication Steps: N-Diffusion

- **N-diffusion forms nMOS source, drain, and n-well contact**
- Use masking to expose where n+ dopants should be diffused or implanted
  - Expose where n+ dopants should be diffused or implanted
- Self-aligned process for forming the diffusion regions
  - Self-aligned=use oxide as additional “mask” to protect the gate regions from implants
- **“Diffusion” usually done today by Ion Implantation**



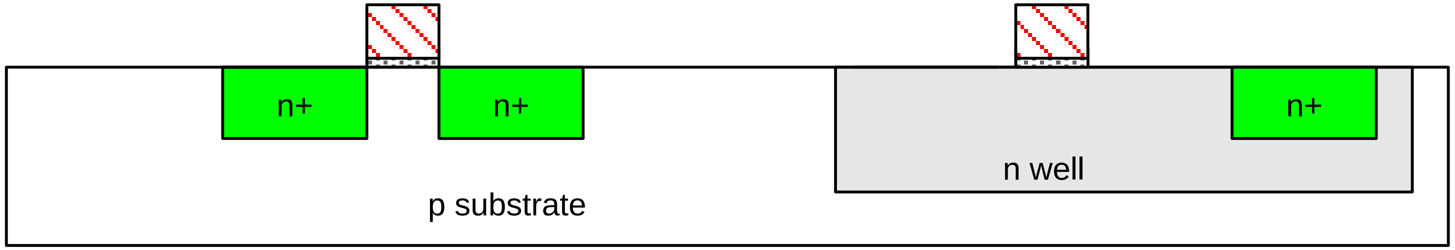


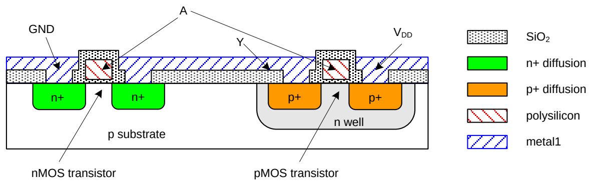
Group V element
to create n-type



Basic Fabrication Steps: N-Diffusion

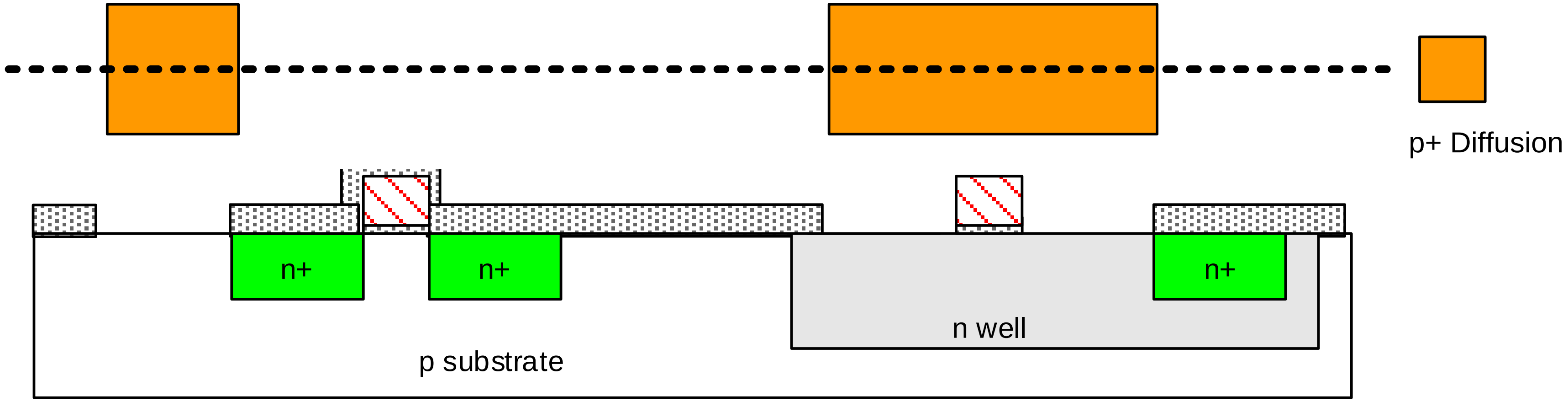
- **N-diffusion forms nMOS source, drain, and n-well contact**
- Use masking to expose where n+ dopants should be diffused or implanted
  - Expose where n+ dopants should be diffused or implanted
- Self-aligned process for forming the diffusion regions
  - Use oxide as additional “mask” to protect the gate regions from implants
- “Diffusion” usually done today by Ion Implantation
- **Strip off oxide to complete patterning step**

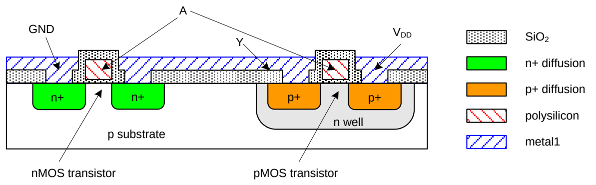




Basic Fabrication Steps: P-Diffusion

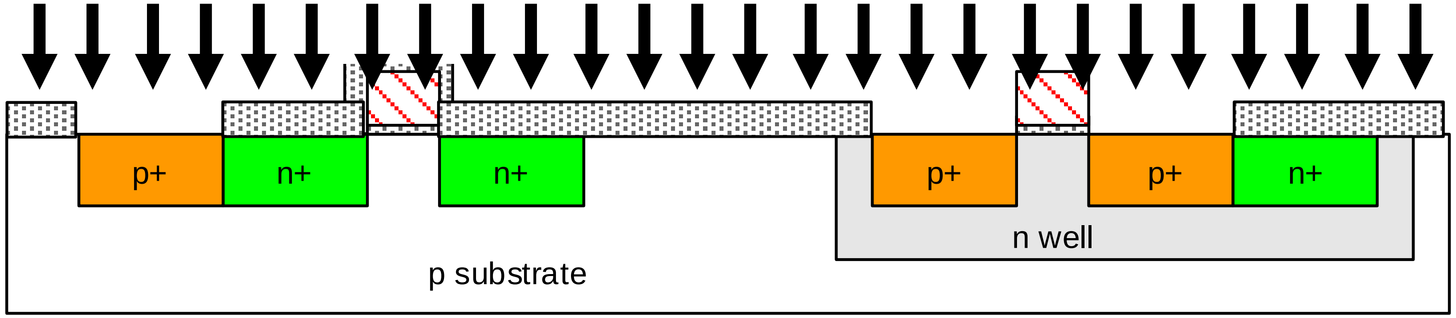
- **Similar set of steps form p+ diffusion regions for pMOS source and drain and p-substrate contacts**
- Negative photomask: exposed regions are protected from implants

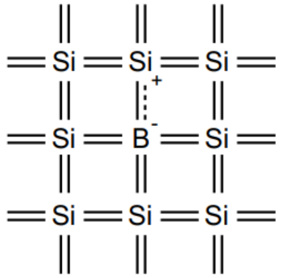




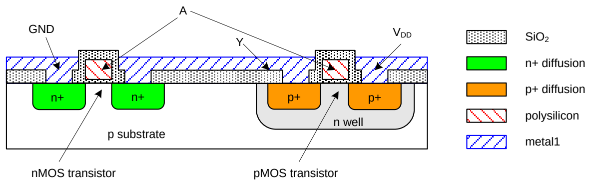
Basic Fabrication Steps: P-Diffusion

- **Similar set of steps form p+ diffusion regions for pMOS source and drain and p-substrate contacts**
- Negative photomask: exposed regions are protected from implants



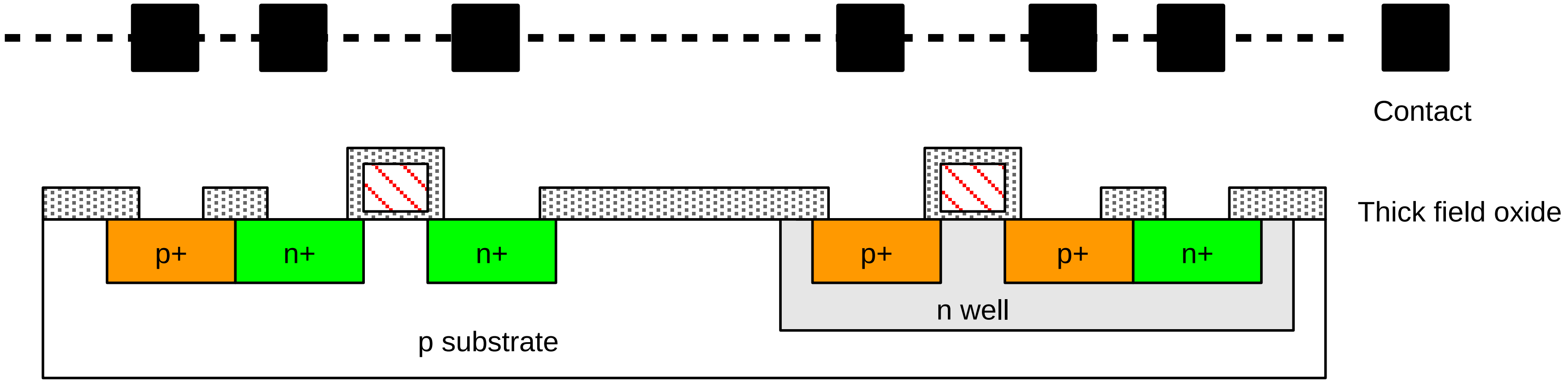


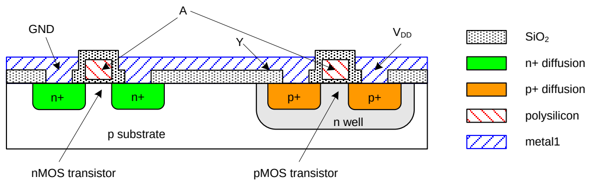
Group III element
to create p-type



Basic Fabrication Steps: Contacts

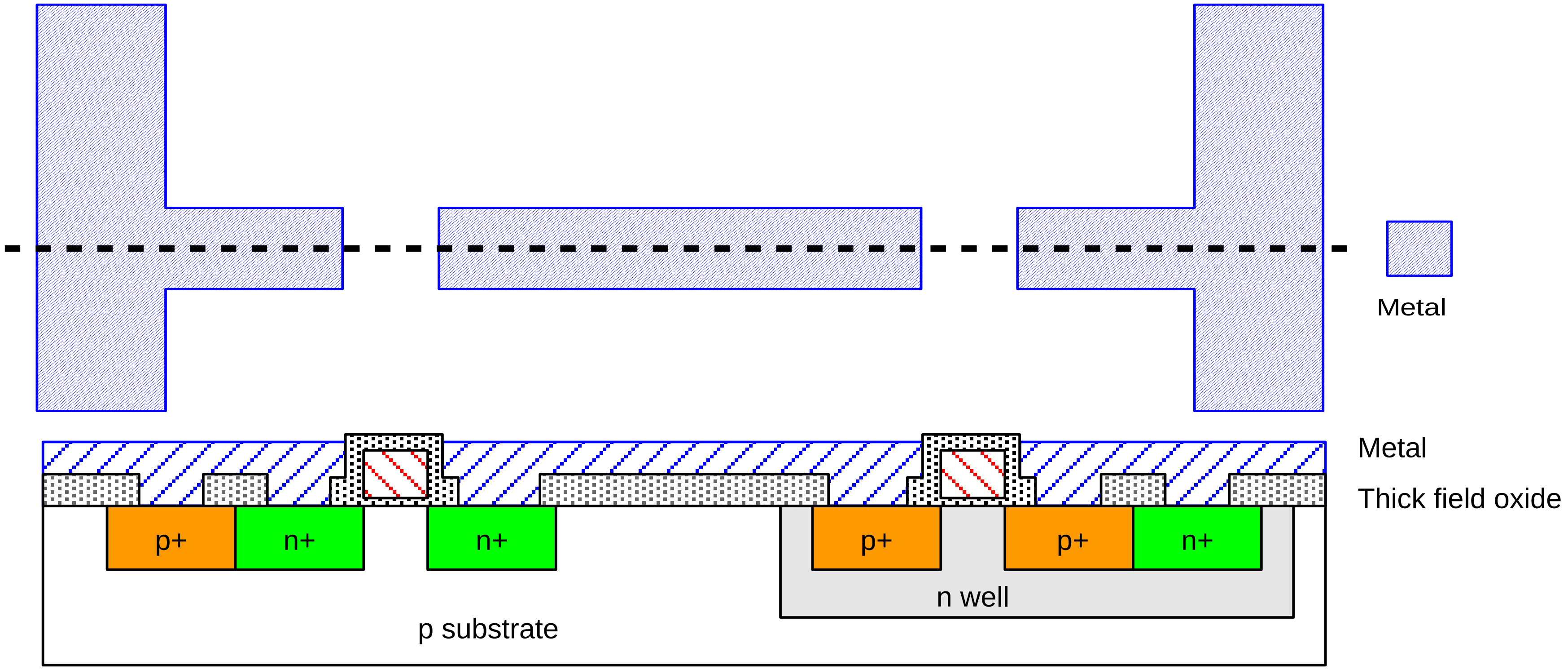
- **Now we need to wire together the devices**
- **Cover chip with thick field oxide**
- **Etch oxide where contact cuts are needed**

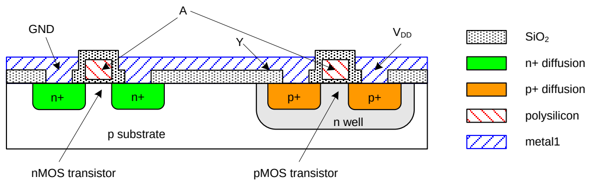




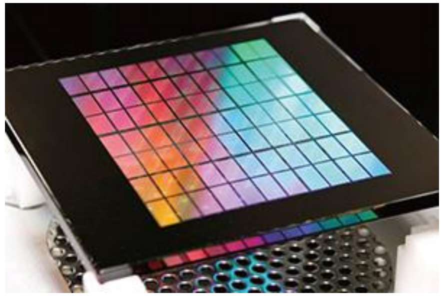
Basic Fabrication Steps: Metalization

- **Sputter on aluminum over whole wafer**
- **Pattern to remove excess metal, leaving wires**

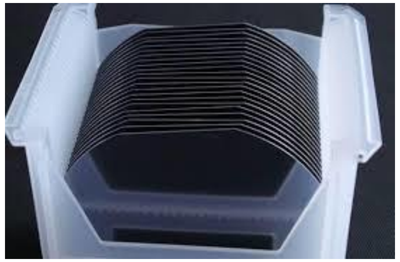


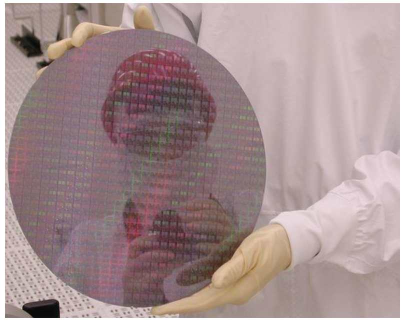


Silicon Fab - Manufacturing



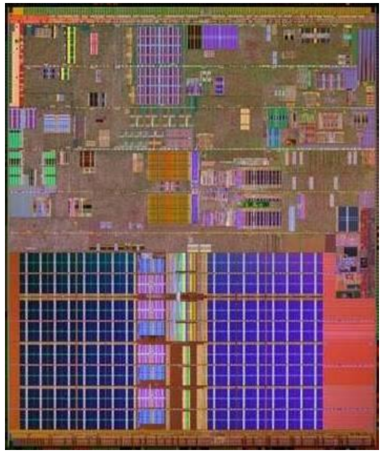
A fab costs
4-5 Billion USD

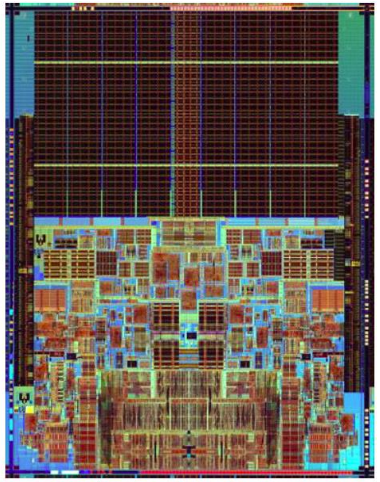




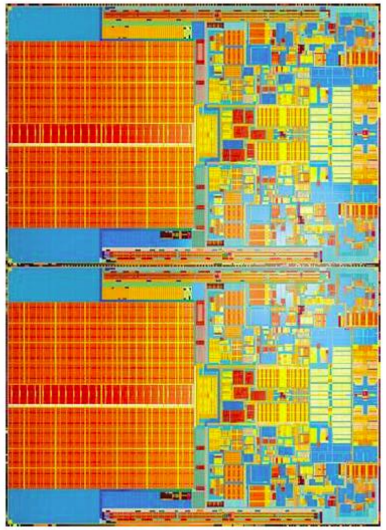
Fabricated Silicon
Price/g = 70 USD/g
@8k USD/wafer

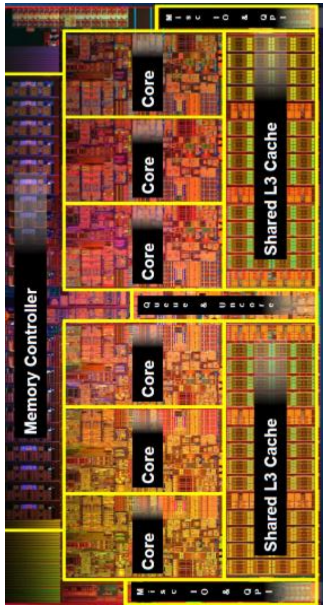
Technology evolution as a driver

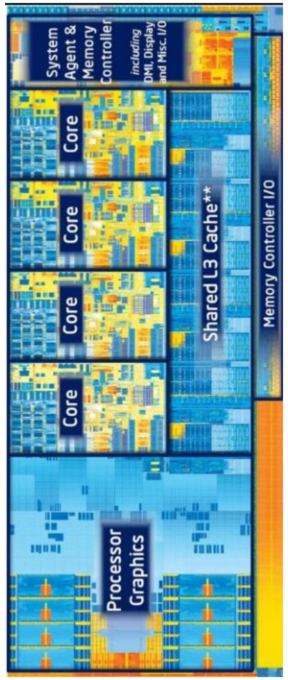

170M

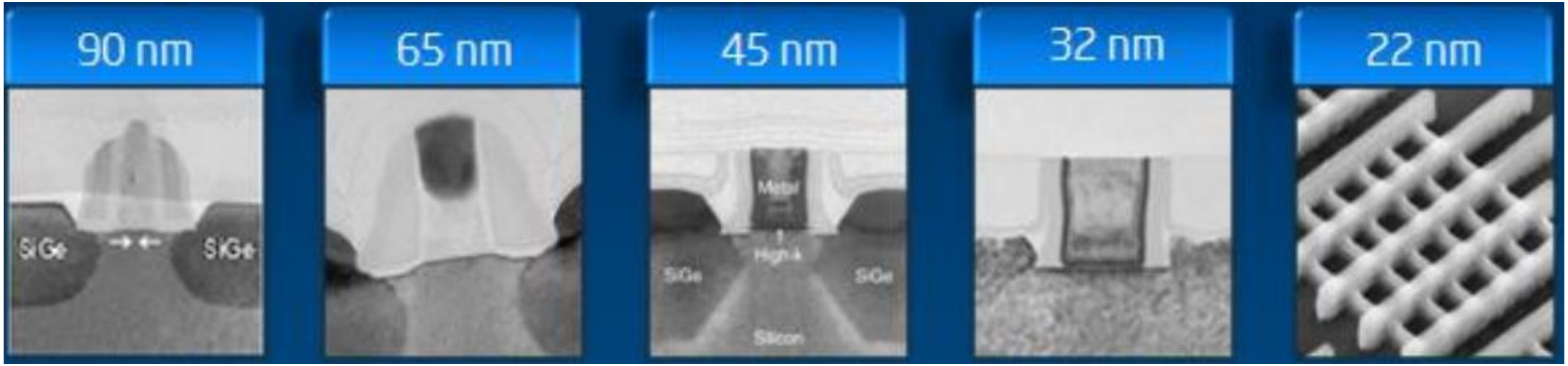

291M

http://www.sudhian.com/img/intel/core2/core2.03.jpg


820M

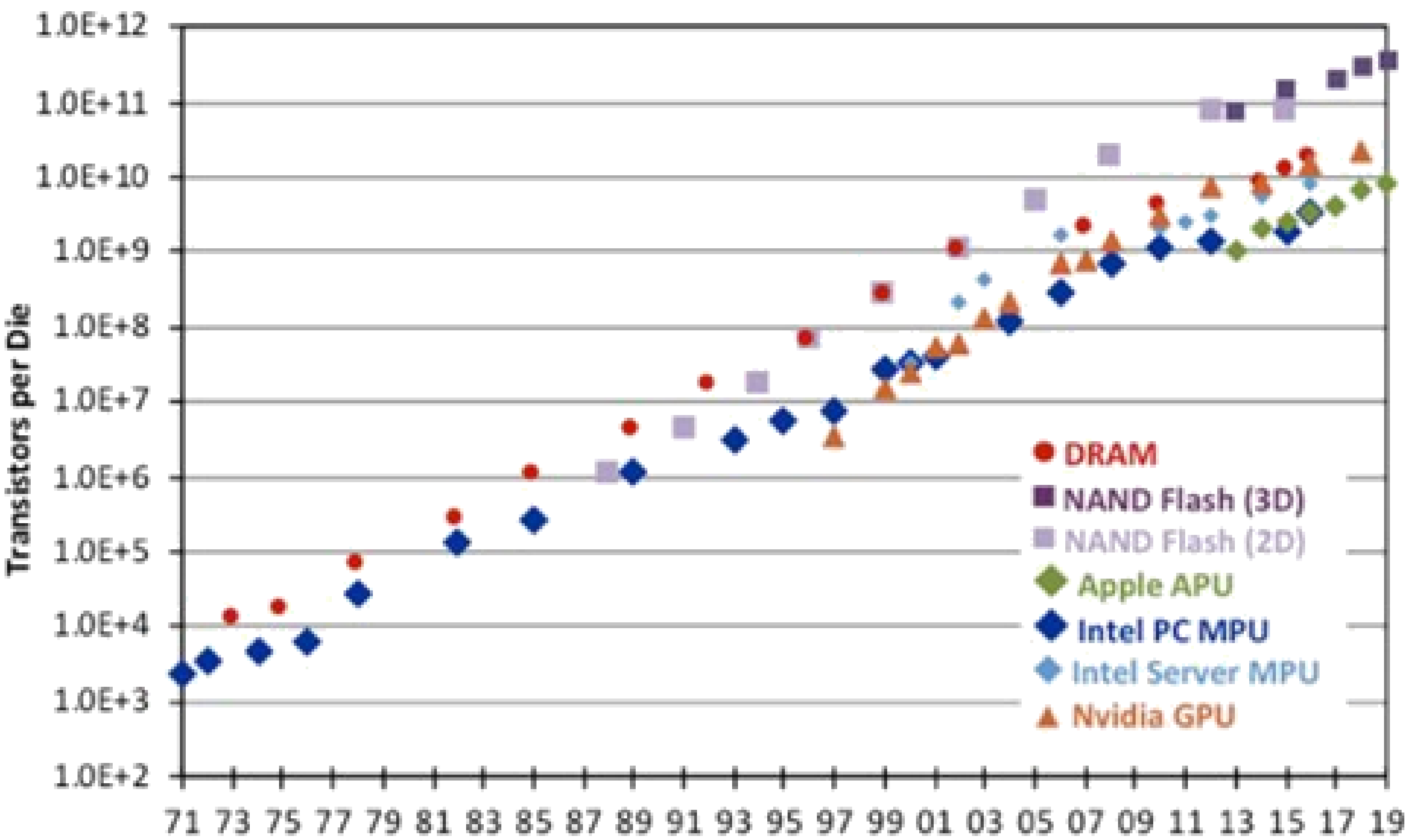

1.2B


1.8B



Lets check Moore's Law …

- Today, we have **many different types of integrated circuits**

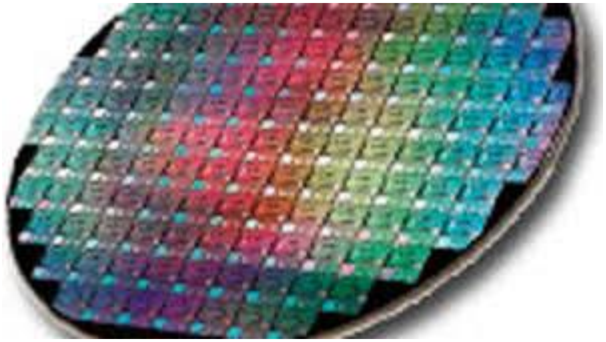


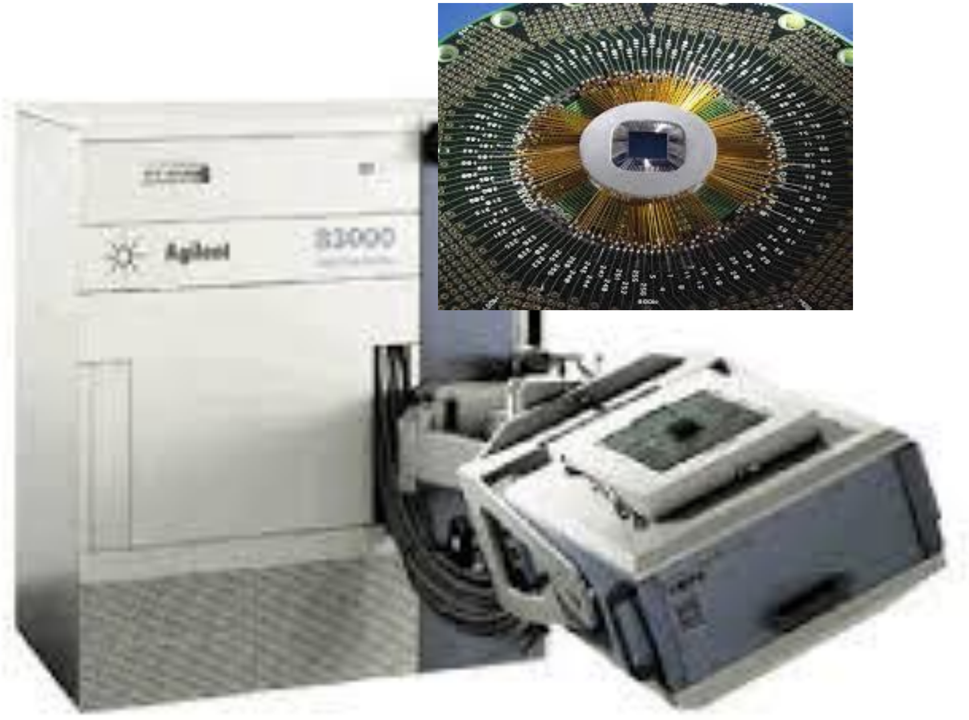
Sources: Intel, SIA, Wikichip, IC Insights

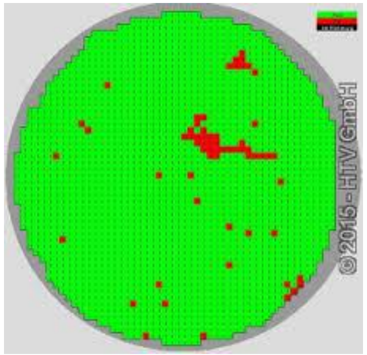
Some growth-rate variation depending on the field, but **generally still 40-60% per year**

Testing

- **Not all chips work as expected…**
- **Testing discards broken or not well performing chips**



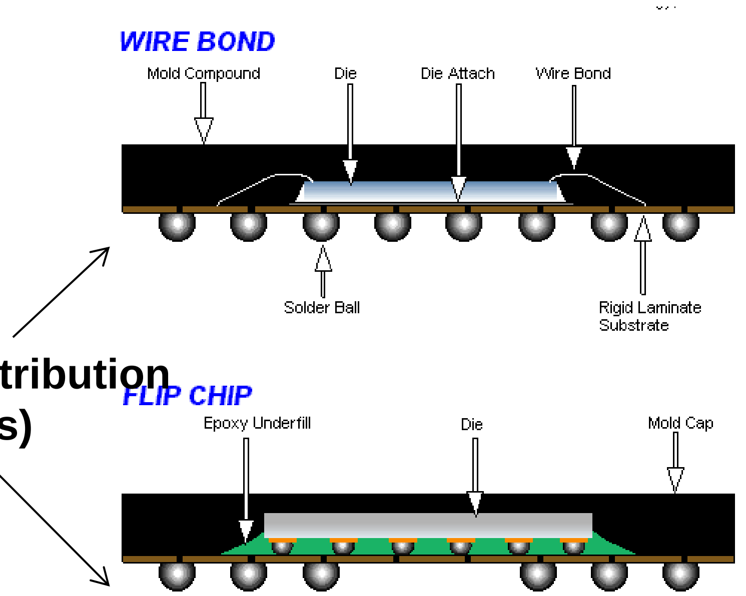




https://www.youtube.com/watch?v=FnLNBkxZFR8

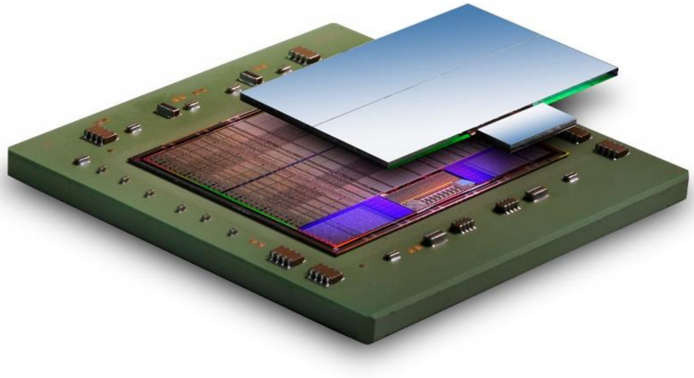
Technologies for Integration

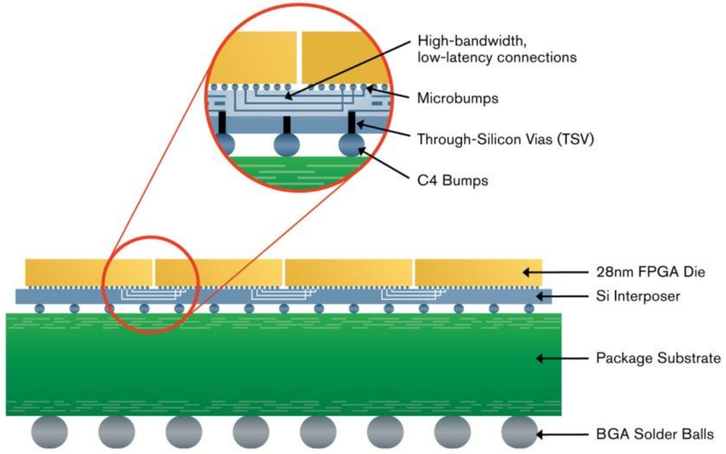
- **Package**
  - Houses and protects the chip
  - Spread the pins of a chip to a larger grid that can connect to a printed circuit board
  - Provides heat sink
  - Made of ceramic or plastic



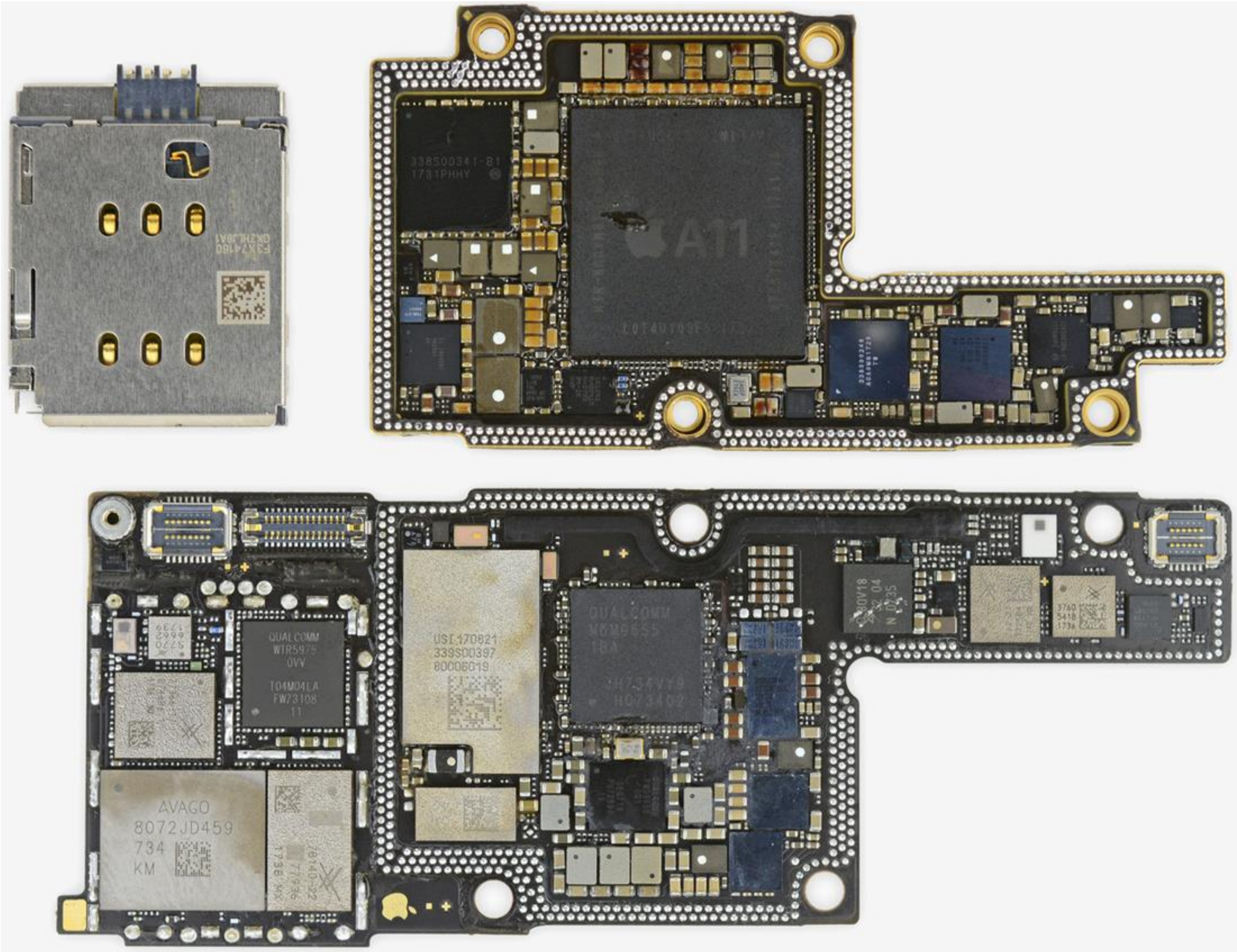
Redistribution layer(s)

- **Multichip Module & Silicon Interposers (2.5D)**
  - Multiple chips mounted directly on a common substrate
  - Connectivity and interface to PCB
  - High level of integration
  - Connections with short wires and high density





The END



VLSI Performance Metrics

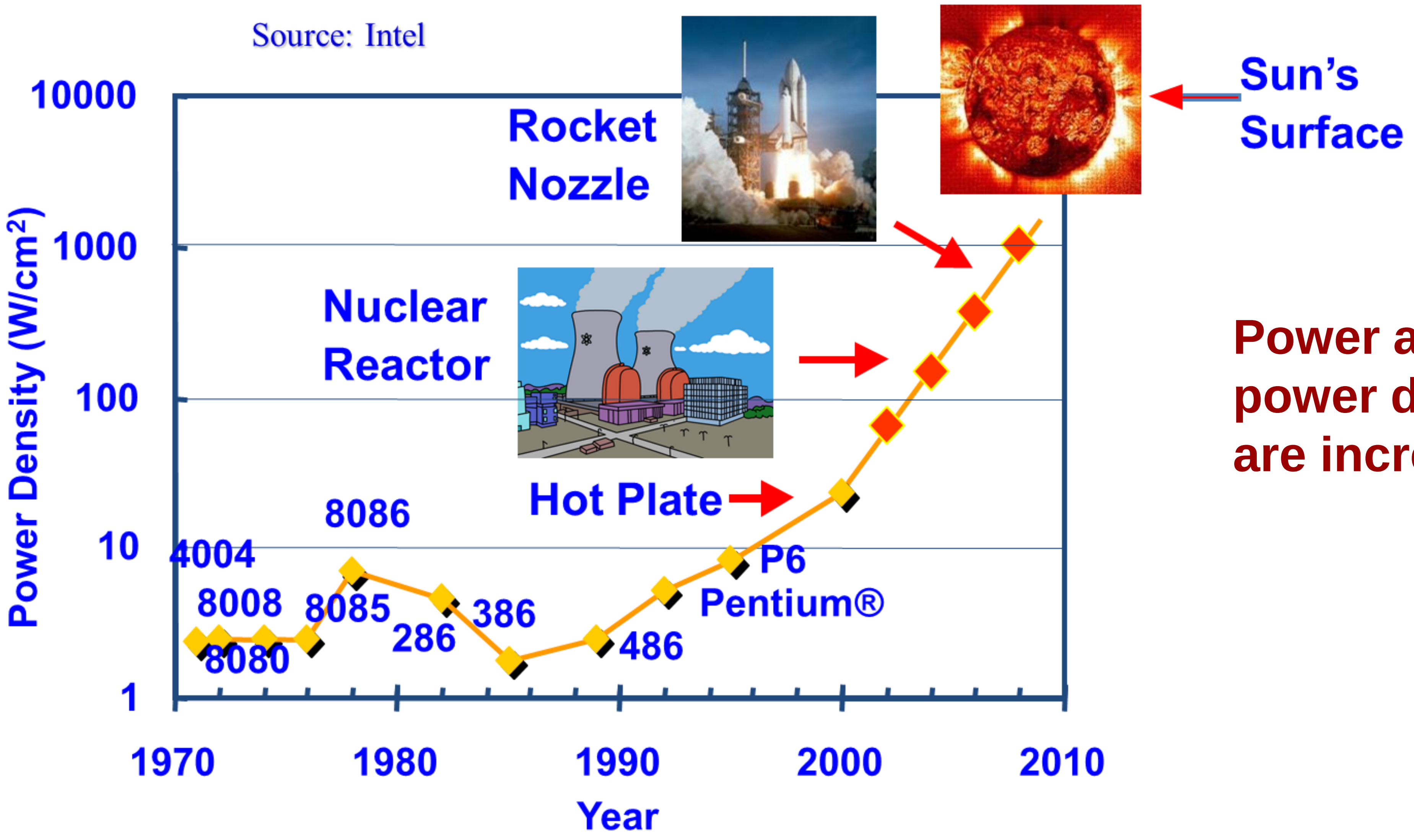
- In **VLSI** design **involves more than one performance metric**:

COST

Density/cost **Power** **Speed** **Reliability**

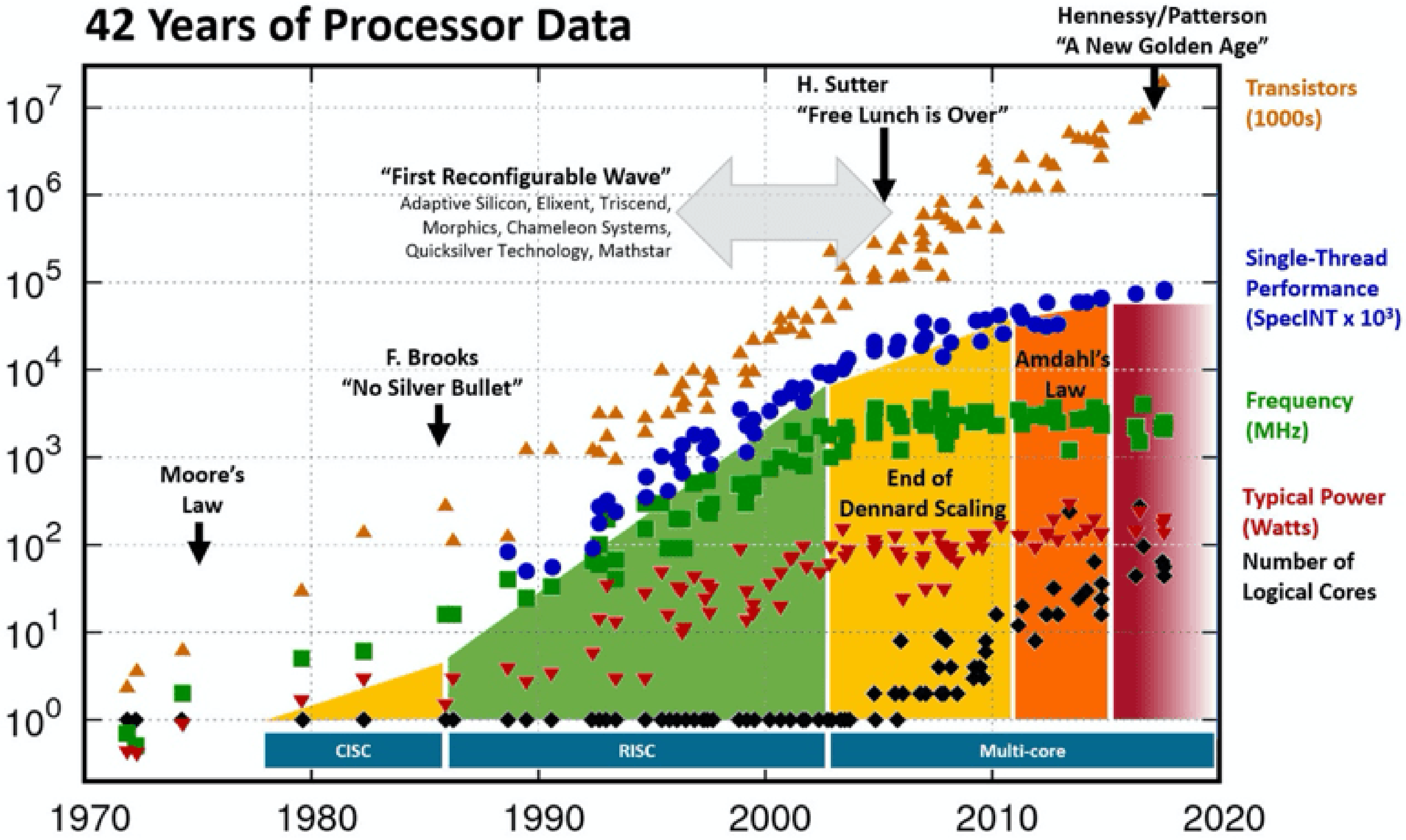
- **The "NO FREE LUNCH THEOREM": Improving on one metric generally involves a penalty on another metrics**
- For every circuit, we have to **evaluate** the **specifications** and **choose the right trade-off** that meet any potential hard constraints on some metrics

Moore’s law on other metrics… ☹



Power and power density are increasing

Moore’s law on other metrics… ☹

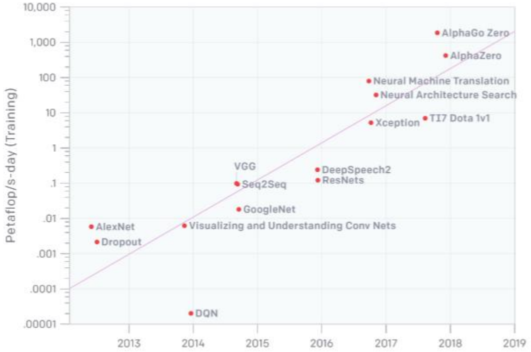


Hennessy and Patterson, Turing Lecture 2018, overlaid over “42 Years of Processors Data”
https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/;“First Wave” added by Les Wilson, Frank Schirrmeister
Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

Speed is saturating

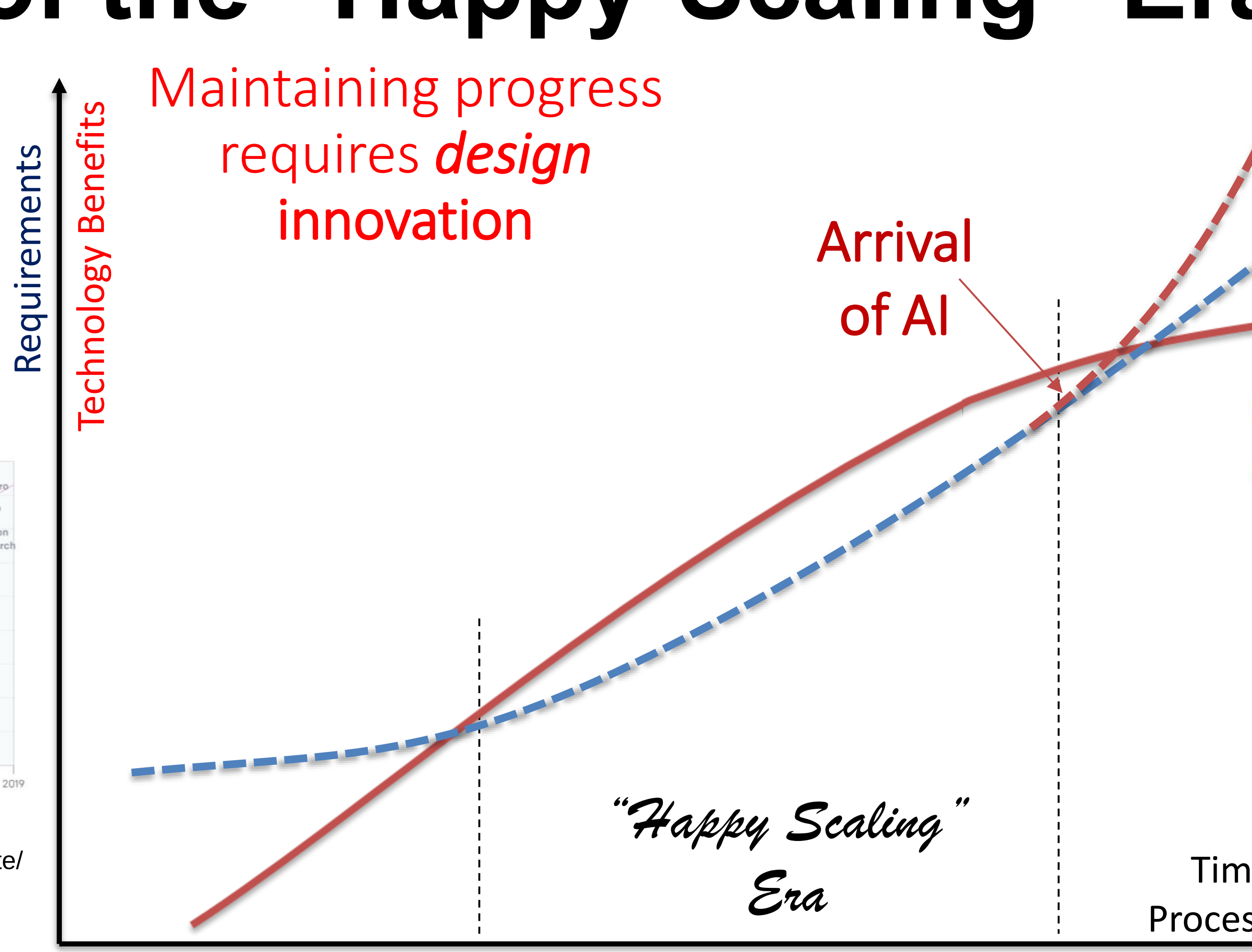
The End of the "Happy-Scaling" Era

New Applications: AI



Increasing complexity
https://blog.openai.com/ai-and-compute/

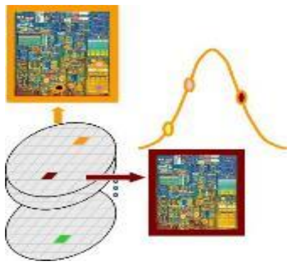
More processing in a reduced power envelope



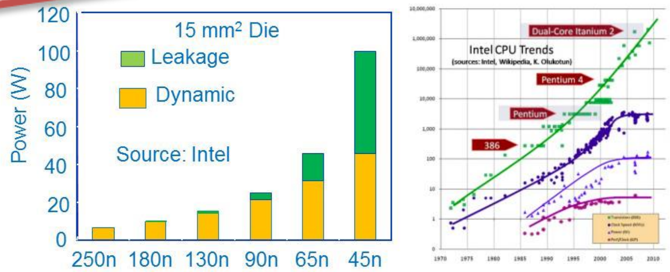
Maintaining progress requires ***design* innovation**

Arrival of AI

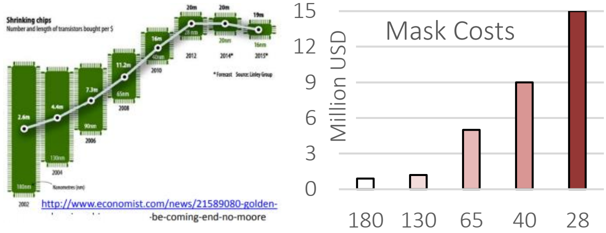
"Happy Scaling" Era



Severe variability and reliability issues



Vanishing energy and performance benefits



http://www.economist.com/news/21589080-golden-be-coming-end-no-moore

Skyrocketing costs

90 nm

28 nm

WORK!

DUE TO THE CURRENT WORKLOAD, THE LIGHT AT THE END OF THE TUNNEL HAS BEEN SWITCHED OFF

Challenges ahead …

- **Moore's law continues**, but keeping it up **and living with its consequences becomes increasingly difficult**

∝ DSM

"Microscopic Problems"

- Ultra-high speed design
- Interconnect
- Noise, Crosstalk
- Reliability, Manufacturability
- Power Dissipation
- Clock distribution.

Everything Looks a Little Different

∝ 1/DSM

"Macroscopic Issues"

- Time-to-Market
- Millions of Gates
- High-Level Abstractions Needed
- Reuse & IP: Portability
- Predictability
- etc.

…and There's a Lot of Them!

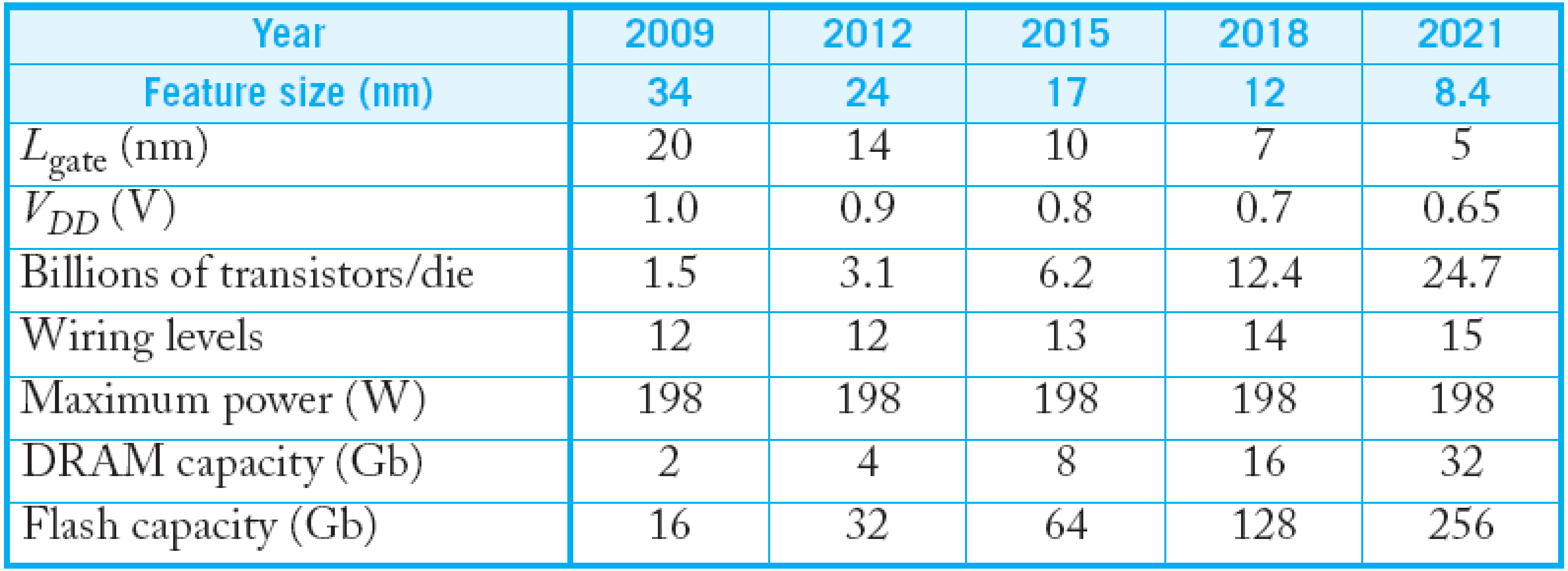
EE-429
Fundamentals of VLSI Design

Future Trends

Andreas Burg

The Crystal Ball of Semiconductor: ITRS

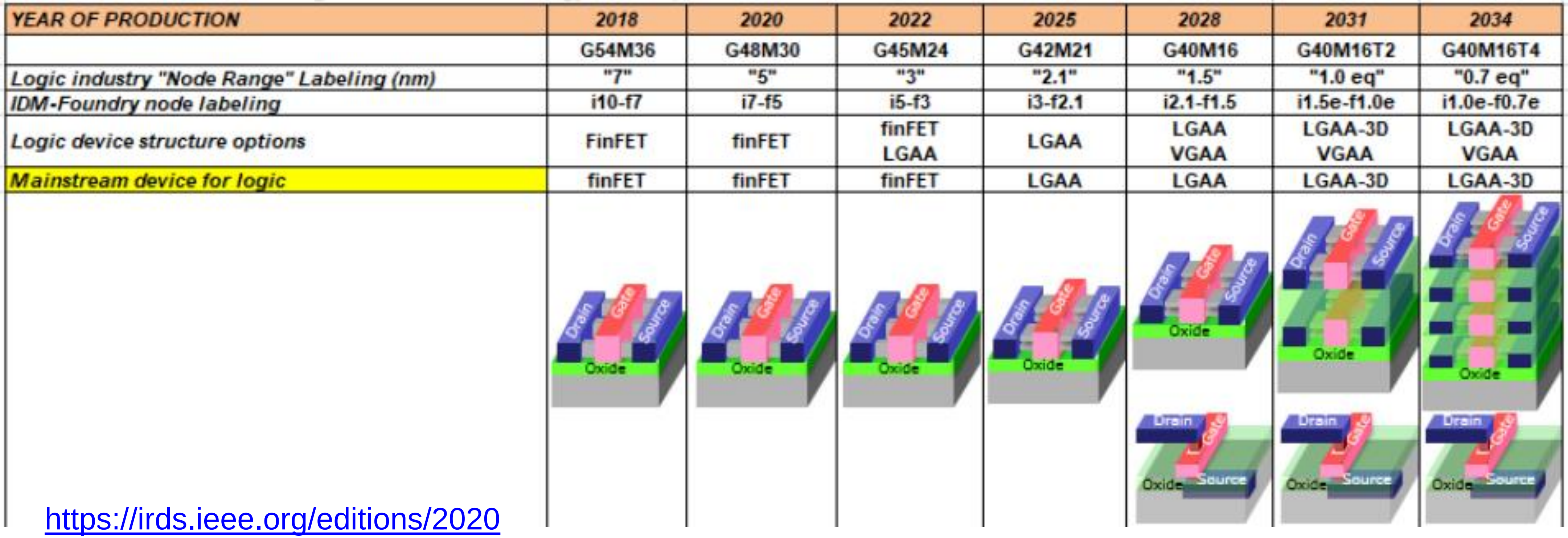
- **ITRS: International Technology Roadmap for Semiconductor**
  - Industry consortium founded in 1998 to predict the future of the semiconductor industry
  - Published a bi-annual report that predicted the future trends in various topics around the semiconductor industry



Year	2009	2012	2015	2018	2021
Feature size (nm)	34	24	17	12	8.4
L_{gate} (nm)	20	14	10	7	5
V_{DD} (V)	1.0	0.9	0.8	0.7	0.65
Billions of transistors/die	1.5	3.1	6.2	12.4	24.7
Wiring levels	12	12	13	14	15
Maximum power (W)	198	198	198	198	198
DRAM capacity (Gb)	2	4	8	16	32
Flash capacity (Gb)	16	32	64	128	256

The ITRS Successor

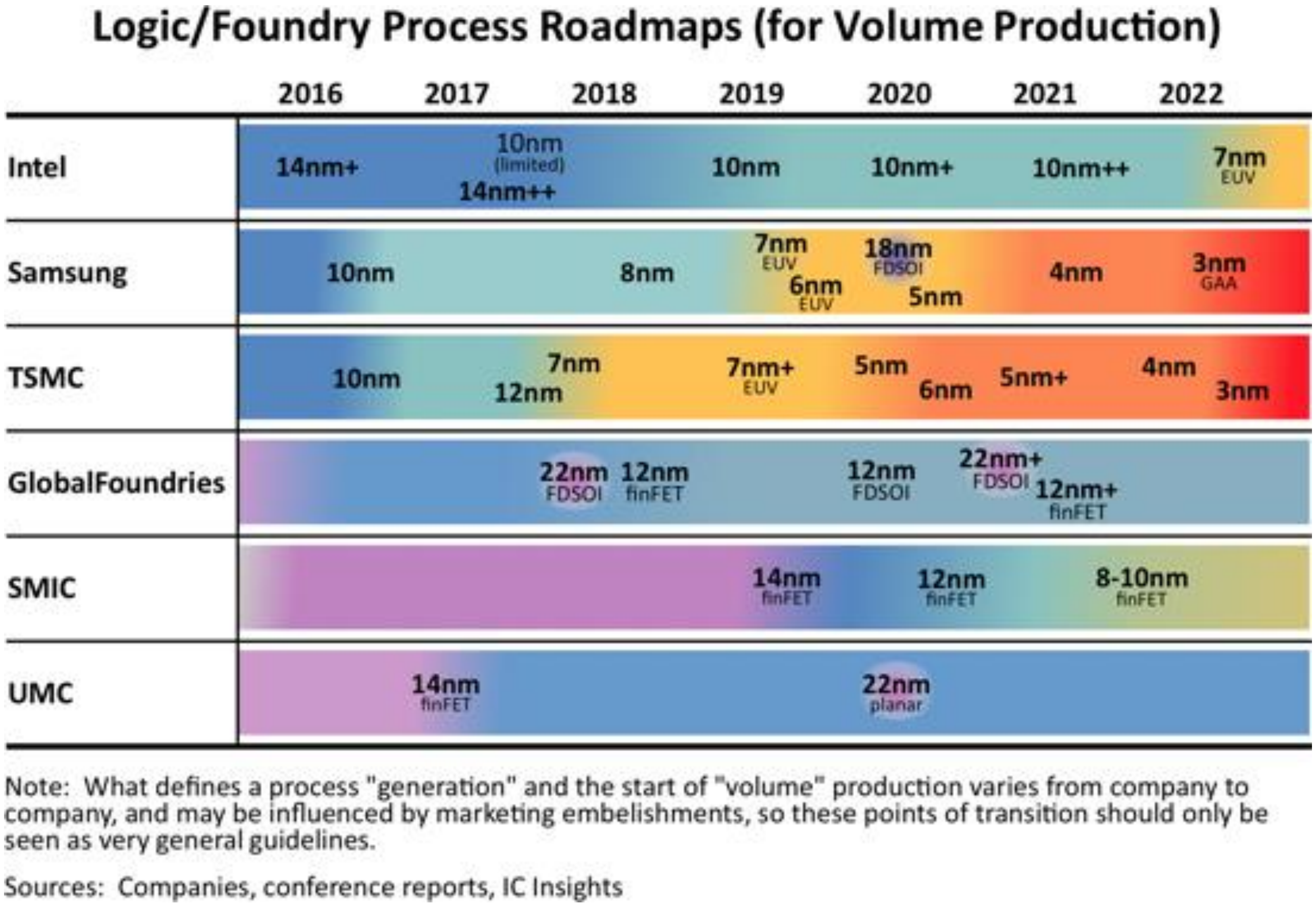
- **The ITRS was upgraded in 2014 to the ITRS 2.0 split into working groups**
  - Motivation: reflect better the changes in the industry to new technologies and toward systems
- **In 2014, ITRS was "retired"** and **replaced** in 2016 **by the "International Roadmap for Devices and Systems" (IRDS) to reflect the system focus**



YEAR OF PRODUCTION	*2018*	*2020*	*2022*	*2025*	*2028*	*2031*	*2034*
	G54M36	G48M30	G45M24	G42M21	G40M16	G40M16T2	G40M16T4
Logic industry "Node Range" Labeling (nm)	"7"	"5"	"3"	"2.1"	"1.5"	"1.0 eq"	"0.7 eq"
IDM-Foundry node labeling	i10-f7	i7-f5	i5-f3	i3-f2.1	i2.1-f1.5	i1.5e-f1.0e	i1.0e-f0.7e
Logic device structure options	FinFET	finFET	finFET LGAA	LGAA	LGAA VGAA	LGAA-3D VGAA	LGAA-3D VGAA
Mainstream device for logic	finFET	finFET	finFET	LGAA	LGAA	LGAA-3D	LGAA-3D

https://irds.ieee.org/editions/2020

Foundry Roadmaps in Times of Consolidation



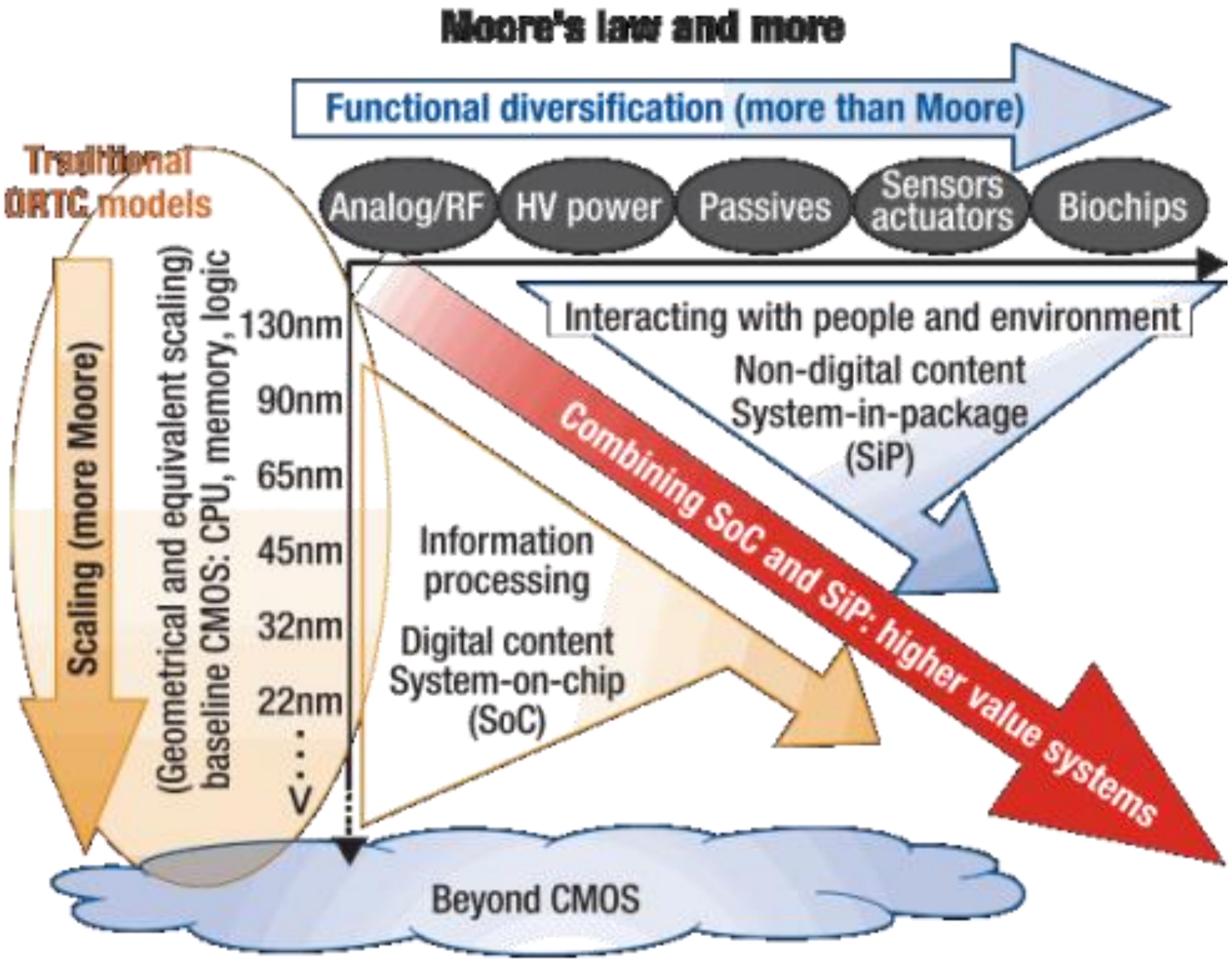
Logic/Foundry Process Roadmaps (for Volume Production)

	2016	2017	2018	2019	2020	2021	2022
Intel	14nm+	10nm (limited) 14nm++		10nm	10nm+	10nm++	7nm EUV
Samsung	10nm		8nm	7nm EUV 6nm EUV	18nm FDSOI 5nm	4nm	3nm GAA
TSMC	10nm	12nm	7nm	7nm+ EUV	5nm 6nm	5nm+	4nm 3nm
GlobalFoundries			22nm FDSOI 12nm finFET		12nm FDSOI	22nm+ FDSOI 12nm+ finFET	
SMIC				14nm finFET	12nm finFET		8-10nm finFET
UMC		14nm finFET			22nm planar		

Note: What defines a process "generation" and the start of "volume" production varies from company to company, and may be influenced by marketing embelishments, so these points of transition should only be seen as very general guidelines.

Sources: Companies, conference reports, IC Insights

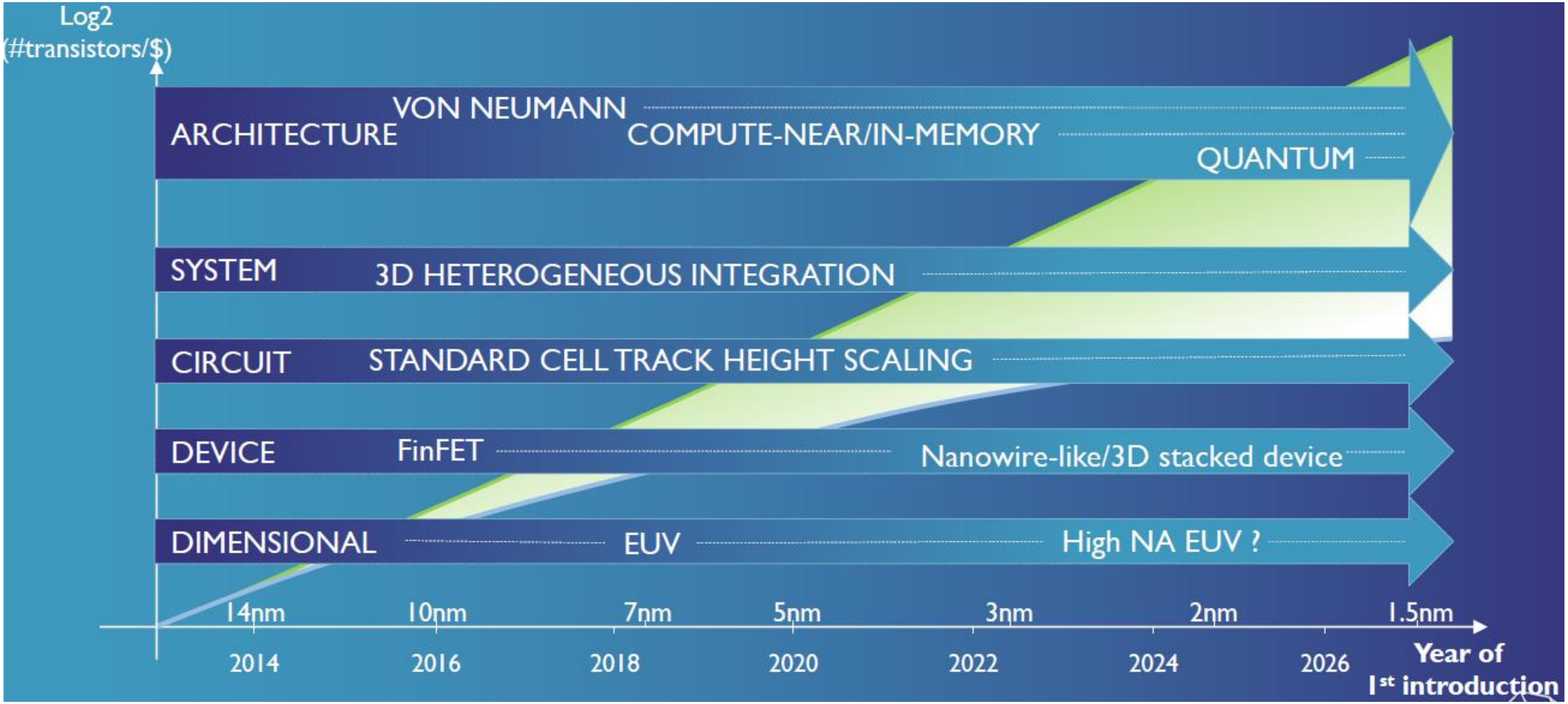
Technology Big Picture & Trends



DTCO: Scaling Beyond the Circuit/Device Level

- **Progress will not only depend on device & circuit technology alone**

DTCO: Design Technology Co-Optimization

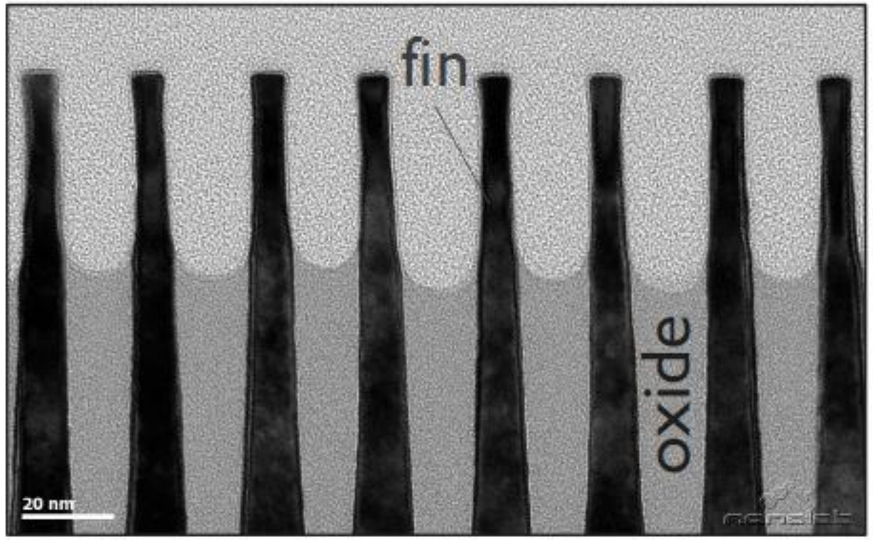


IMEC, ISSCC 2020

IMEC: Changes on Device Level Necessary

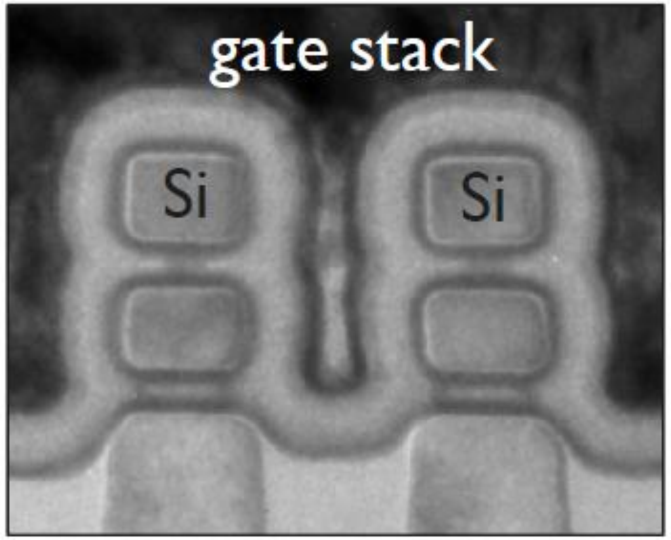
- **Improvements to device structures based on CMOS, aiming for higher density and better gate control**

FINFET



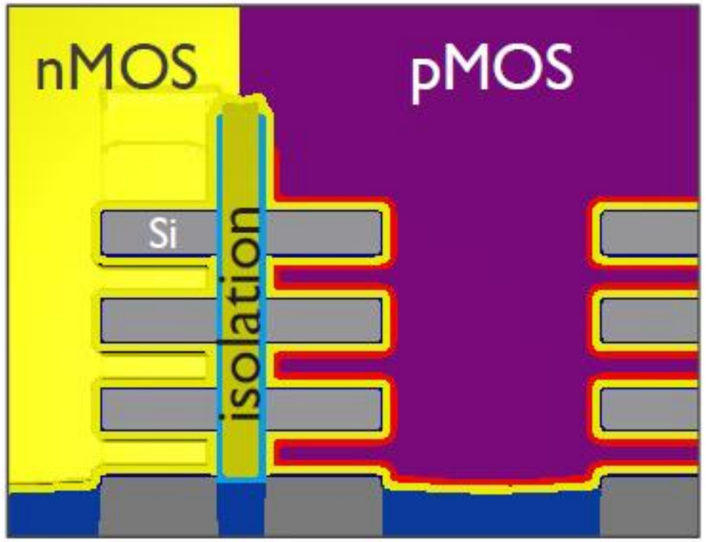
Fin aspect ratio increase
Fin depopulation: #fins/device ↓

NANOSHEETS



Improve electrostatics
Increase drivability

FORKSHEET

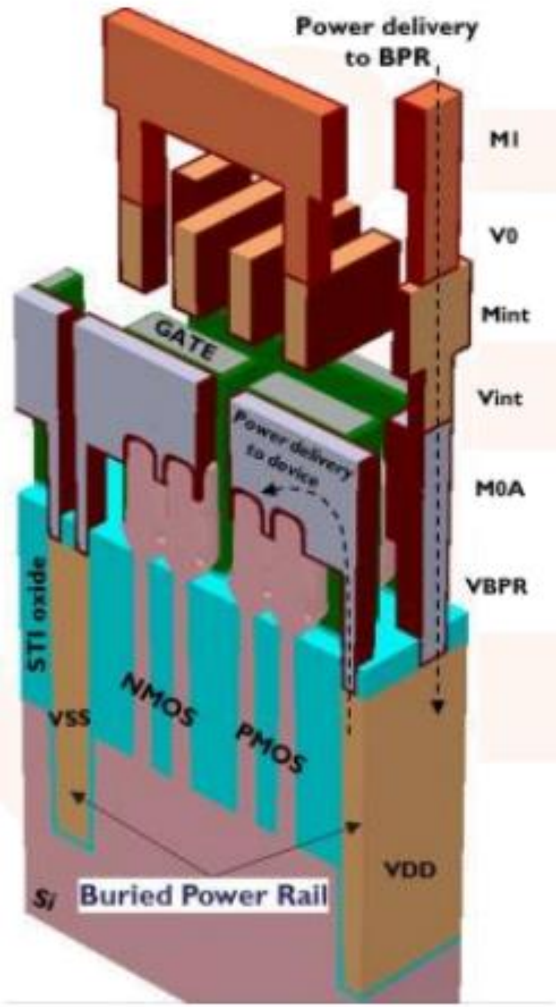


Reduce the N-P separation

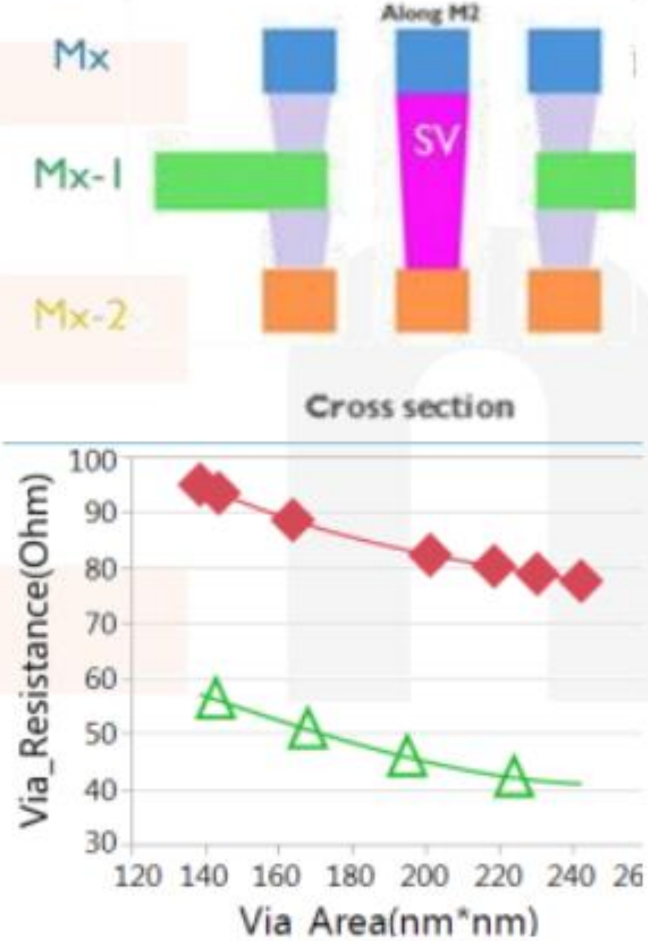
Source: IMEC

IMEC: Backend Innovations

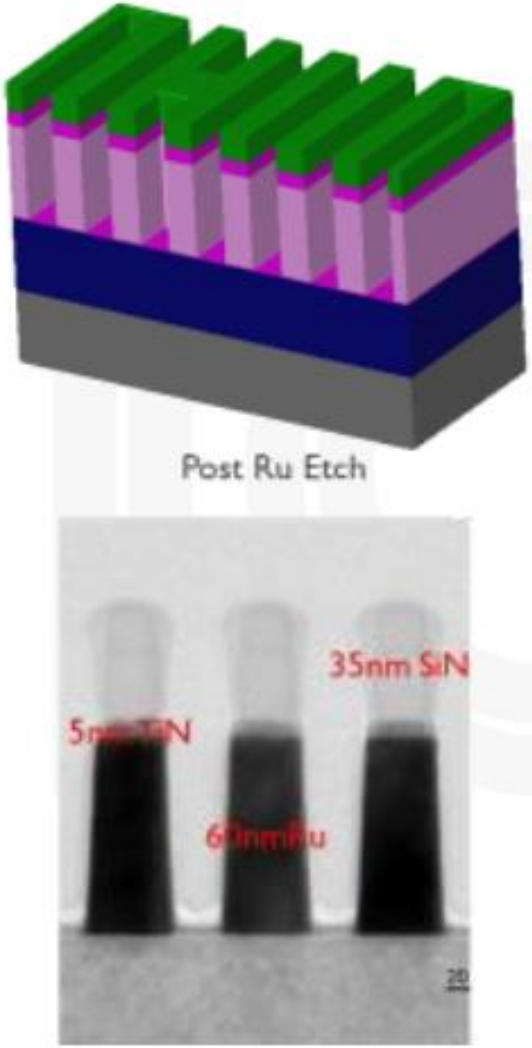
Source: IMEC



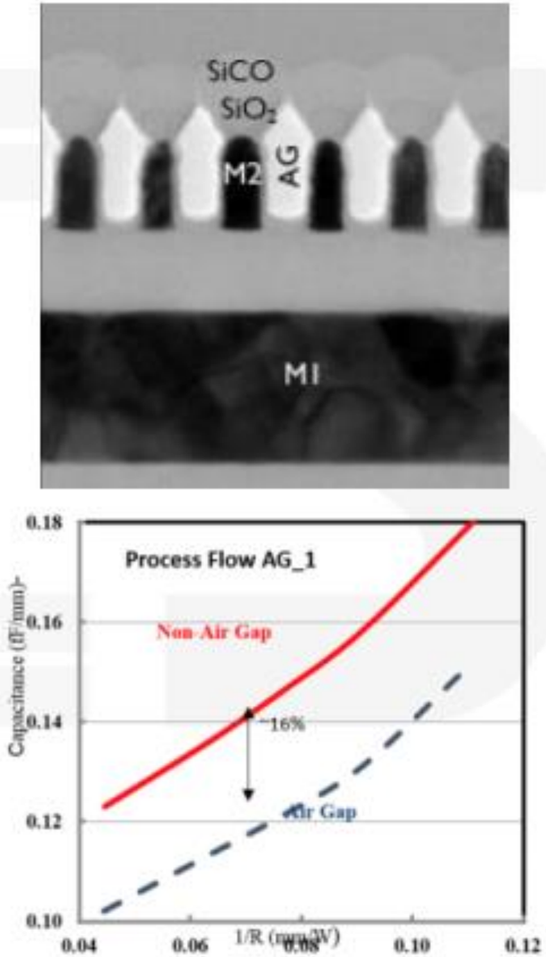
Buried Power Rails (BPR)



Super Vias



Barrier-less Ru



Air Gap ILD

Web Sites and Links

- **Lots of history and fun-facts about the transistor: https://www.pbs.org/transistor/**

- **Youtube shorts**
  - Silicon Doodles: https://youtube.com/playlist?list=PL9EoZAzU-bTAFJ1G_PnHaL7nAvmYSQPy1&si=UmtzisjljjGYdupp
  - Chip closeups: https://www.youtube.com/@EvilmonkeyzDesignz/featured

- **Youtube channels on VLSI Technology and Design**
  - News and latest VLSI technologies: https://www.youtube.com/@AnastasiInTech

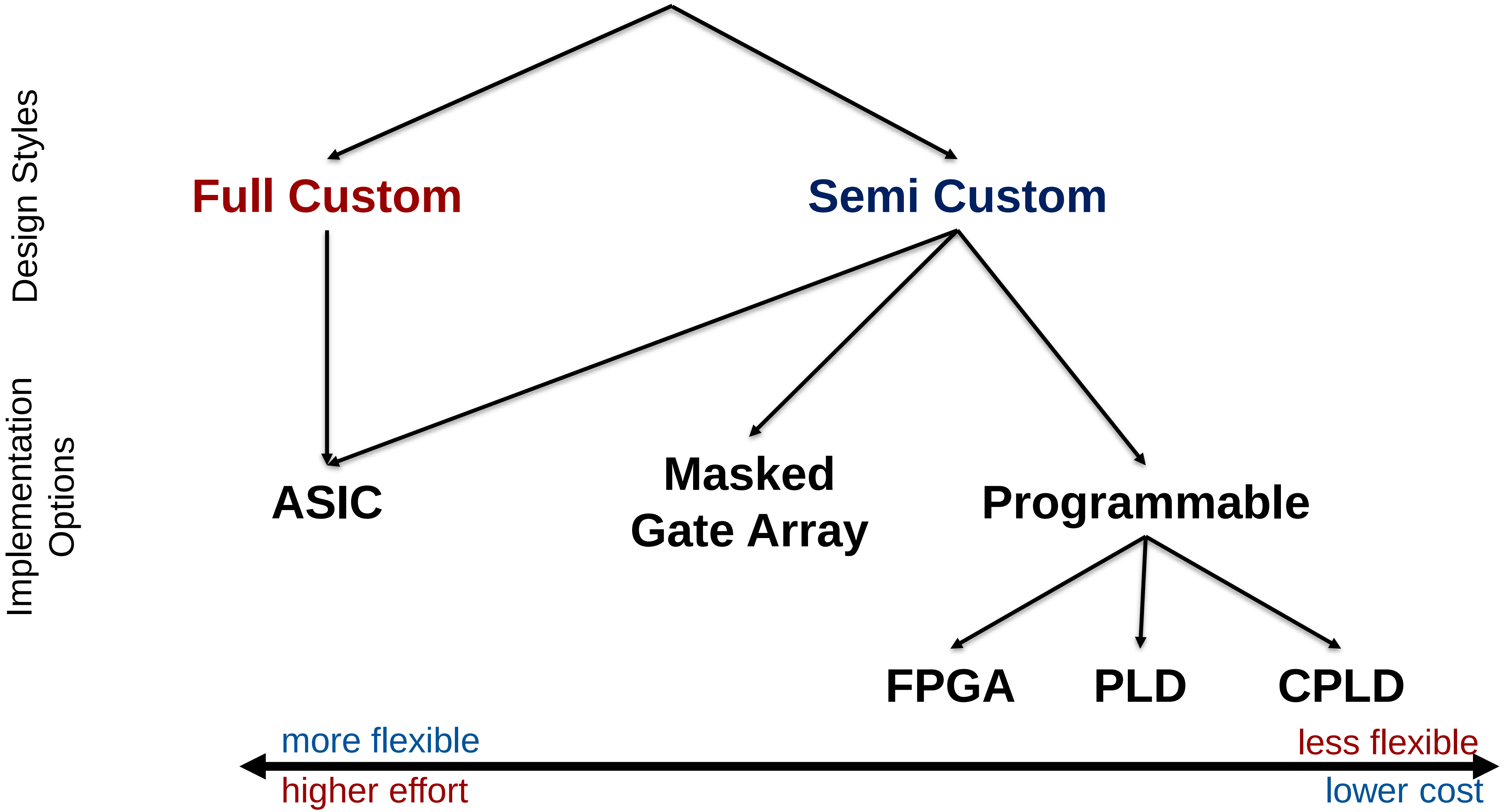
EE-429

Fundamentals of VLSI Design

Digital IC Design Flows
How we design Chips

Andreas Burg

VLSI Design Styles vs Implementation Options



Design Styles

Implementation Options

Full Custom

Semi Custom

ASIC

Masked Gate Array

Programmable

FPGA

PLD

CPLD

more flexible

higher effort

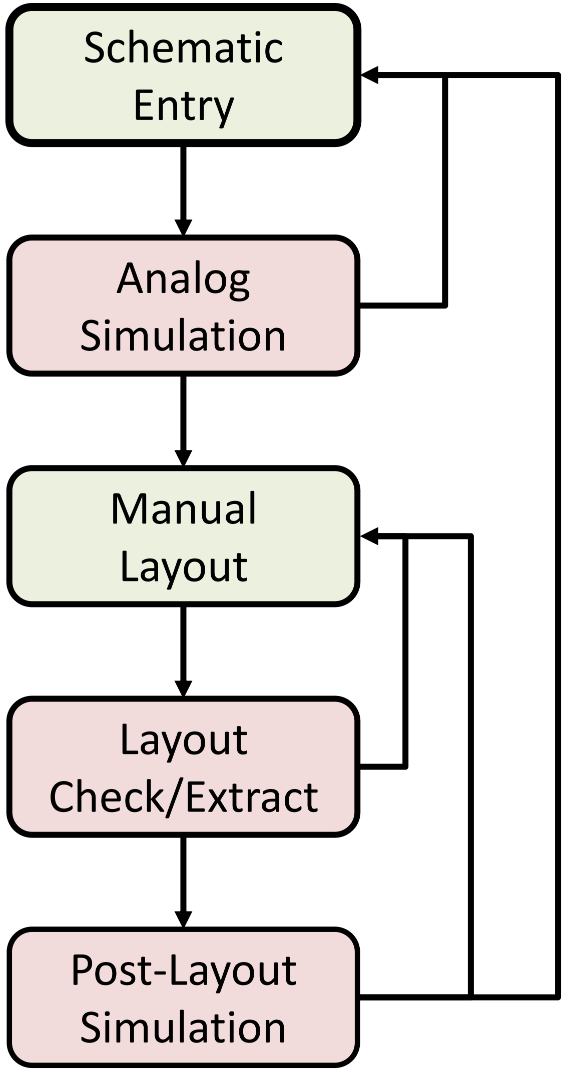
less flexible

lower cost

VLSI Design Styles – Full Custom

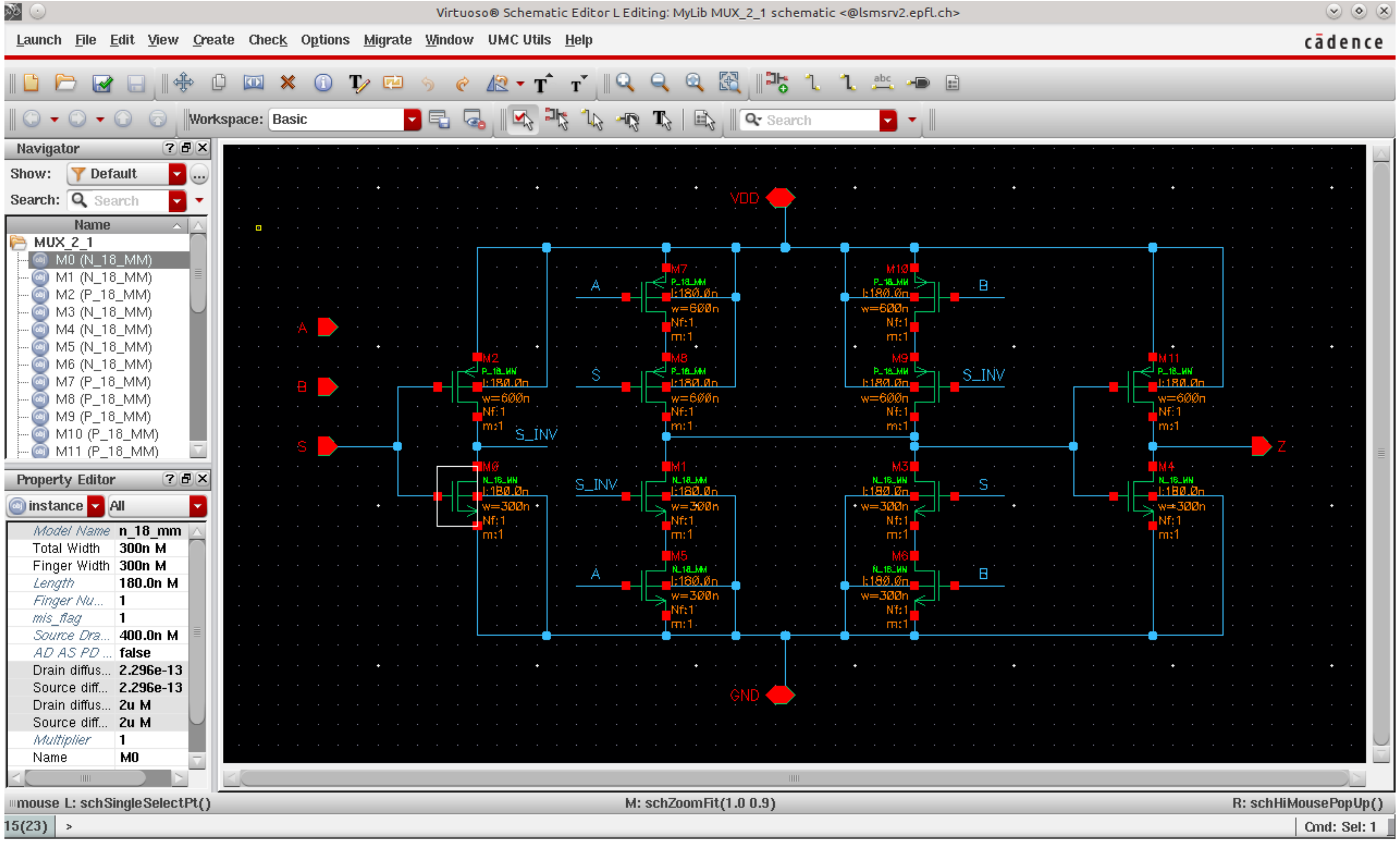
- **Advantages**
  - Everything is done on device/transistor level
    - Full control over all device parameters
    - Full flexibility w.r.t. circuit topology
  - Excellent performance
  - Very high accuracy in simulations
  - No strict separation between analog and digital parts
  - Still the only option for analog design

- **Disadvantages**
  - Everything is done manually
    - Limited design capacity (size) to few hundred devices
    - Long design time and limited re-use
  - No abstraction/approximation
    - Slow



VLSI Design Styles – Full Custom

- **Schematic Entry**



Property	Value
Model Name	n_18_mm
Total Width	300n M
Finger Width	300n M
Length	180.0n M
Finger Nu...	1
mis_flag	1
Source Dra...	400.0n M
AD AS PD ...	false
Drain diffus...	2.296e-13
Source diff...	2.296e-13
Drain diffus...	2u M
Source diff...	2u M
Multiplier	1
Name	M0

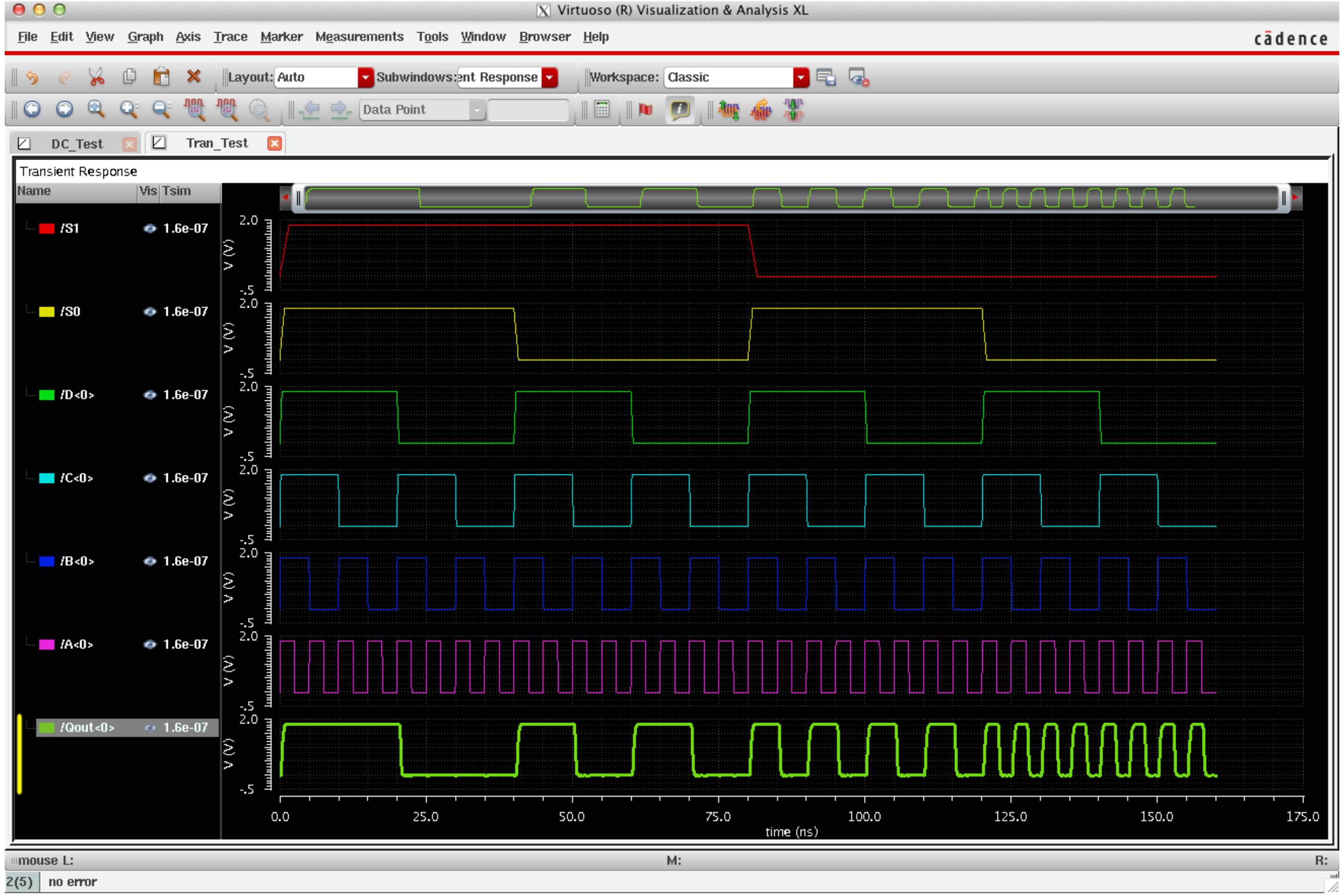
mouse L: schSingleSelectPt()
M: schZoomFit(1.0 0.9)
R: schHiMousePopUp()

15(23) >

Cmd: Sel: 1

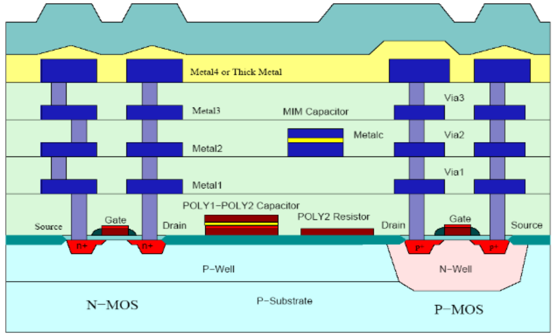
VLSI Design Styles – Full Custom

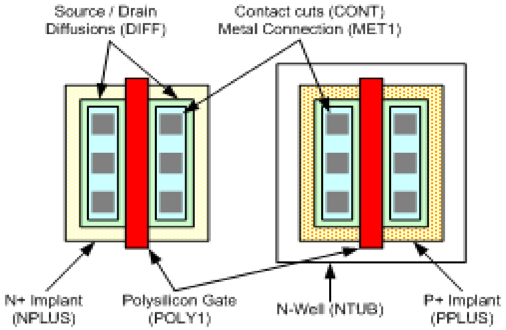
- **Simulation using Spice or Spectre**

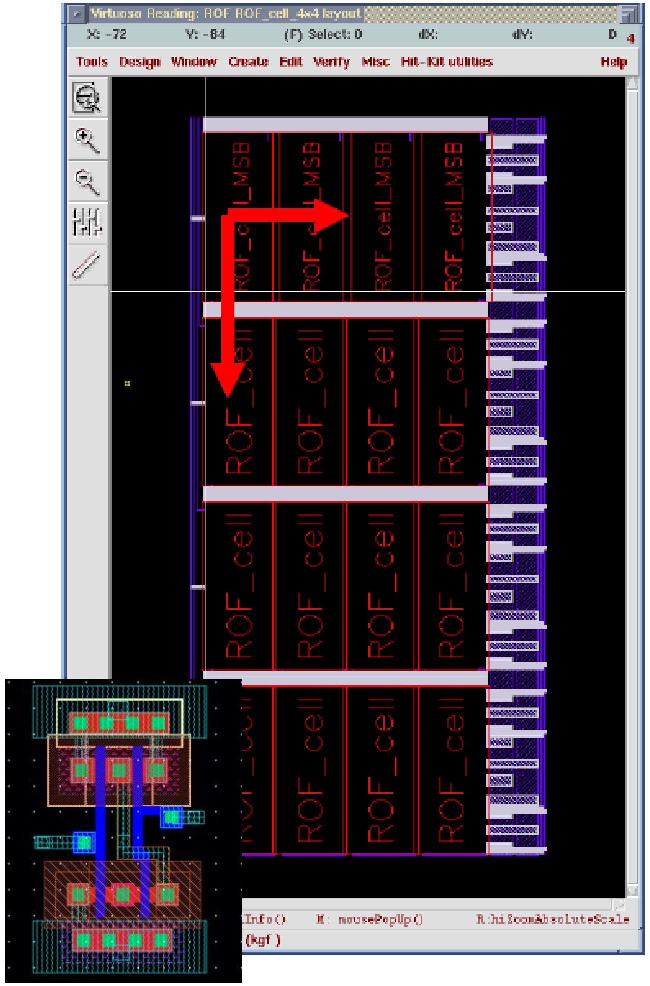


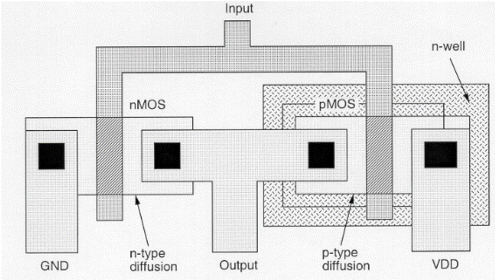
VLSI Design Styles – Full Custom

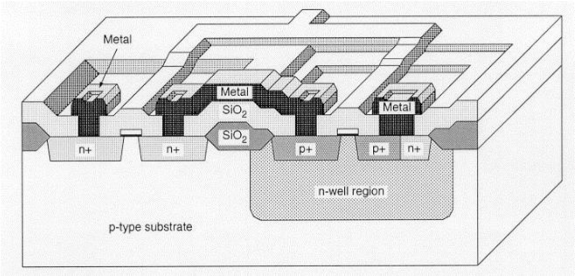
- **Layout: layers represent the masks for production**









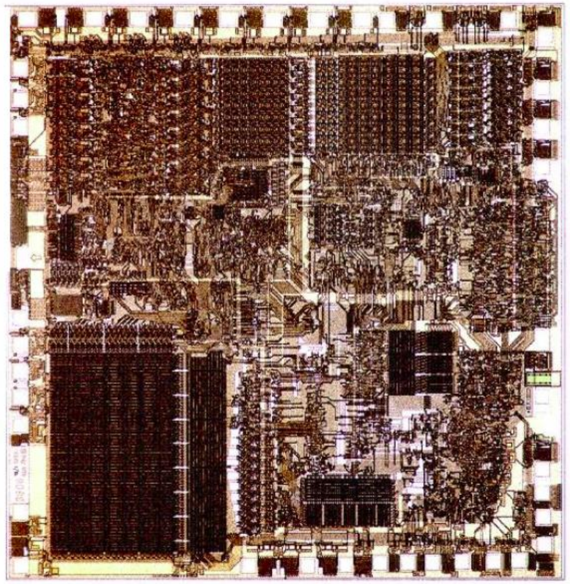


VLSI Design Styles – Full Custom

- In **full-custom** style, the designer has **many degrees of freedom** to optimize:
  - Adjust individual transistor dimensions (width, length, aspect ratio, etc.) to satisfy:
    - DC specifications (voltage levels, switching thresholds)
    - Transient specifications (delay times, rise- and fall-times)
  - Freely choose the most appropriate topology (placement and routing) for each circuit block.
  - Decide on interconnection strategy between blocks.
  - Decide for the global distribution of power, ground and clock.

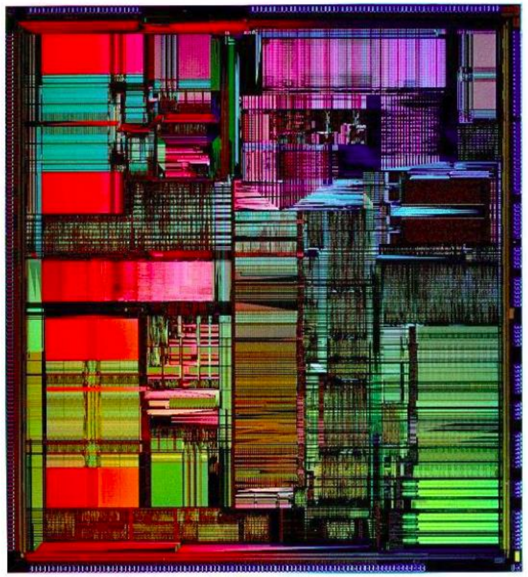
Limits of Full Custom Design

- **Increasing integration density no longer allows for design on transistor level, neither on schematic, nor on layout level**



http://download.intel.com/museum/exhibits/hist_micro/hof/large_jpegs/8088B1.jpg

Intel 8088, 1979
Full-custom design



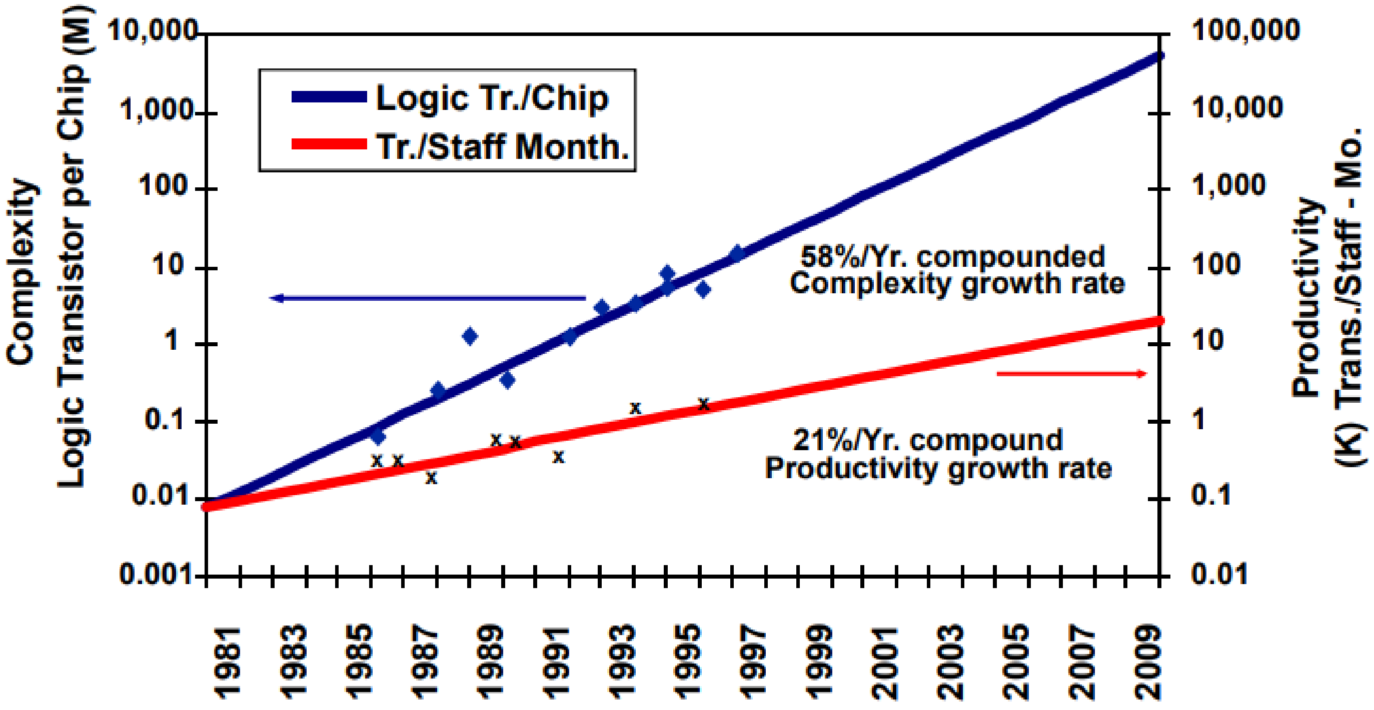
http://download.intel.com/museum/exhibits/hist_micro/hof/large_jpegs/pent8.jpg

Intel Pentium, 1993
Few macros, but mostly built using automatic tools

- **Need for a more automated that leaves the details to EDA tools**

Productivity Gap

- **Moore's law enables rapid increase in number of transistors, i.e., more complex chips every year, BUT productivity needs to keep up to use them**
  - Number of engineers does not increase significantly each year
  - More engineers does not easily result in higher productivity



Source: Sematech

Very Large Scale Integration:

How to design a complex system?

⇩

Need for a well-defined DESIGN METHODOLOGY and
for automatic TOOLS
to handle the complexity.

Architecture Design: Structured Design Principle

- **Hierarchy:** “Divide and conquer” technique involves dividing a module into submodules and then repeating this operation on the sub-modules until the complexity of the smaller parts becomes manageable.

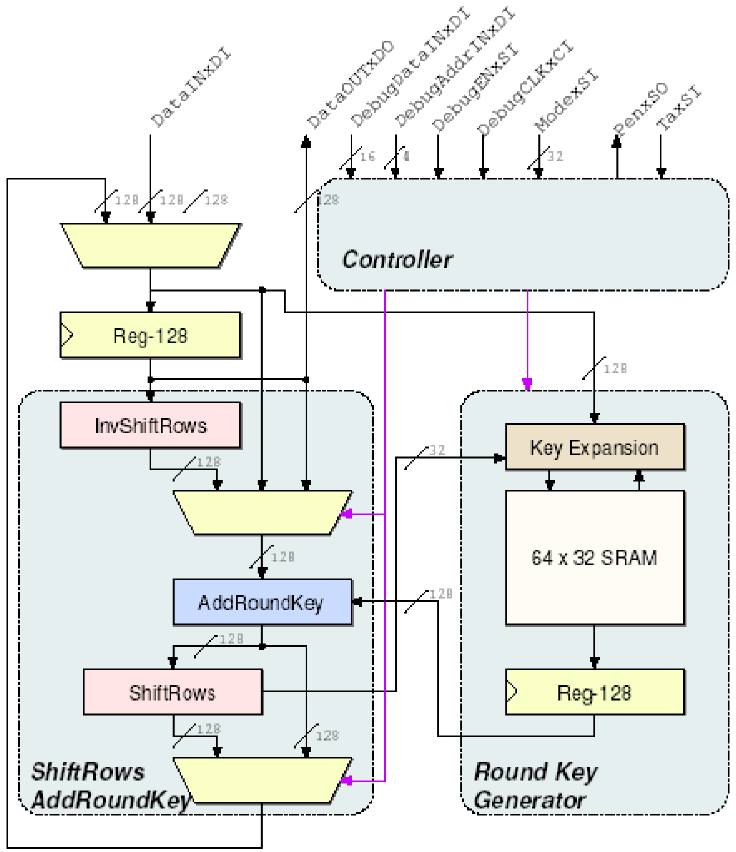
- **Regularity:** The hierarchical decomposition of a large system should result in not only simple, but also similar blocks, as much as possible. Regularity usually reduces the number of different modules that need to be designed and verified, at all levels of abstraction.

- **Modularity:** The various functional blocks which make up the larger system must have well-defined functions and interfaces.

- **Locality:** Internal details remain at the local level. The concept of locality also ensures that connections are mostly between neighboring modules, avoiding long-distance connections as much as possible.

Architecture Design

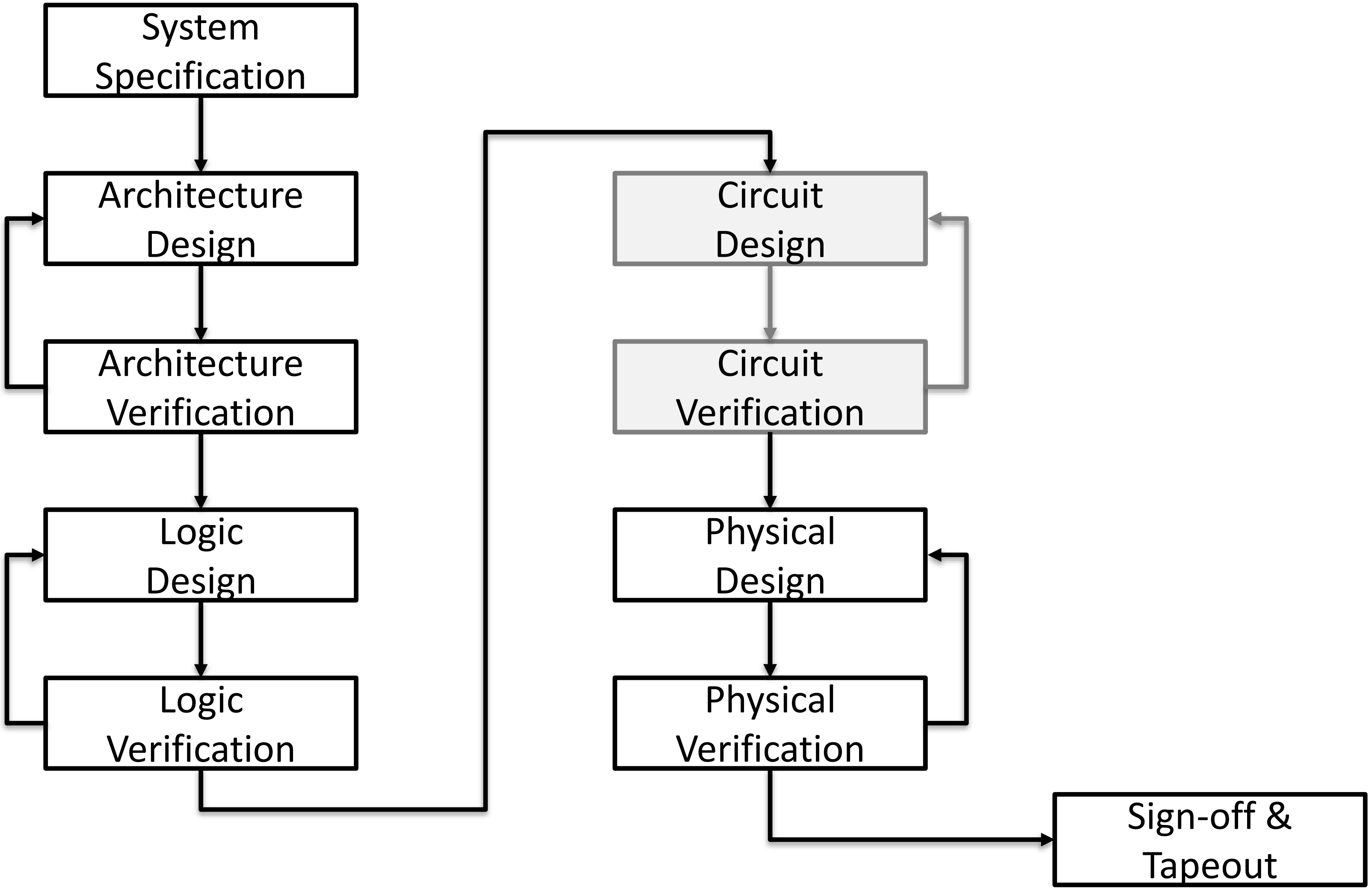
- **The golden rule: ALWAYS START WITH A BLOCK DIAGRAM!**



Use hierarchy also in your block diagram

- **Identify blocks**
  What do we need to perform the functionality
- **Visualize structure**
  How are blocks connected
- **Find critical paths**
  Which block is the most critical (speed, area, power)?
- **Divide and Conquer**
  Use hierarchy, i.e., draw sub-block diagrams

High-Level Design Flow



Principle Idea of Semicustom Design Flow

- **Render the design process more efficiently by using**
  - **Hierarchy**: build complex designs from a collection of smaller and much simpler components which by themselves are again hierarchical
  - **Abstraction**: simplified *description*/characterization *of components as a model (black box)* to better use them on the next level of hierarchy
  - **Design automation**: algorithms and tools to realize an abstract design description from components

Abstraction Levels: Technology

- **Layers of doped semiconductor material or metal interconnect**
  - Possible arrangemet of these layers is determined by manufacturing process
  - Details are one of the best guarded secrets of a foundry (years to develop)

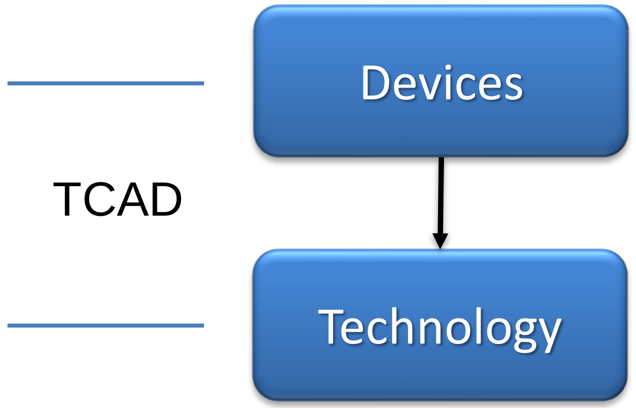
- **Composition of layers determines electrical and other characteristics**
  - Characterized by the complex laws of physics
  - Technology Computer Aided Design (TCAD) tools used for analysis and simulation

Abstraction Levels: Devices

- **Basic building blocks for circuit design that abstract the physics of the technology layer to electrical characteristics**

- ***Foundries* supply a *Process Design Kit* (PDK) which provides**
  - Devices (Transistors, Resistors, Capacitors, Diodes)
  - Layers (e.g., for interconnect)

- **Various “Flavors” of PDKs are available, e.g.:**
  - General Purpose/High Speed/Low Power
  - RF/Image Sensor
  - Flash/DRAM
- **Devices are abstracted to *compact electrical models* for circuit simulations and *design rules***
  - Generated through TCAD and/or measurements

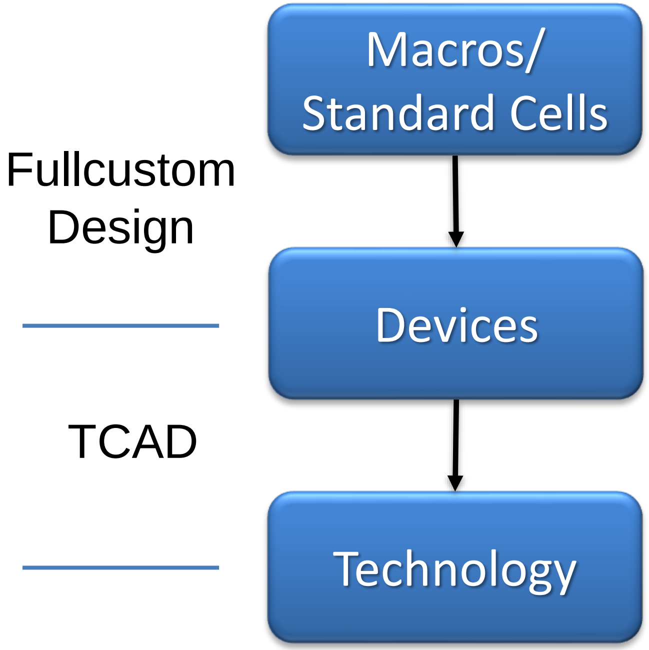


Abstraction Levels: Macros and Standard Cells

- **Basic analog or digital circuits as hierarchical building blocks for more complex circuits**
  - Built from devices of the PDK with full control over all device parameters
  - Characterized and optimized through circuit simulation, e.g., using SPICE
  - Manual, carefully optimized compact layout

- **Library and IP providers offer for example:**
  - Standard cells: libraries of basic digital gates
  - Digital macros such as RAMs/ROMs
  - Analog macros such as ADCs/DACs/…

- **Standard cells/macros are described by abstract models that capture**
  - Functionality.
  - Interface (logical and electrical).
  - Performance (e.g., delay and power)
  - Physical appearance (e.g., size and shape)



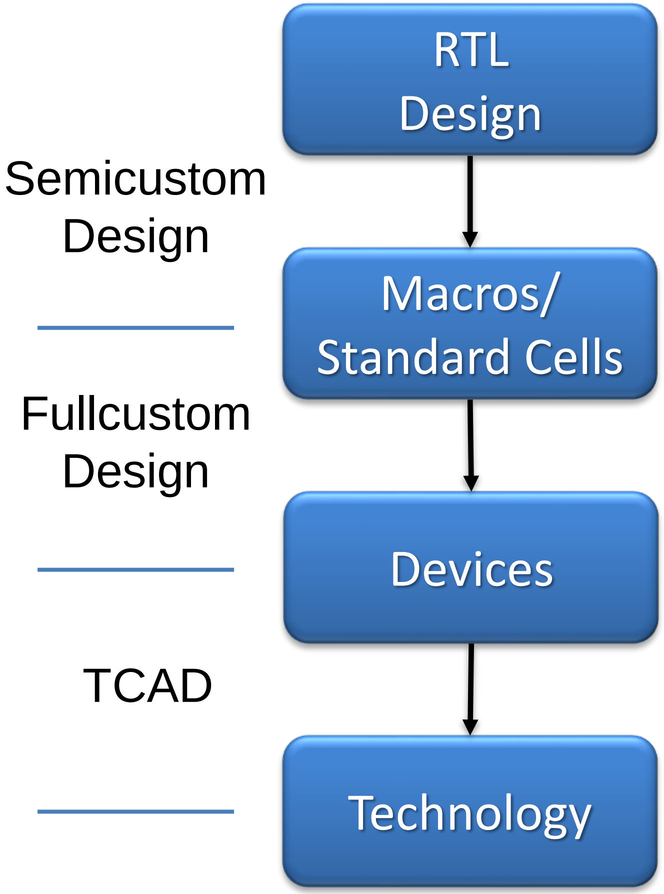
Abstraction Levels: Register Transfer Level (RTL)

- **Complex digital blocks (IPs) and complete complex digital ICs**

- **Specification of a digital circuit in a Hardware Description Language (HDL)**
  - Defines computational logic and storage elements in a abstract way

- **Semicustom design:**
  - Frontend: Automatic translation of HDL into a Gate-Level netlist (a circuit built from standard cells and macro blocks)
  - Backend: Physical implementation of the design based on basic building blocks

- **RTL designs can be abstracted through behavioral models**



RTL Design

Semicustom Design

Macros/ Standard Cells

Fullcustom Design

Devices

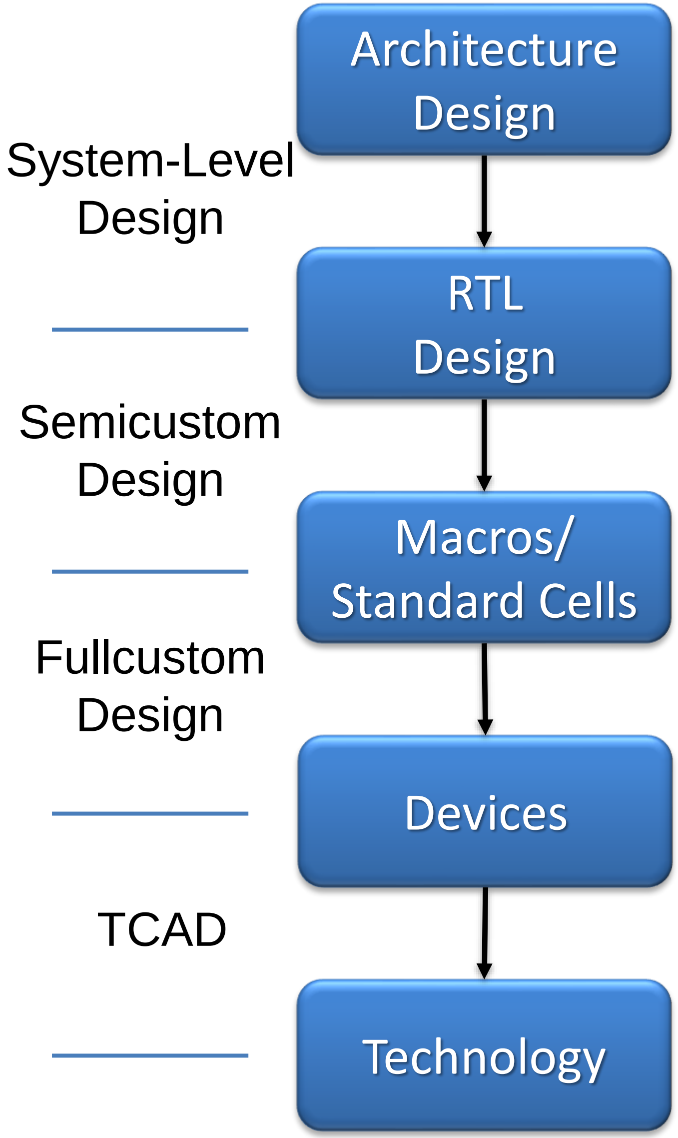
TCAD

Technology

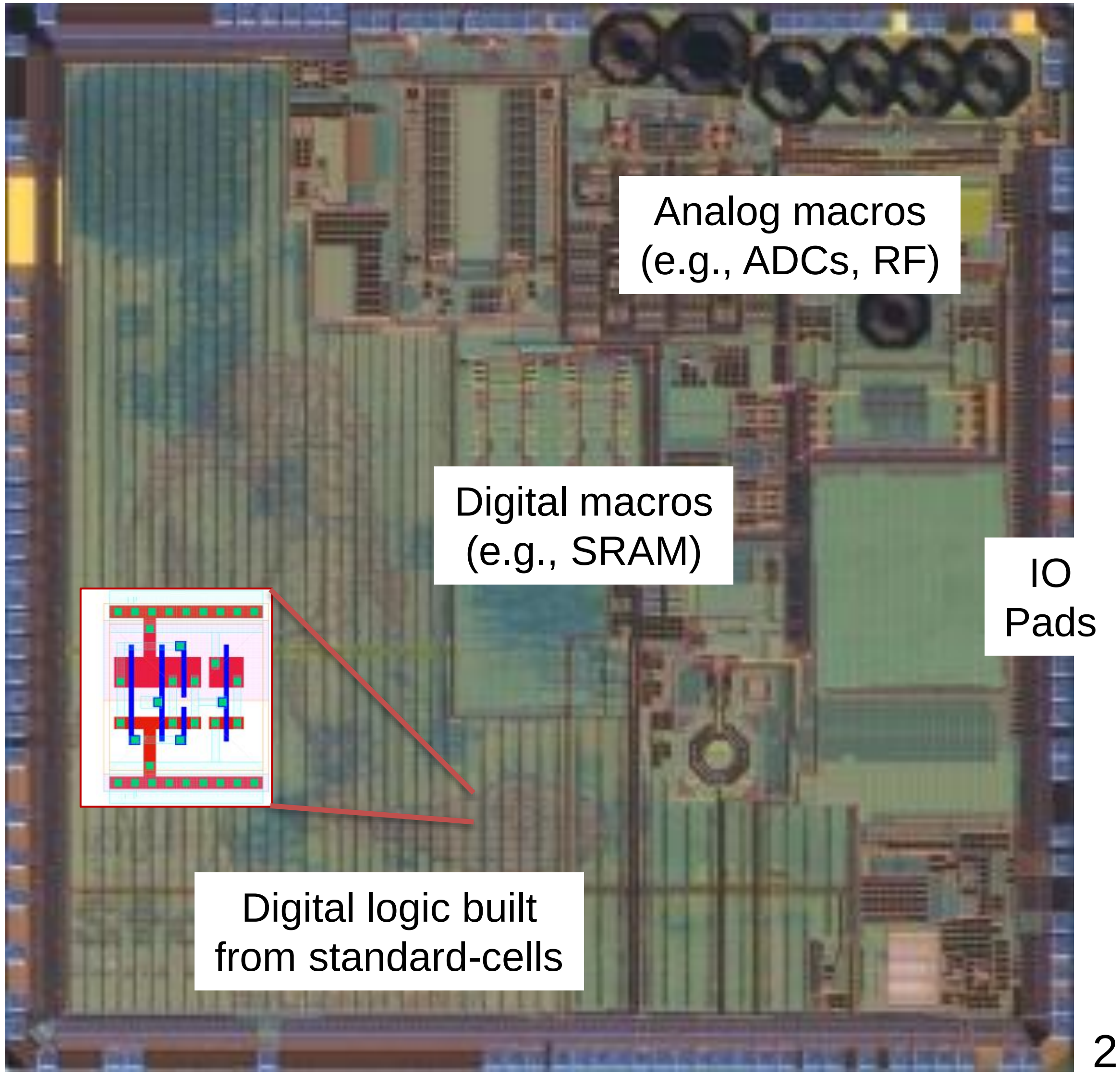
Abstraction Levels: Architecture

- **Entire system (e.g., cell phone)**
  - Complex system comprised of individual (partially highly complex) components
  - Communicating between components often through complex interfaces and protocols

- **Described either**
  - still in RTL by instantiating many other RTL components or
  - in a high-level or HDL language instantiating behavioral models of the RTL components



A Digital IC with a Semicustom Design Flow



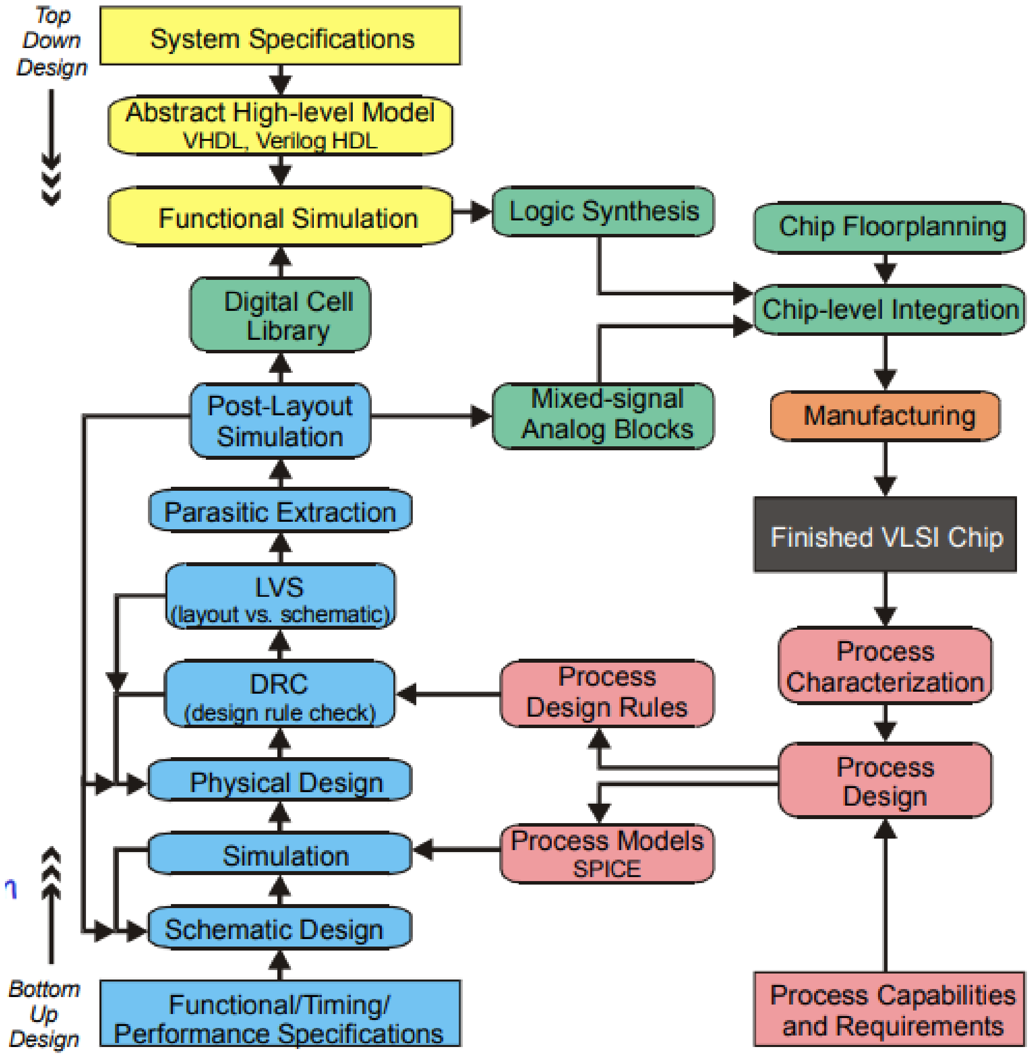
2G Cellular SoC, IIS, ETHZ

Composed from few macros and digital logic realized with standard cells & IO pads

Semi-Custom with Full-Custom

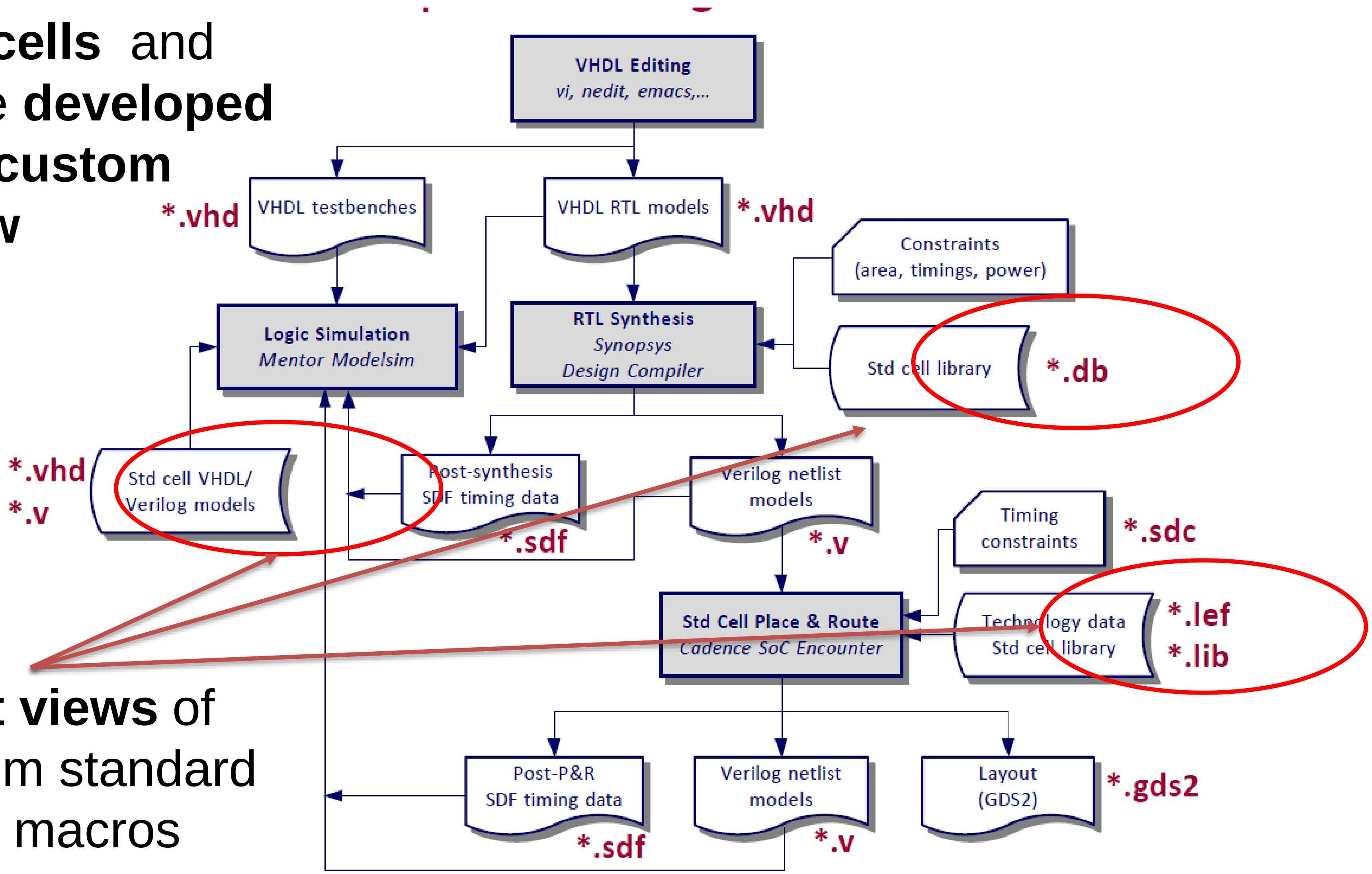
- **Semi-custom design builds on**
  - full-custom components as basic building blocks
  - full-custom principles and insights for optimization

- Examples
  - Standard cells
  - Macro cells (e.g., memories)
  - Methods for sizing
  - Methods for buffering
  - Analysis of timing/power
  - ...



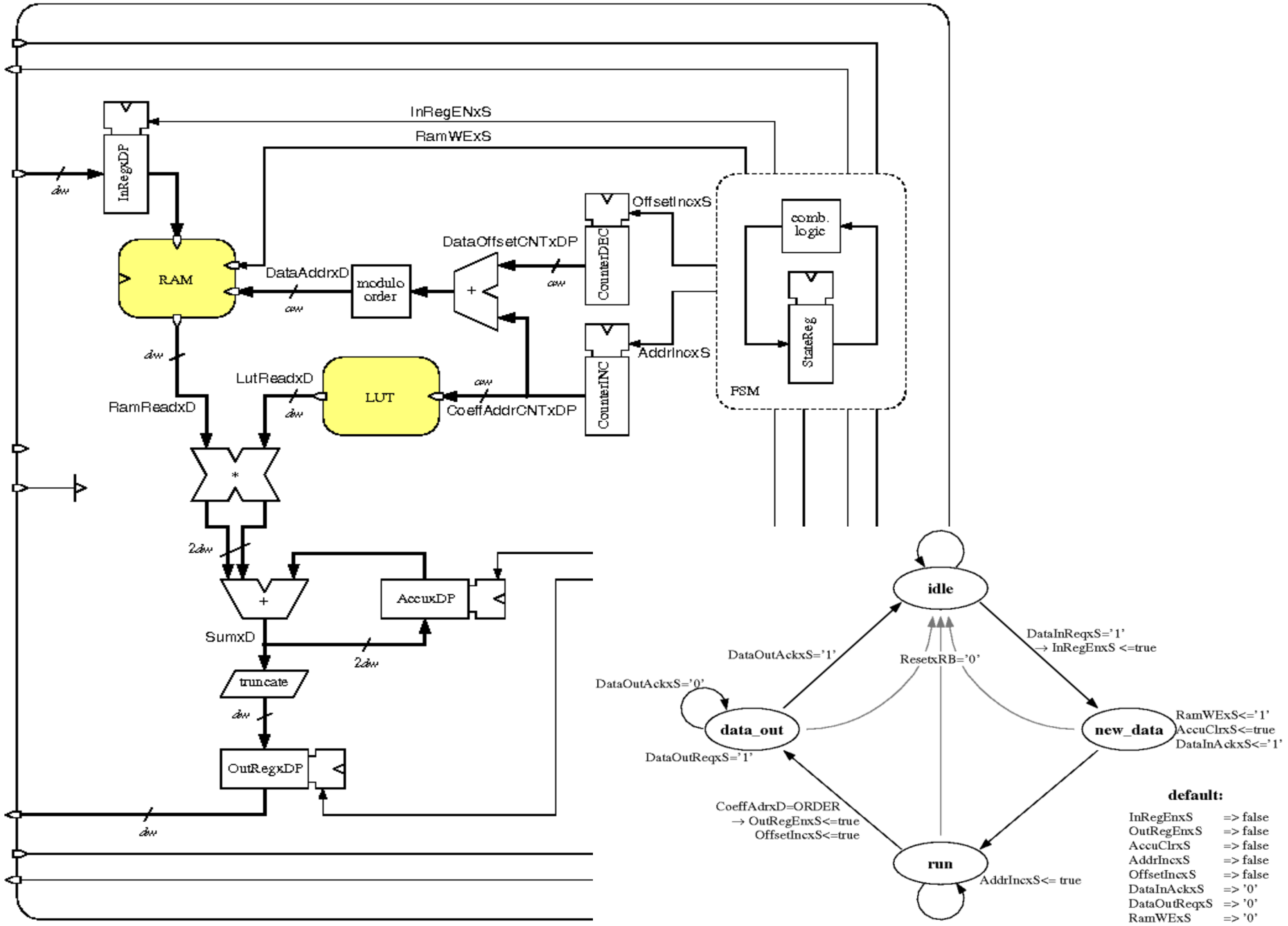
Semicustom Design Flow

Standard cells and macros are **developed in** the **full-custom** design **flow**



Abstract views of full-custom standard cells and macros

Semicustom Design Flow: Architecture Design



Semicustom Design Flow: Design Entry in VHDL

- **Modern hardware description languages allow structural descriptions that basically determine all interconnections between well-defined blocks:**

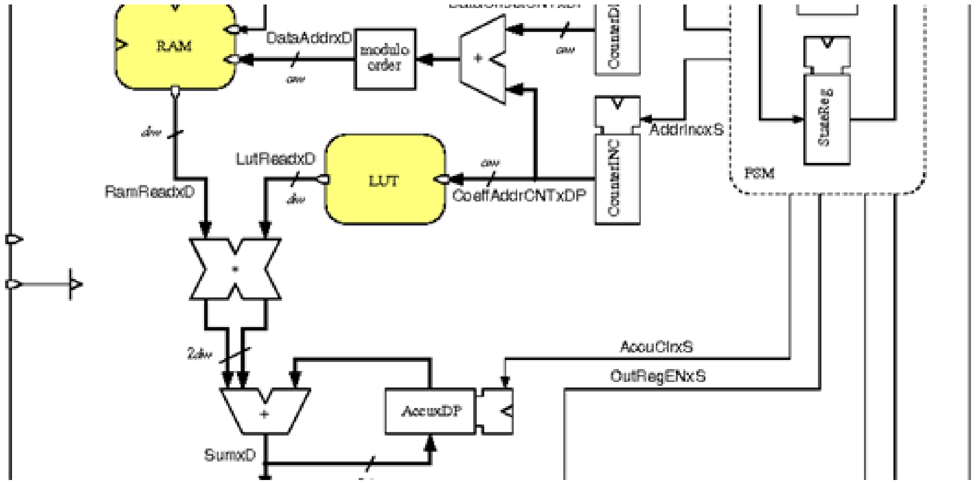
- `adder: component adder port map (a,b,ci,co,s);`

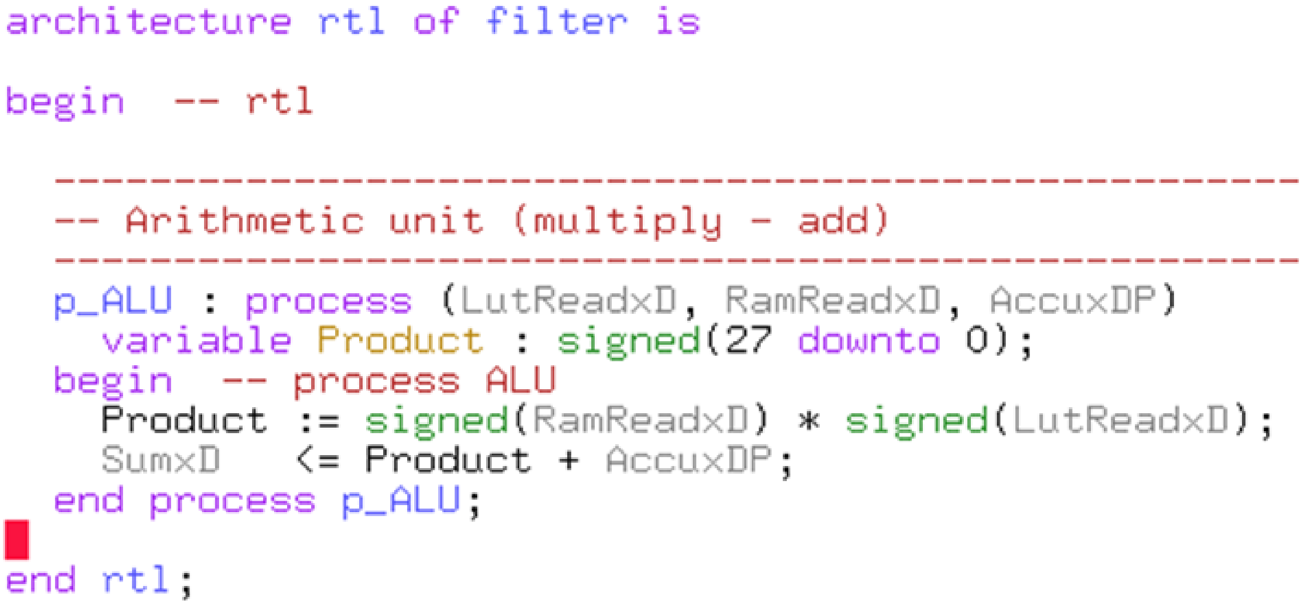
- **as well as purely behavioral descriptions like:**

- `sum <= a + b - c;`

- **The synthesizer is responsible for converting the behavioral description into an optimized design.**

Semicustom Design Flow: Design Entry in VHDL





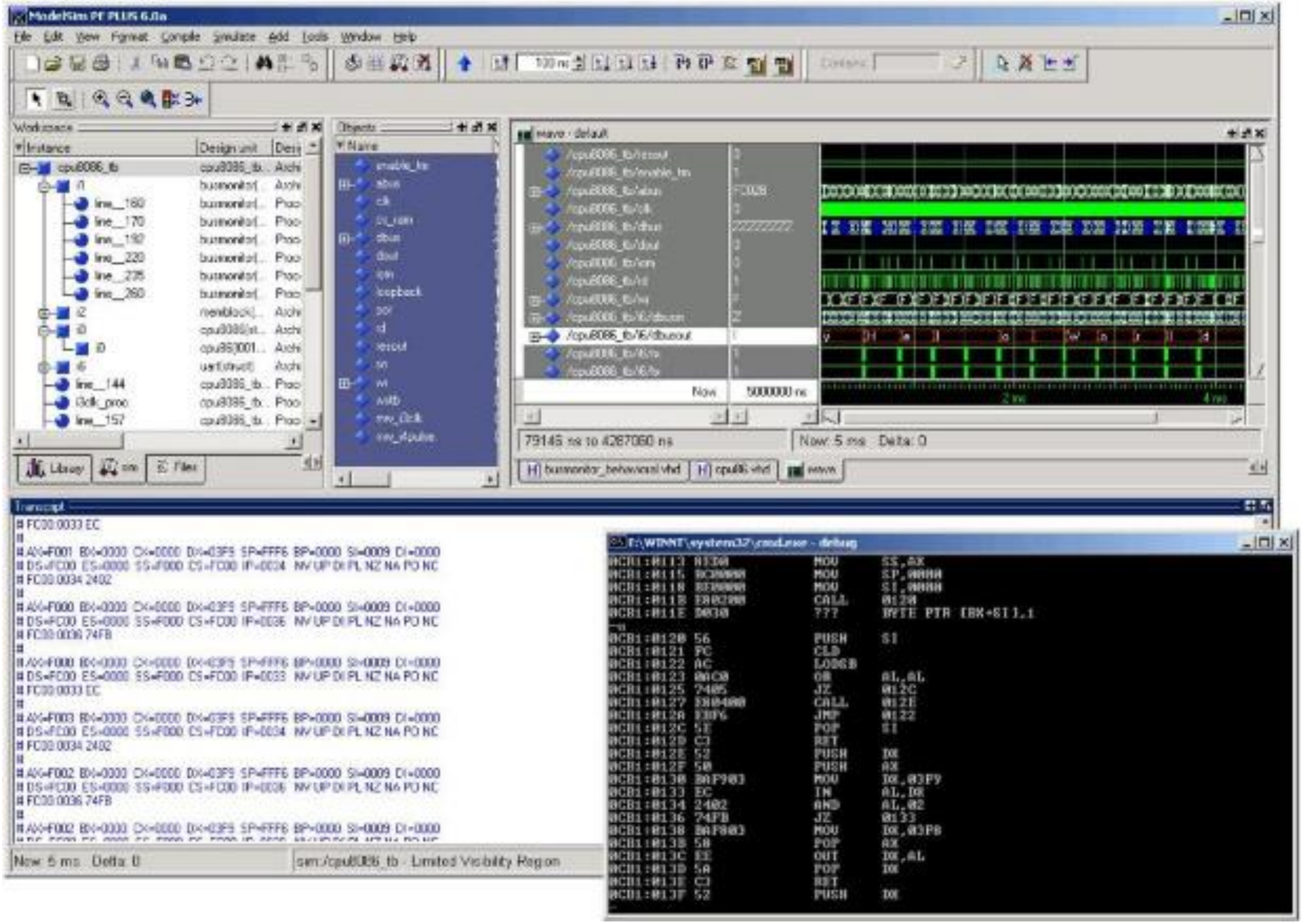
```
architecture rtl of filter is

begin  -- rtl

  ----------------------------------------------------
  -- Arithmetic unit (multiply - add)
  ----------------------------------------------------
  p_ALU : process (LutReadxD, RamReadxD, AccuxDP)
    variable Product : signed(27 downto 0);
  begin  -- process ALU
    Product := signed(RamReadxD) * signed(LutReadxD);
    SumxD    <= Product + AccuxDP;
  end process p_ALU;

end rtl;
```

Semicustom Design Flow: Behavioral Verification

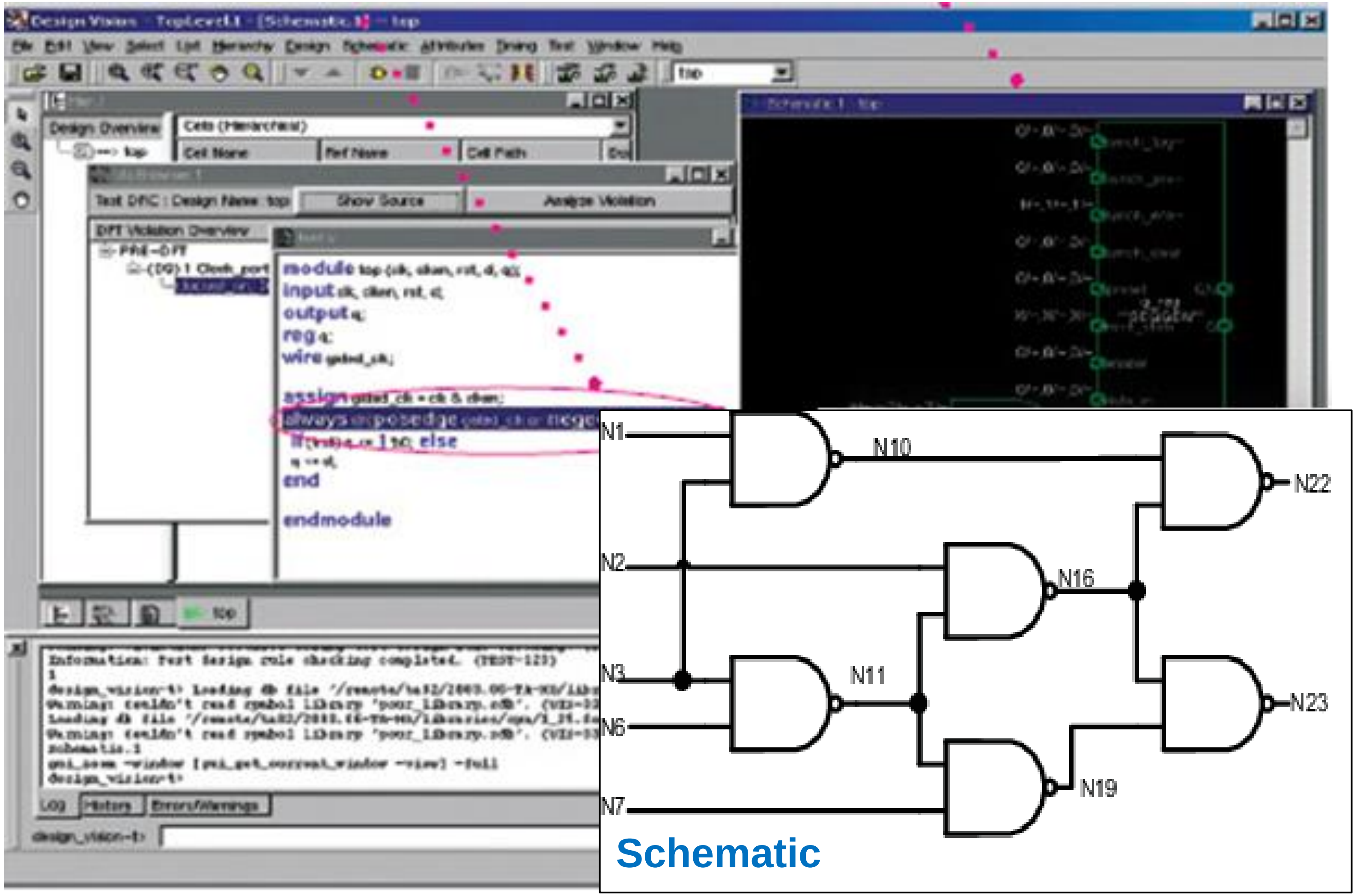


```
# FC00:0033 EC
#
# AX=F001 BX=0000 CX=0000 DX=03F9 SP=FFF6 BP=0000 SI=0009 DI=0000
# DS=FC00 ES=0000 SS=F000 CS=FC00 IP=0034  NV UP DI PL NZ NA PO NC
# FC00:0034 2402
#
# AX=F000 BX=0000 CX=0000 DX=03F9 SP=FFF6 BP=0000 SI=0009 DI=0000
# DS=FC00 ES=0000 SS=F000 CS=FC00 IP=0036  NV UP DI PL NZ NA PO NC
# FC00:0036 74FB
#
# AX=F000 BX=0000 CX=0000 DX=03F9 SP=FFF6 BP=0000 SI=0009 DI=0000
# DS=FC00 ES=0000 SS=F000 CS=FC00 IP=0033  NV UP DI PL NZ NA PO NC
# FC00:0033 EC
#
# AX=F003 BX=0000 CX=0000 DX=03F9 SP=FFF6 BP=0000 SI=0009 DI=0000
# DS=FC00 ES=0000 SS=F000 CS=FC00 IP=0034  NV UP DI PL NZ NA PO NC
# FC00:0034 2402
#
# AX=F002 BX=0000 CX=0000 DX=03F9 SP=FFF6 BP=0000 SI=0009 DI=0000
# DS=FC00 ES=0000 SS=F000 CS=FC00 IP=0036  NV UP DI PL NZ NA PO NC
# FC00:0036 74FB
#
# AX=F002 BX=0000 CX=0000 DX=03F9 SP=FFF6 BP=0000 SI=0009 DI=0000
```

```
0CB1:0113 8ED0          MOV     SS,AX
0CB1:0115 BC0000        MOV     SP,0000
0CB1:0118 BE0000        MOV     SI,0000
0CB1:011B E80200        CALL    0120
0CB1:011E D030          ???     BYTE PTR [BX+SI],1
-u
0CB1:0120 56            PUSH    SI
0CB1:0121 FC            CLD
0CB1:0122 AC            LODSB
0CB1:0123 0AC0          OR      AL,AL
0CB1:0125 7405          JZ      012C
0CB1:0127 E80400        CALL    012E
0CB1:012A EBF6          JMP     0122
0CB1:012C 5E            POP     SI
0CB1:012D C3            RET
0CB1:012E 52            PUSH    DX
0CB1:012F 50            PUSH    AX
0CB1:0130 BAF903        MOV     DX,03F9
0CB1:0133 EC            IN      AL,DX
0CB1:0134 2402          AND     AL,02
0CB1:0136 74FB          JZ      0133
0CB1:0138 BAF803        MOV     DX,03F8
0CB1:013B 58            POP     AX
0CB1:013C EE            OUT     DX,AL
0CB1:013D 5A            POP     DX
0CB1:013E C3            RET
0CB1:013F 52            PUSH    DX
```

Semicustom Design Flow: Logic Synthesis

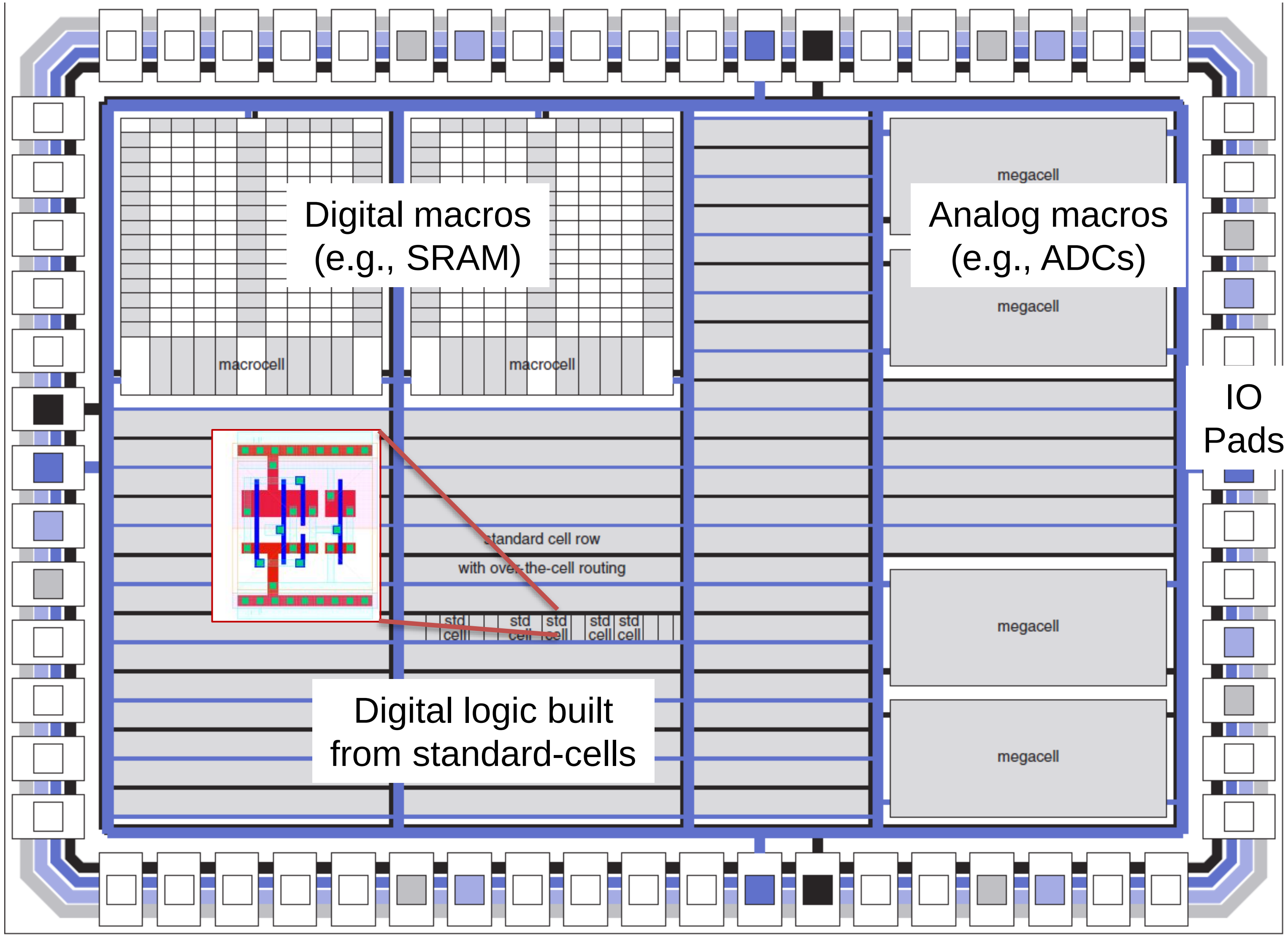
- **Turn HDL description into a schematic/netlist of pre-defined standard-cells**



Netlist: textual representation of a schematic in **VERILOG** (or VHDL)

```
module c17 (N1,N2,N3,N6,N7,N22,N23);
input N1,N2,N3,N6,N7;
output N22,N23;
wire N10,N11,N16,N19;
nand NAND2_1 (N10, N1, N3);
nand NAND2_2 (N11, N3, N6);
nand NAND2_3 (N16, N2, N11);
nand NAND2_4 (N19, N11, N7);
nand NAND2_5 (N22, N10, N16);
nand NAND2_6 (N23, N16, N19);
endmodule
```

Semicustom Design Flow: Floorplan



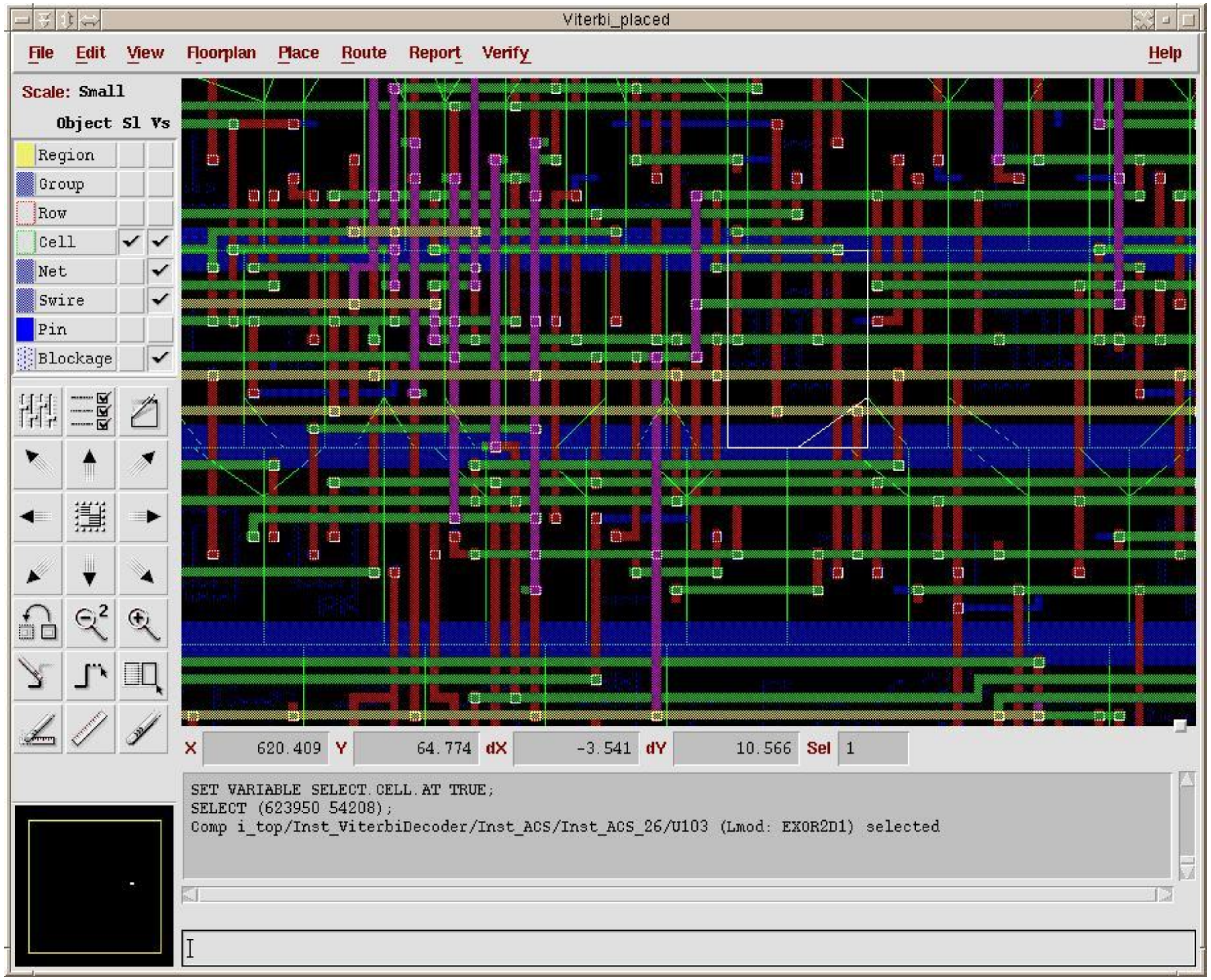
A **Floorplan defines** the **structure of the layout**, but not all its details.

A floorplan contains for ex.:

- Power distribution
- Location of the IO
- Locations of macros
- Areas for standard cells
- Routing channels
- …

Semicustom Design Flow Automatic Place & Route

- **Standard-cells are placed automatically** in the regions defined in the floorplan
- **Automatic routing** with Manhatten style to be able to handle millions of cells

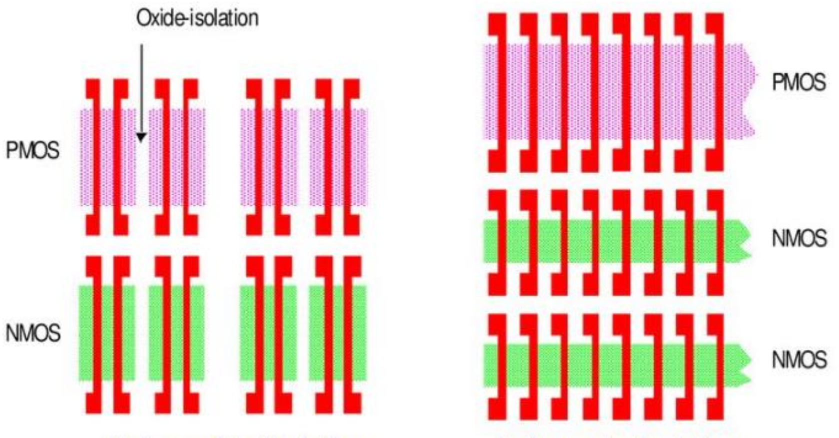


X 620.409 Y 64.774 dX -3.541 dY 10.566 Sel 1

```
SET VARIABLE SELECT.CELL.AT TRUE;
SELECT (623950 54208);
Comp i_top/Inst_ViterbiDecoder/Inst_ACS/Inst_ACS_26/U103 (Lmod: EXOR2D1) selected
```

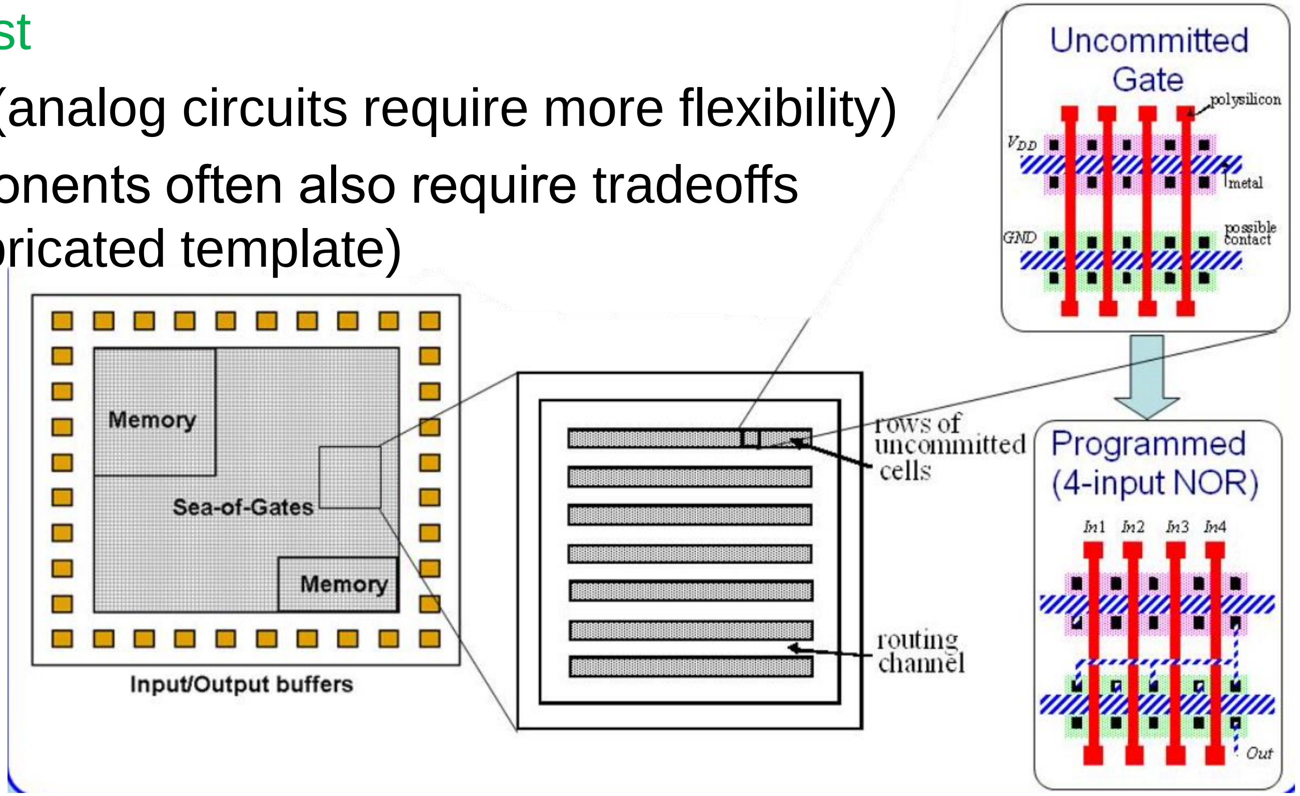
Cell Based Design and Sea-of-Gates

- **Regular custom ASIC for which the costly low-level (FEOL) masks are fixed**
  - Fixed, sometimes pre-fabricated template including especially the FEOL, sometimes also lowest level interconnect (Cell-based design)
- **Design flexibility provided through custom routing and vias (BEOL)**
  - Compromise between full flexibility and cost
  - Almost exclusively used for digital circuits (analog circuits require more flexibility)
  - Memories and other “custom” digital components often also require tradeoffs (sometimes available as part of the pre-fabricated template)



Using oxide-isolation

Using gate-isolation



The Semiconductor Ecosystem

Fabless Companies

QUALCOMM
SanDisk
MARVELL
TEXAS INSTRUMENTS
BROADCOM.
AMD
PMC
PMC-SIERRA
Micron
freescale
semiconductor

CAD/EDA

SYNOPSYS
cādence
SIEMENS
Mentor Graphics

IP Vendors

ARM
FARADAY

Fab/Foundry

TOWERJAZZ
tsmc
UMC
SMIC
GLOBAL FOUNDRIES