Advanced VLSI Design

Instructor

Chintan Patel

(contact using email: cpatel2@cs.umbc.edu)

Text & References

No required text

See syllabus for details on References

Online Cadence Documentation

Prerequisite

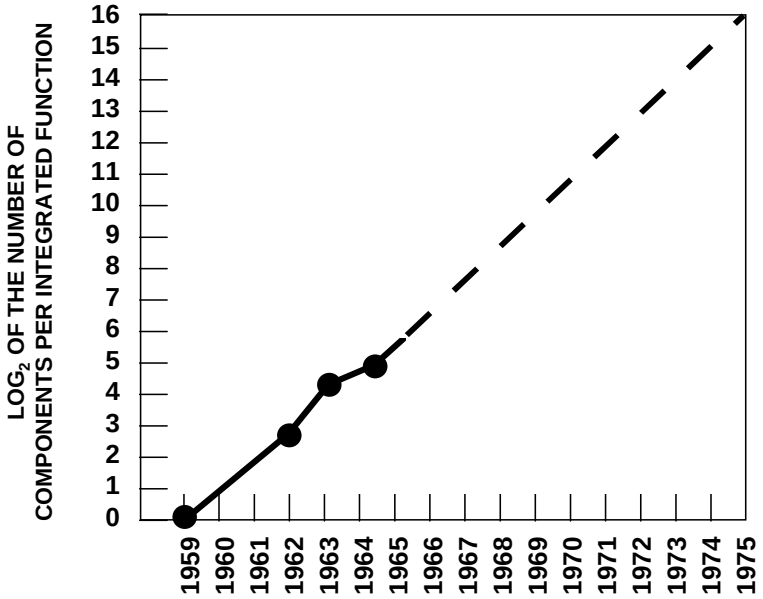
CMPE 640, Digital Design, Good Layout Skills, Computer Architecture

Further Info

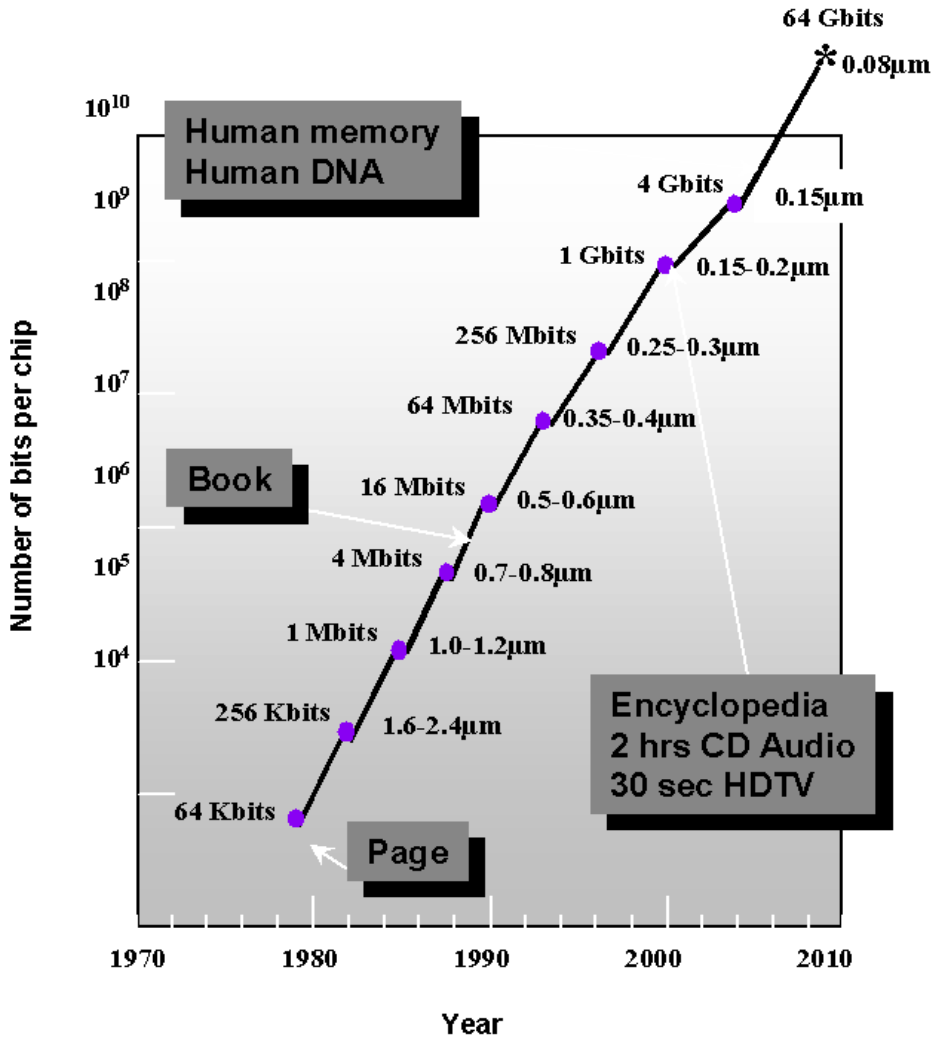
http://www.cs.umbc.edu/~cpatel2

Moore's Law

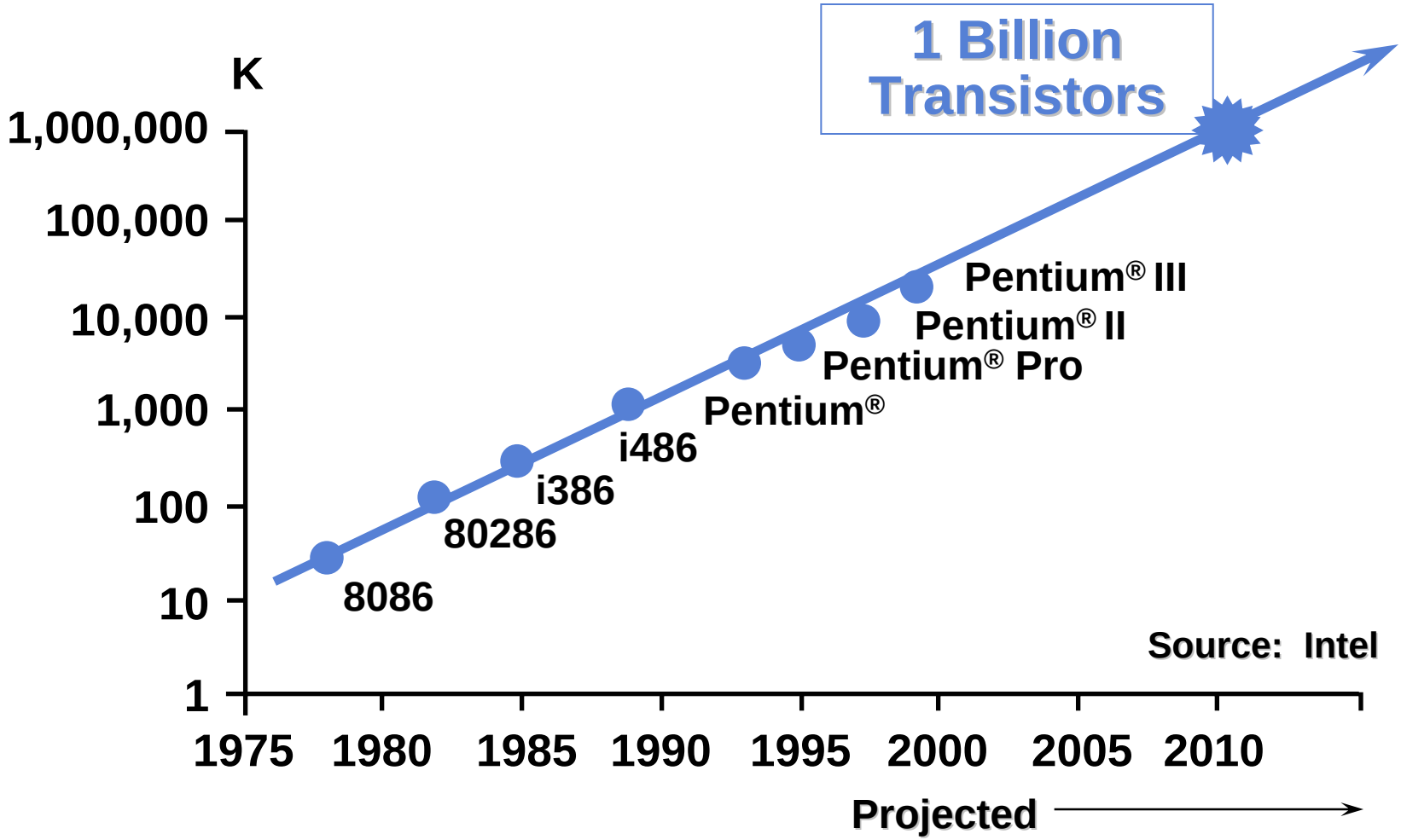
- In 1965, Gordon Moore noted that the number of transistors on a chip doubled every 18 to 24 months
  - Transistors/Chip increasing by 50% per year (by 4× in 3.5 years)
  - Gate Delay decreasing by 13% per year (by ½ in 5 years)
- This rate of improvement will continue until about 2018 at least



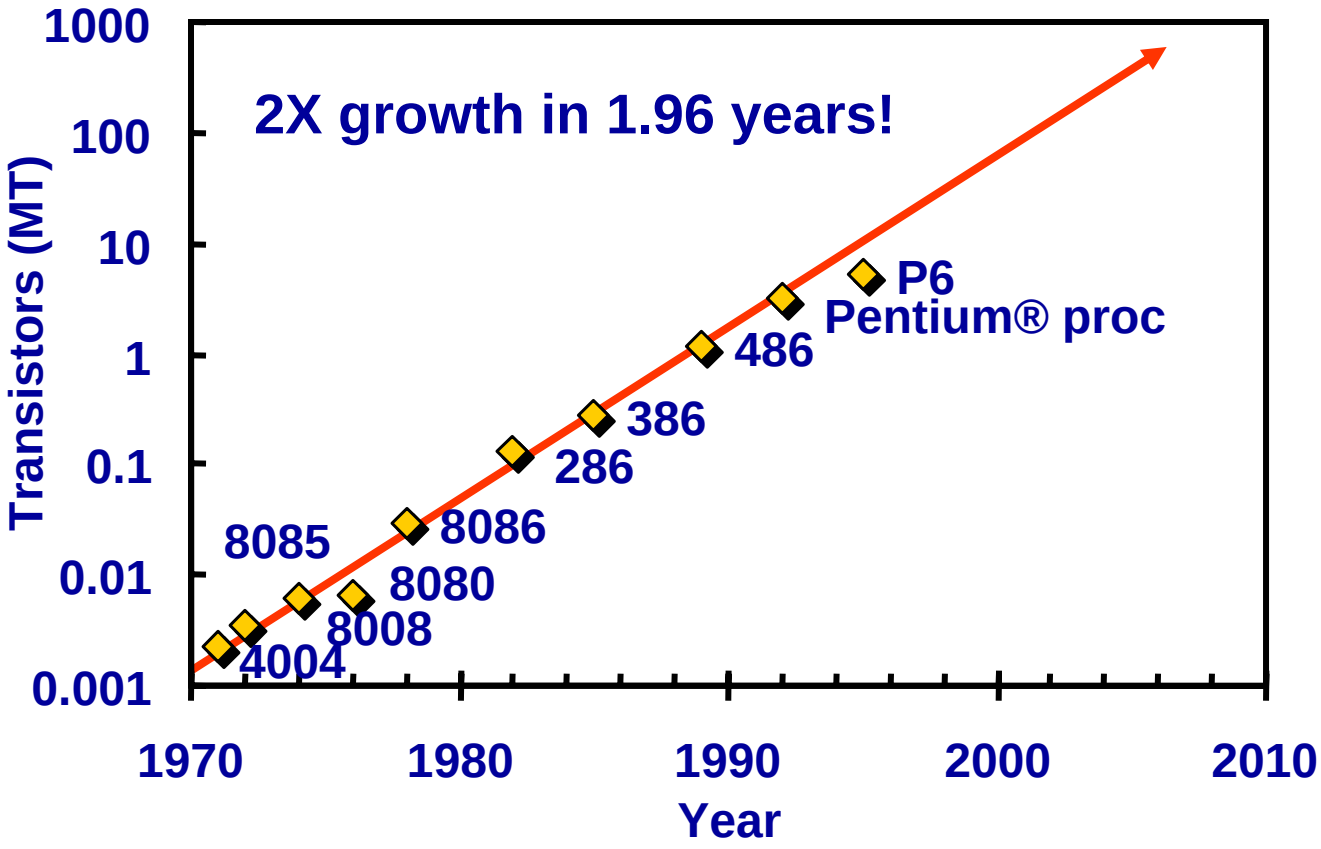
Evolution in Complexity



Transistor Counts



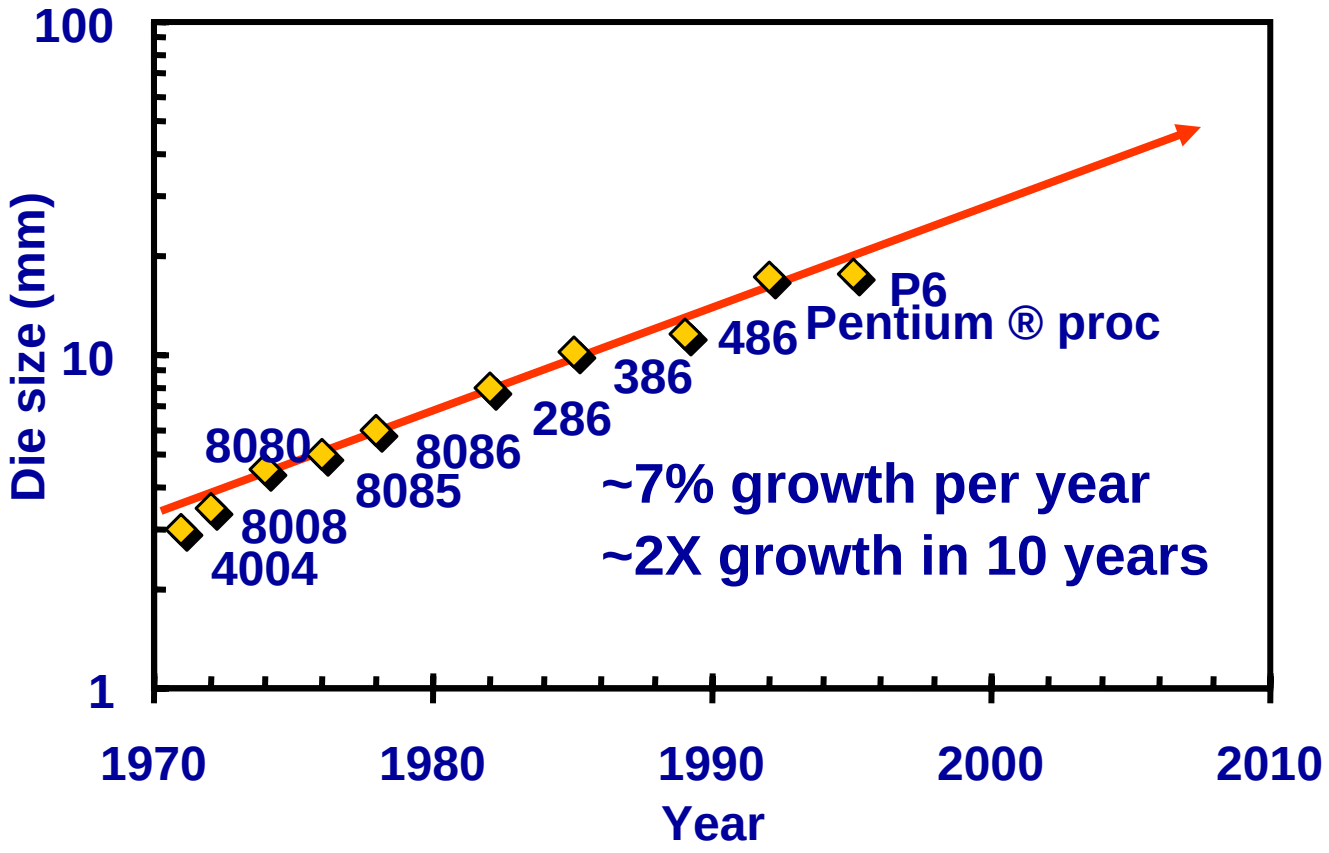
Moore's Law In Microprocessors



Transistors on Lead Microprocessors double every 2 years

Courtesy: Intel

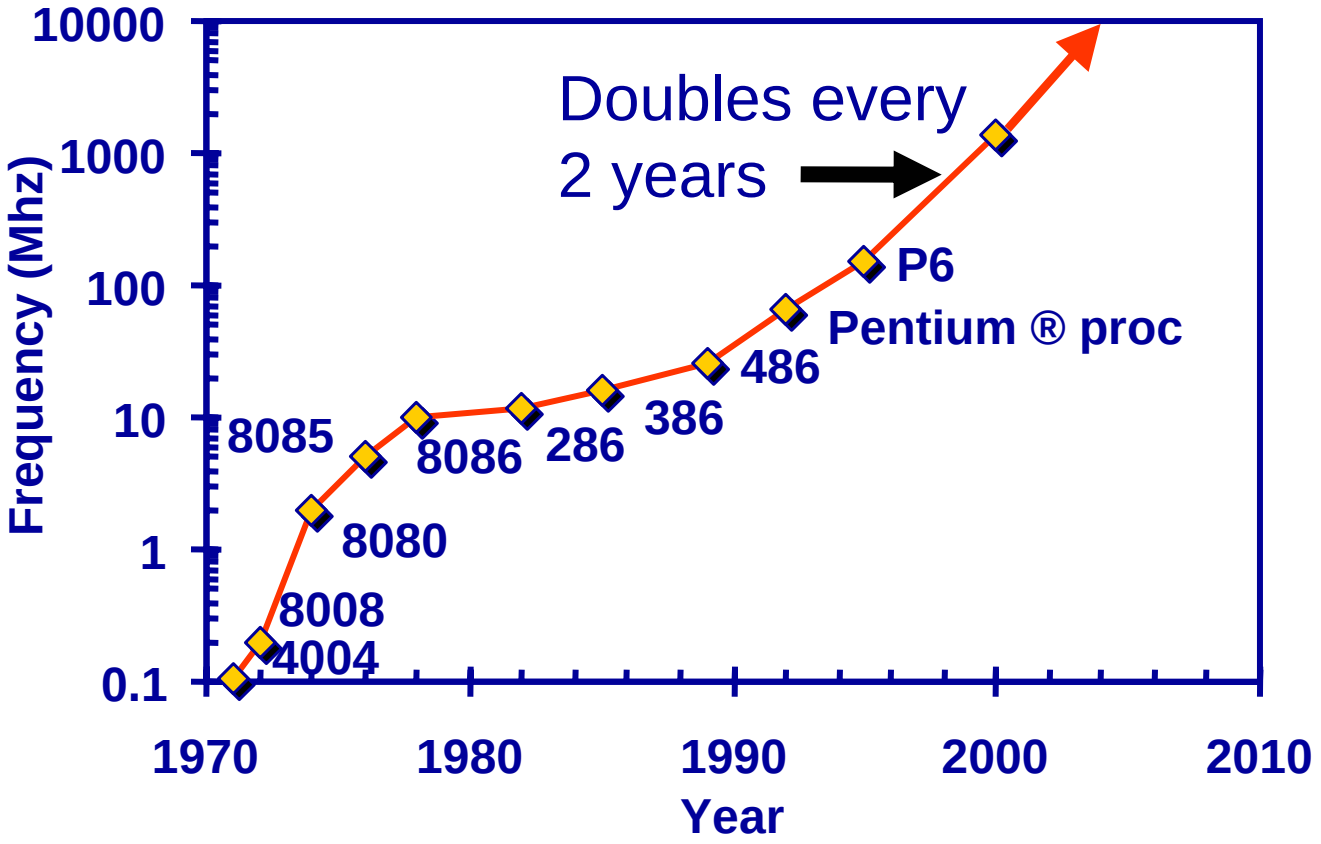
Die Size Growth



Die size grows by 14% to satisfy Moore's Law

Courtesy, Intel

Frequency

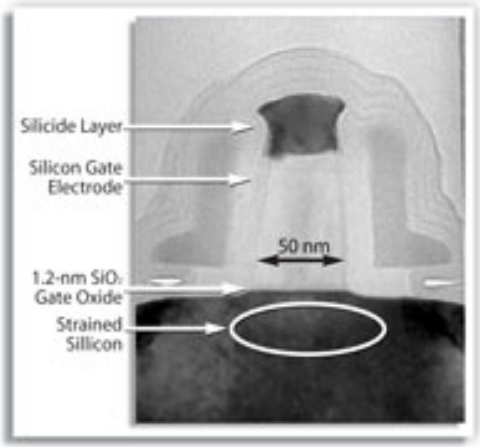


Lead Microprocessors frequency doubles every 2 years

Courtesy, Intel

Technology Drivers

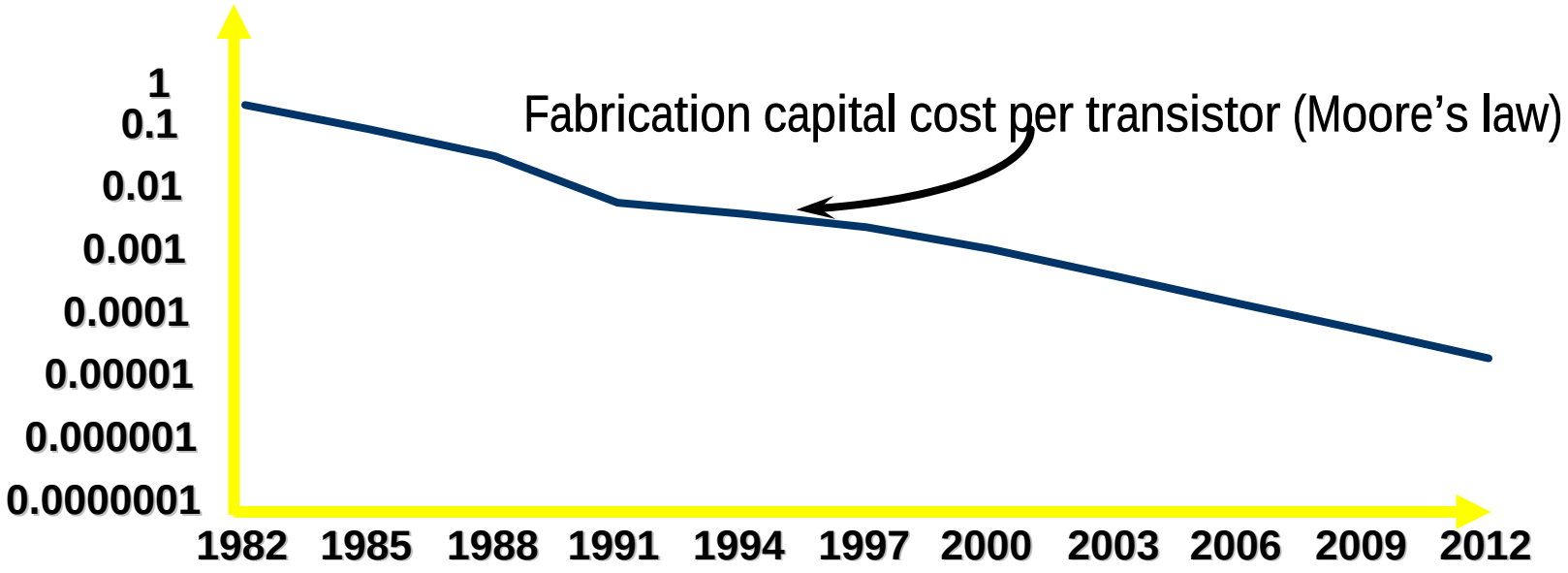
- **Decreasing lithographic feature size, e.g. measured by the transistor gate length**
  - 0.13 um, 90nm, 65nm, … 10nm (?)
- **Increasing wafer size**
  - 8 in. diameter … 12 in.
- **Increasing number of metal interconnect layers**
  - 6 … 8 … 9 …



Courtesy, Intel

Cost Scaling

- **Cost per transistor scales down**
  - Approximately constant cost per wafer to manufacture
  - About $2,000 - $4,000 per wafer
- **But cost to first chip scales up!**
  - Design cost increases with transistor count
  - Mask cost increases with each new family



Adapted from: Dr. Paul D Franzon, www.ece.ncsu.edu/erl/faculty/paulf.html

Adapted from: Digital Integrated Circuits A Design Perspective, 2nd Edition, Jan M. Rabaey et al. © 2003 Prentice Hall/Pearson

Cost Scaling : Some Examples

Chip	Metal layers	Line width	Wafer cost	Def./ cm^2	Area mm^2	Dies/wafer	Yield	Die cost
386DX	2	0.90	$900	1.0	43	360	71%	$4
486 DX2	3	0.80	$1200	1.0	81	181	54%	$12
Power PC 601	4	0.80	$1700	1.3	121	115	28%	$53
HP PA 7100	3	0.80	$1300	1.0	196	66	27%	$73
DEC Alpha	3	0.70	$1500	1.2	234	53	19%	$149
Super Sparc	3	0.70	$1700	1.6	256	48	13%	$272
Pentium	3	0.80	$1500	1.5	296	40	9%	$417

Semiconductor Roadmap

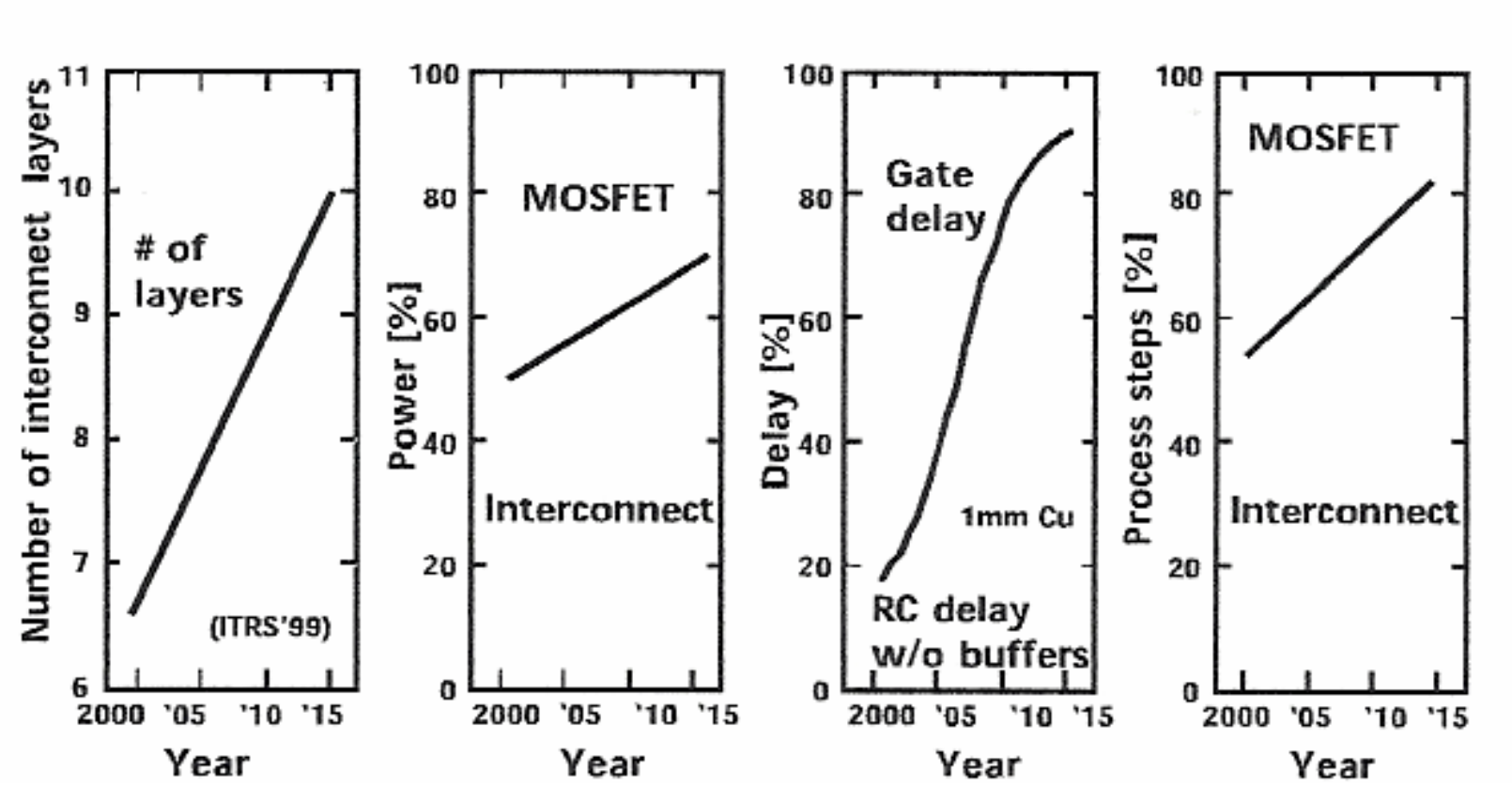
- **Projections for 'leading edge' ASIC/MPU: (www.itrs.net)**

Table 1i High-Performance MPU and ASIC Product Generations and Chip Size Model—Near-term Years

Year of Production	*2005*	*2006*	*2007*	*2008*	*2009*	*2010*	*2011*	*2012*	*2013*
DRAM ½ Pitch (nm) (contacted)	*80*	*70*	*65*	*57*	*50*	*45*	*40*	*36*	*32*
MPU/ASIC Metal 1 (M1) ½ Pitch (nm) (f)	*90*	*78*	*68*	*59*	*52*	*45*	*40*	*36*	*32*
MPU Physical Gate Length (nm)	*32*	*28*	*25*	*23*	*20*	*18*	*16*	*14*	*13*
ASIC									
ASIC usable Mtransistors/cm^2 (auto layout)	225	283	357	449	566	714	899	1,133	1,427
ASIC max chip size at production (mm^2) (maximum lithographic field size)	*858*	*858*	*858*	*858*	*858*	*858*	*858*	*858*	*858*
ASIC maximum functions per chip at production (Mtransistors/chip) (fit in maximum lithographic field size)	1,928	2,430	3,061	3,857	4,859	6,122	7,713	9,718	12,244
Chip Frequency (MHz)									
On-chip local clock [1]	5,204	6,783	9,285	10,972	12,369	15,079	17,658	20,065	22,980
Chip-to-board (off-chip) speed (high-performance, for peripheral buses)[2]	3,125	3,906	4,883	6,103	7,629	9,536	11,920	14,900	18,625
Maximum number wiring levels—maximum [3]	15	15	15	16	16	16	16	16	17
Maximum number wiring levels—minimum [3]	11	11	11	12	12	12	12	12	13

Scaling Trends: Interconnect

- The impact of the wiring increases with each generation



Challenges in Digital Design

"Microscopic Problems"

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

Everything Looks a Little Different

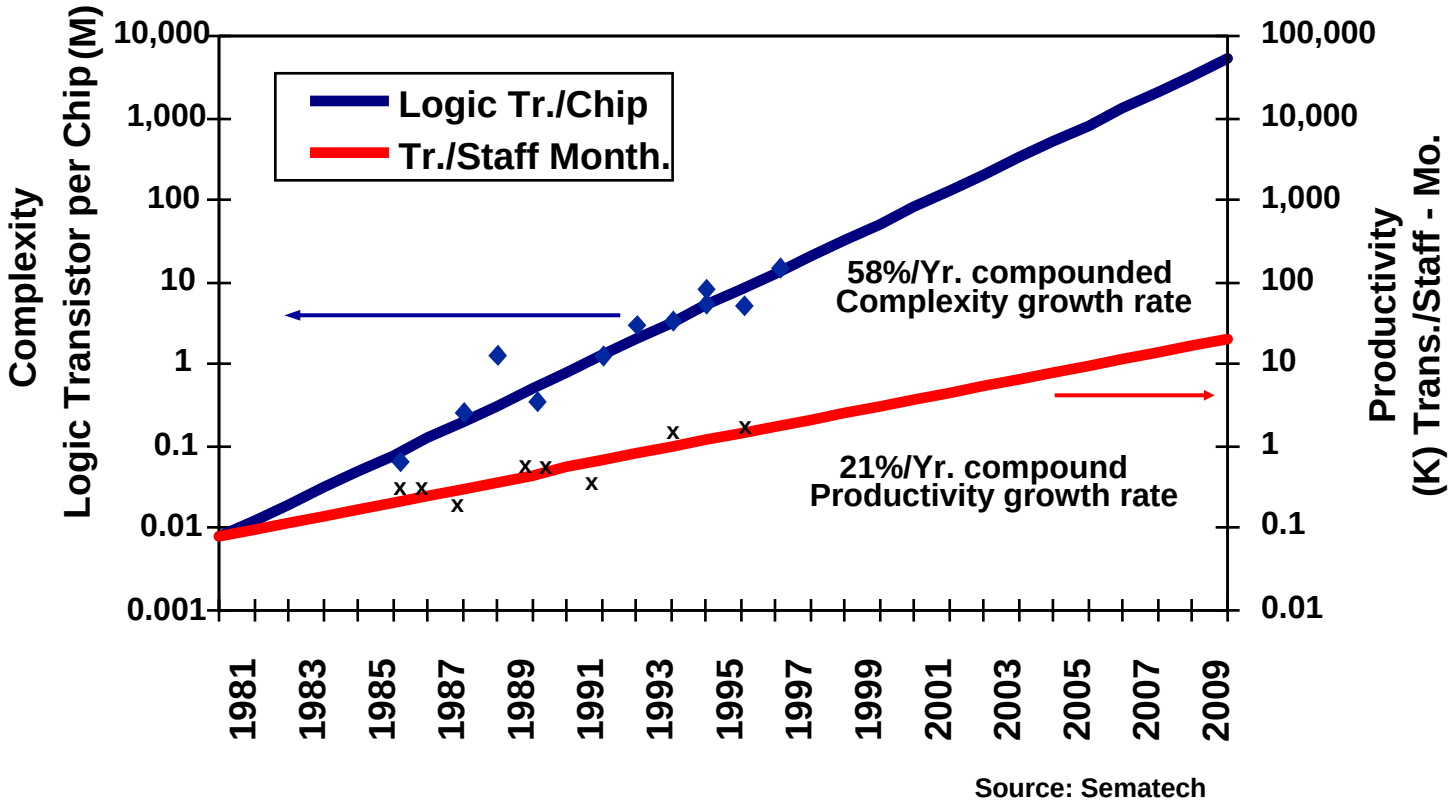
?

"Macroscopic Issues"

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

…and There's a Lot of Them!

Productivity Trends



Complexity outpaces design productivity

ASICs vs. What?

- **Application Specific Integrated Circuit**
  - A chip designed to perform a particular operation as opposed to General Purpose integrated circuits
  - An ASIC is generally NOT software programmable to perform a wide variety of different tasks
- **An ASIC will often have an embedded CPU to manage suitable tasks**
- **An ASIC may be implemented as an FPGA**
  - Sometimes considered a separate category
  - Hybrid implementation: programmable logic and application specific blocks
- **Platform-based Implementations: System on a Chip**

ASICs vs What?

- **General Purpose Integrated Circuits**

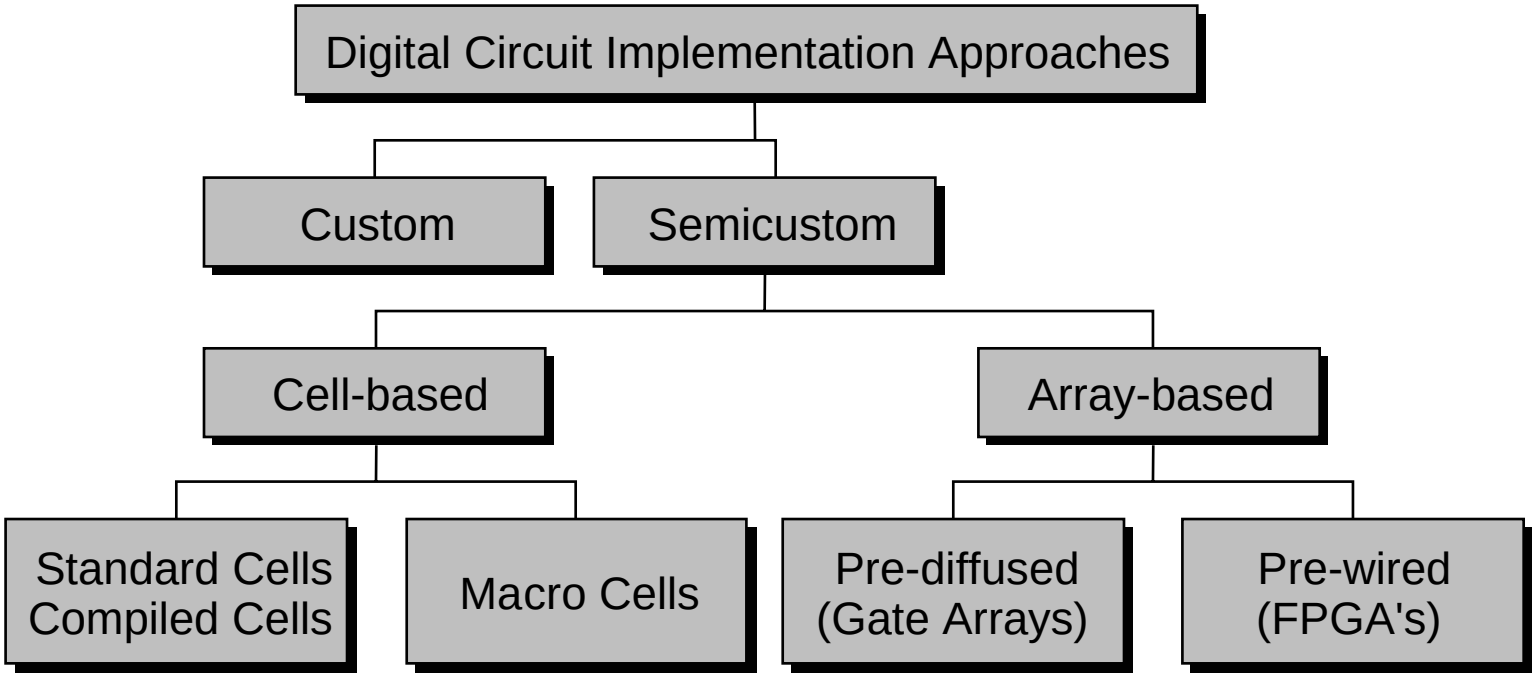
  Examples:

  - Programmable microprocessors (e.g. Intel Pentium Series, Motorola HC-11)
    - Used in PCs to washing machines
  - Programmable Digital Signal Processors (e.g. TI TMS 320 Series)
    - Used in many multimedia, sensor processing and communications applications
  - Memory (DRAMs, SRAMs, etc.)

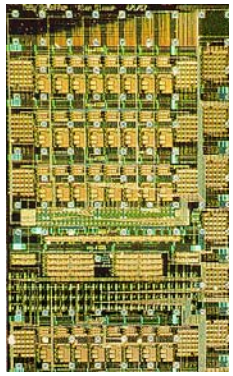
ASIC Examples

- Video processor to decode or encode MPEG-2 digital TV signals
- Low power dedicated DSP/controller / convergence device for mobile phones
- Encryption processor for security
- Many examples of graphics chips
- Network processor for managing packets, traffic flow, etc.

Implementation Choices

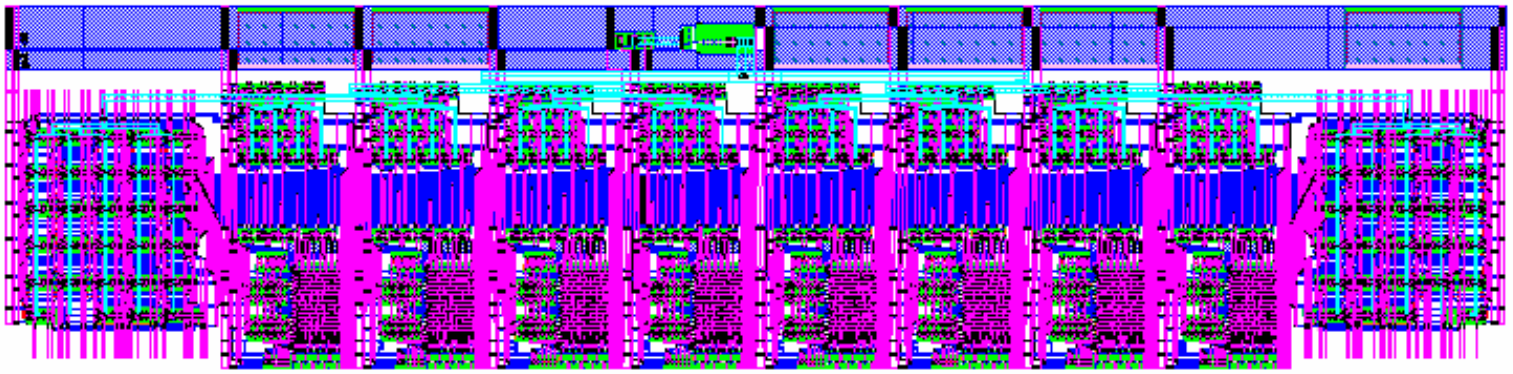


Full-Custom ASICs



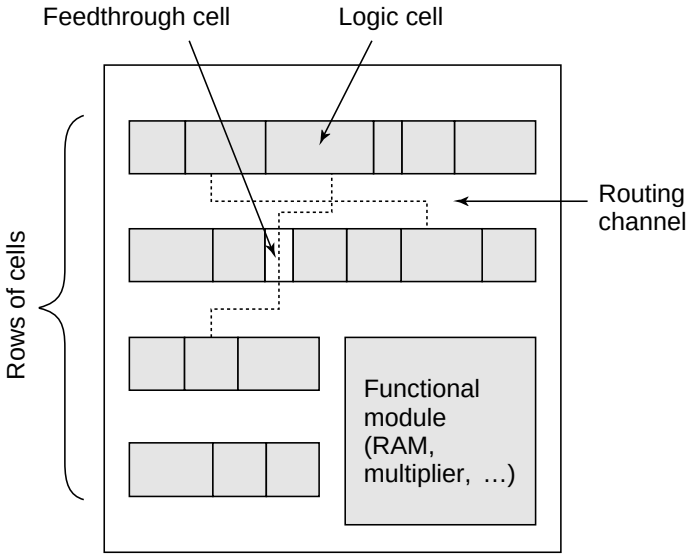
- Every transistor is designed and drawn by hand
- Typically only way to design analog portions of ASICs
- Gives the highest performance but the longest design time
- Full set of masks required for fabrication

- **Custom digital circuits are used in applications where their performance premium justifies their design cost**

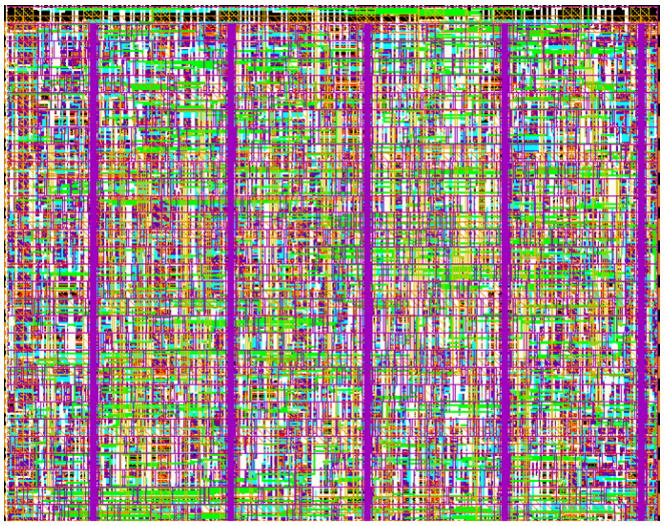


Standard Cell Based Design

- or 'Cell Based IC' (CBIC) or 'semi-custom'
- **Standard Cells** are custom designed and then inserted into a **library**
- These cells are then used in the design by being placed in rows and wired together using 'place and route' CAD tools
- Some standard cells, such as RAM and ROM cells, and some datapath cells (e.g. a multiplier) are tiled together to create **macrocells**



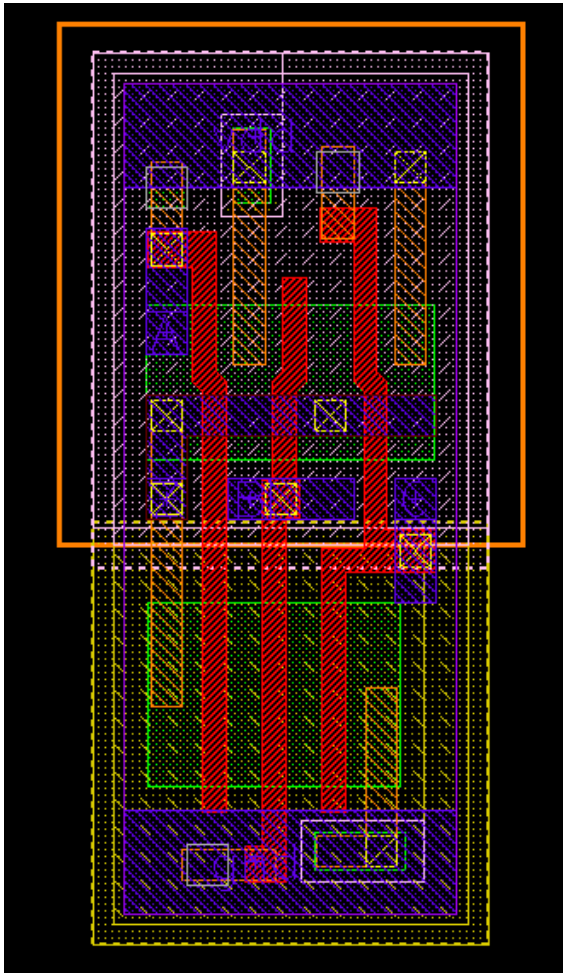
Older Technologies:
Routing channel required due to small number of interconnect layers



Newer Technologies:
Cell-structure hidden under interconnect layers

Standard Cells

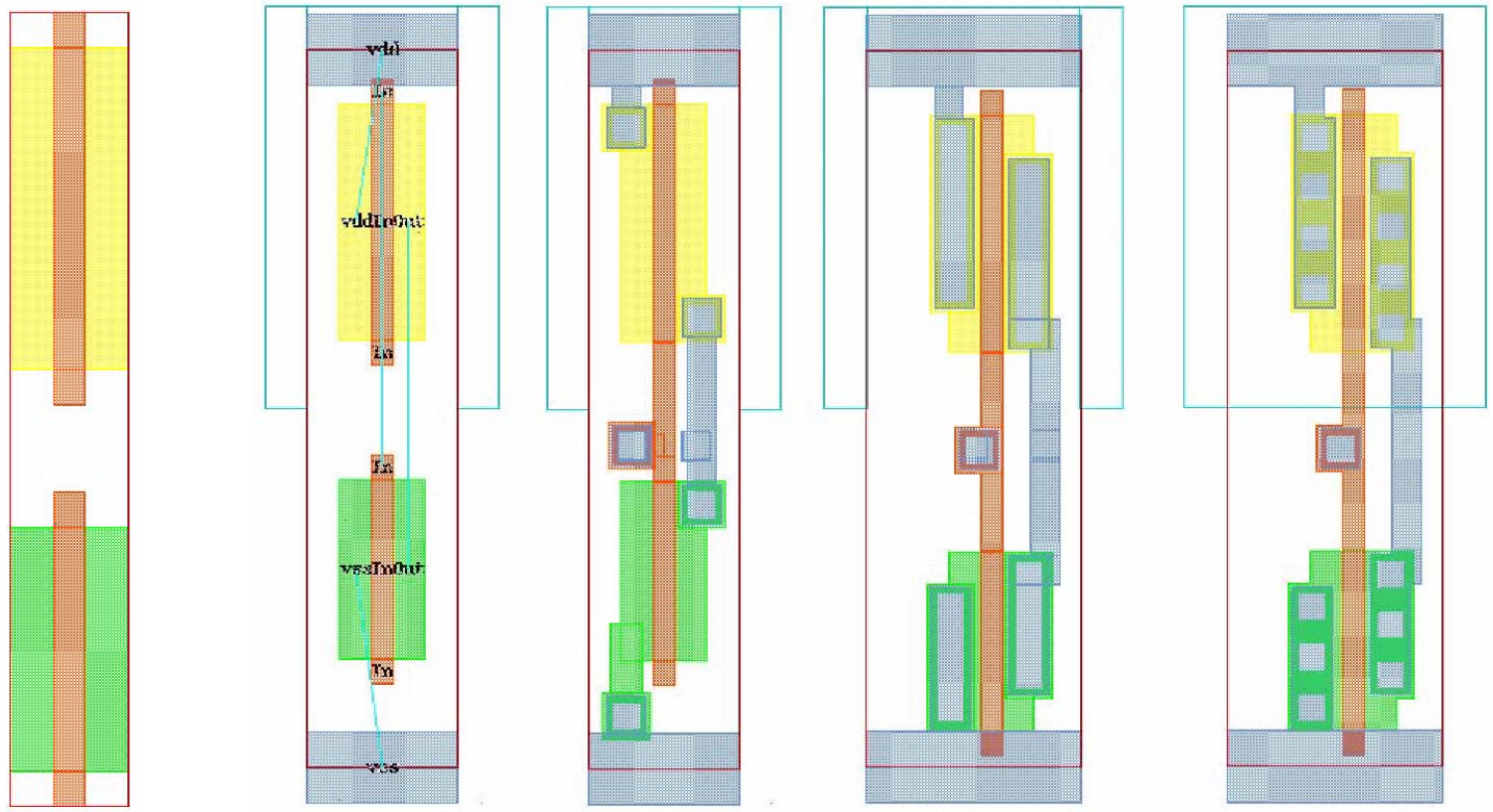
- **Example:**



Path	1.2V - 125°C	1.6V - 40°C
$In1—t_{pLH}$	$0.073+7.98C+0.317T$	$0.020+2.73C+0.253T$
$In1—t_{pHL}$	$0.069+8.43C+0.364T$	$0.018+2.14C+0.292T$
$In2—t_{pLH}$	$0.101+7.97C+0.318T$	$0.026+2.38C+0.255T$
$In2—t_{pHL}$	$0.097+8.42C+0.325T$	$0.023+2.14C+0.269T$
$In3—t_{pLH}$	$0.120+8.00C+0.318T$	$0.031+2.37C+0.258T$
$In3—t_{pHL}$	$0.110+8.41C+0.280T$	$0.027+2.15C+0.223T$

3-input NAND cell
(from ST Microelectronics):
C = Load capacitance
T = input rise/fall time

Standard Cells: Compiled Cells

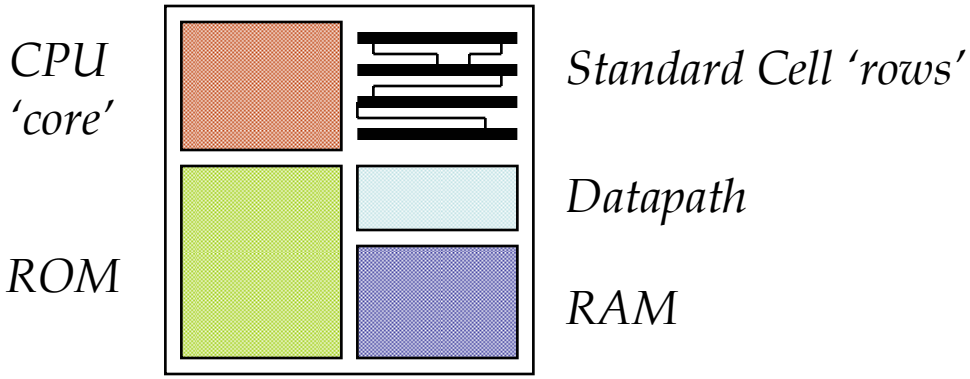


Initial transistor geometries | Placed transistors | Routed cell | Compacted cell | Finished cell

Courtesy Acadabra

Standard Cell Based Design (Contd.)

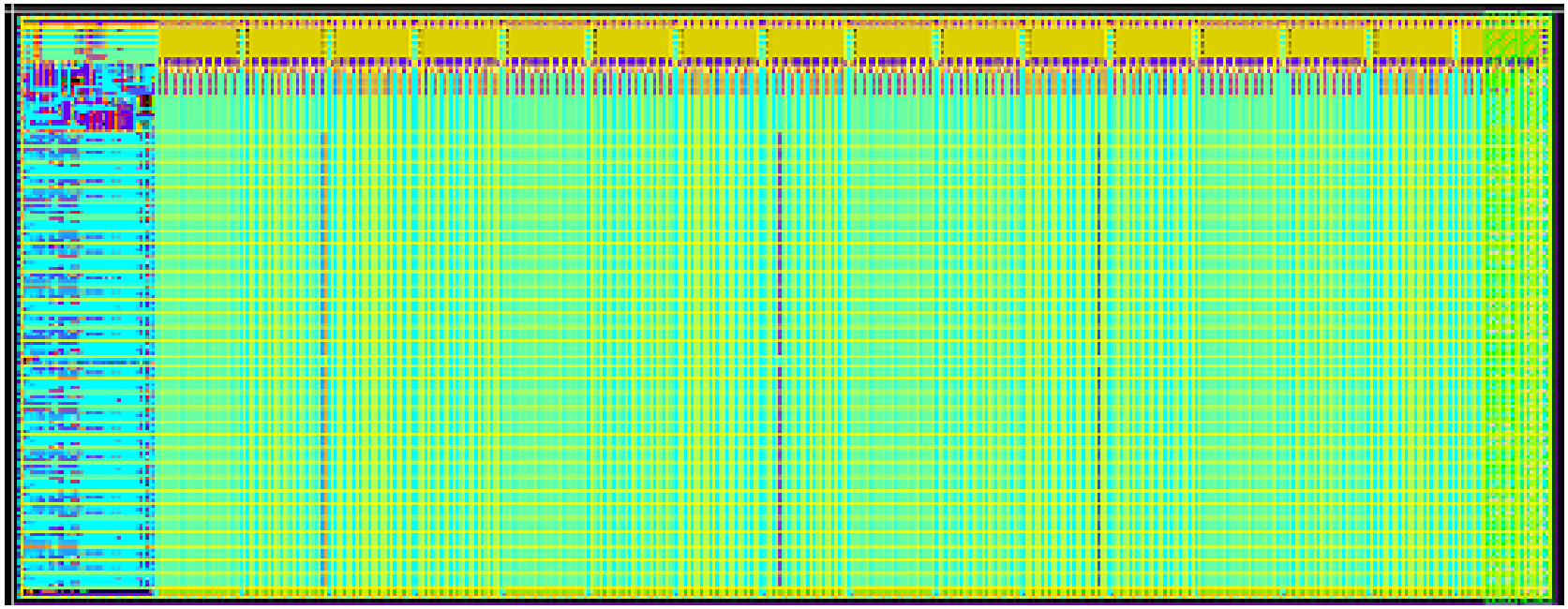
- **Sample ASIC Floorplan**



  - Standard Cell designs are usually synthesized from an RTL (Register Transfer Language) description of the design
  - Full set of masks (22+) still required

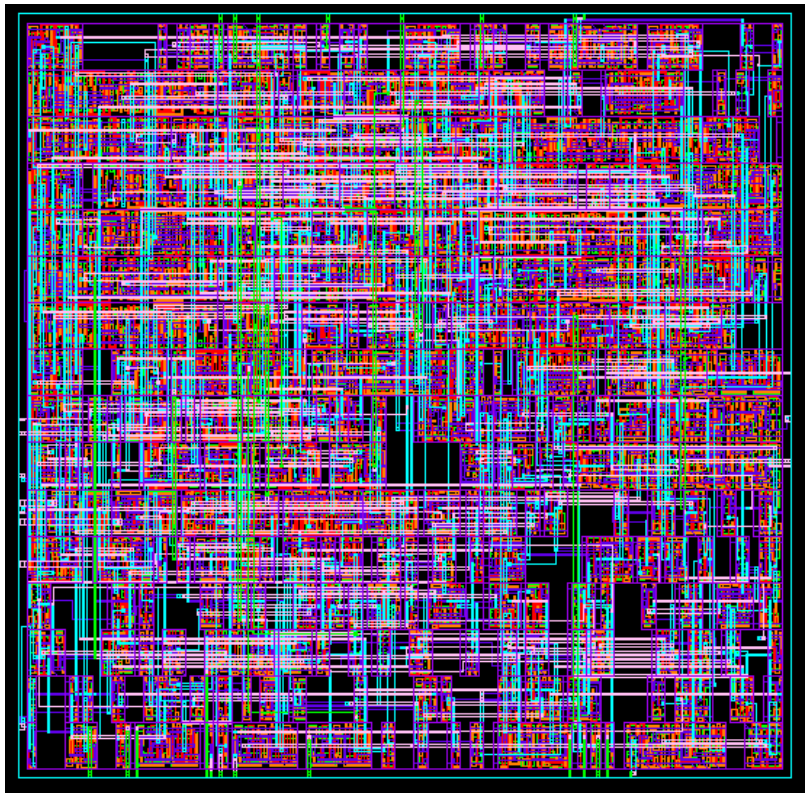
- **Fabless semiconductor company model**
  - Company does design only. Fab performed by another company (e.g. TSMC, UMC, IBM, Philips, LSI).
  - Back-end (place and route, etc.) might be performed at that company or with their assistance

Macro Modules

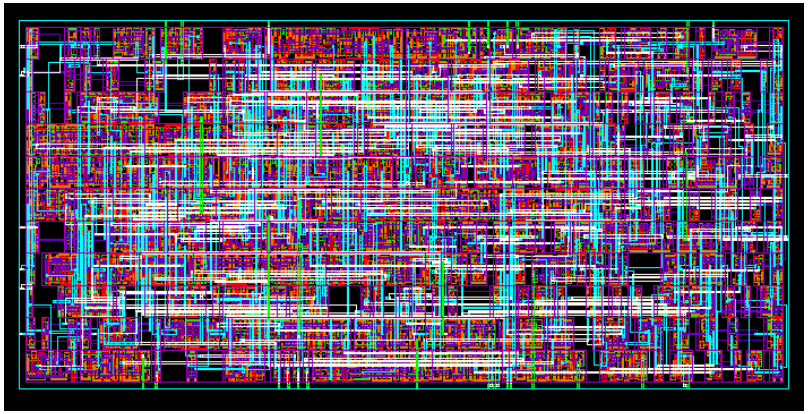


256×32 (or 8192 bit) SRAM
Generated by hard-macro module generator

"Soft" Macromodules

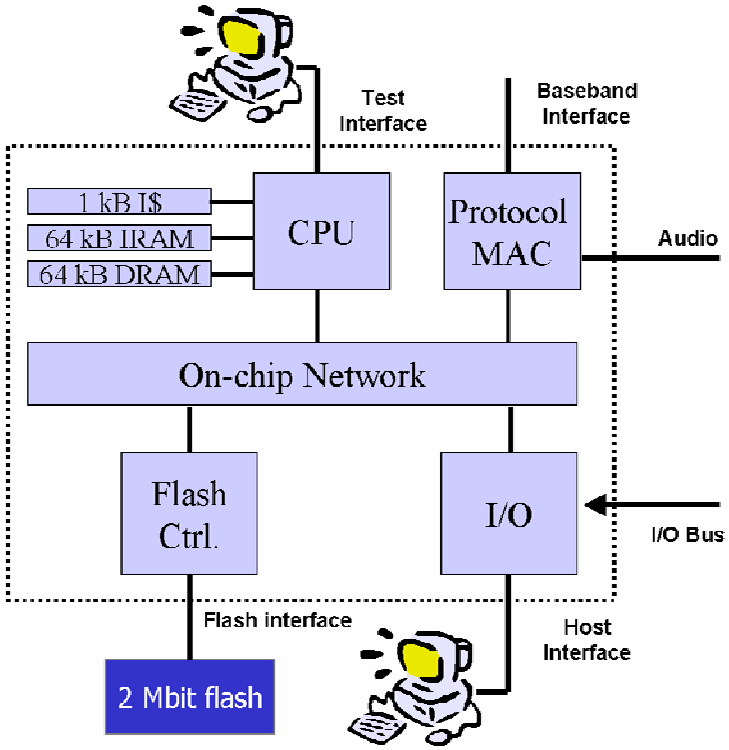


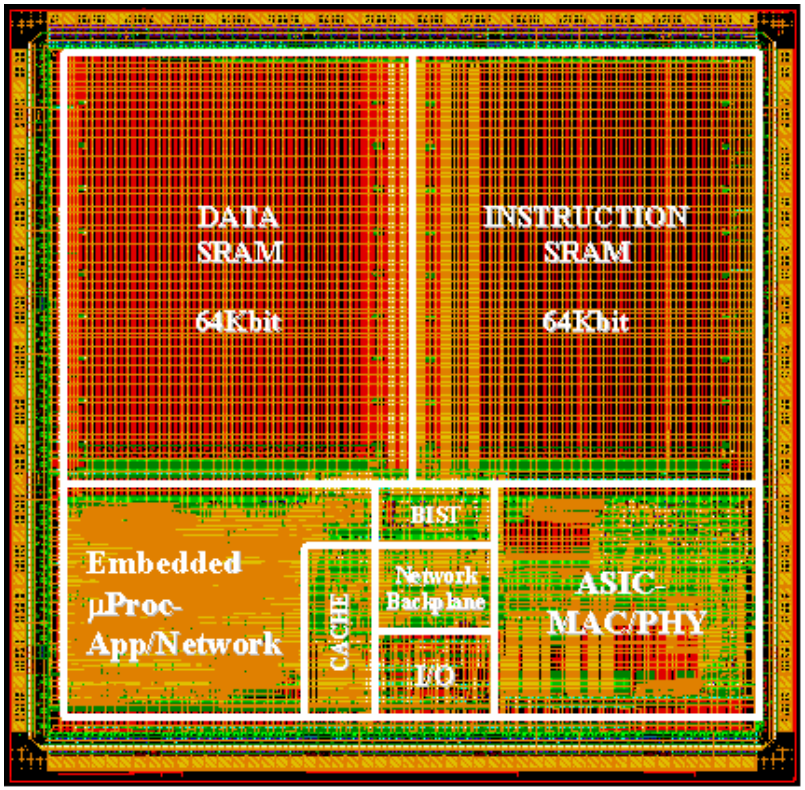
```
string mat = "booth";
directive (multtype = mat);
output signed [16] Z = A * B;
```



Synopsys DesignCompiler

"Intellectual Property (IP)"



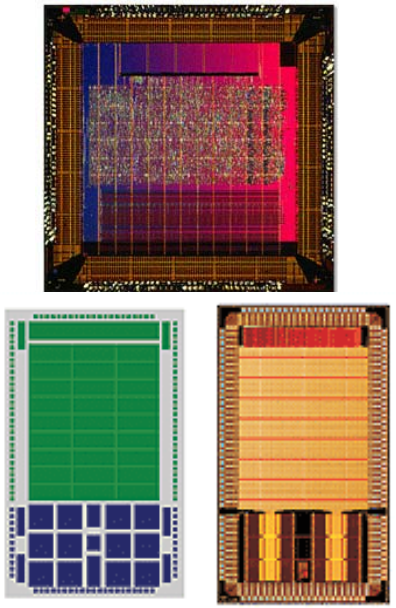


A Protocol Processor for Wireless

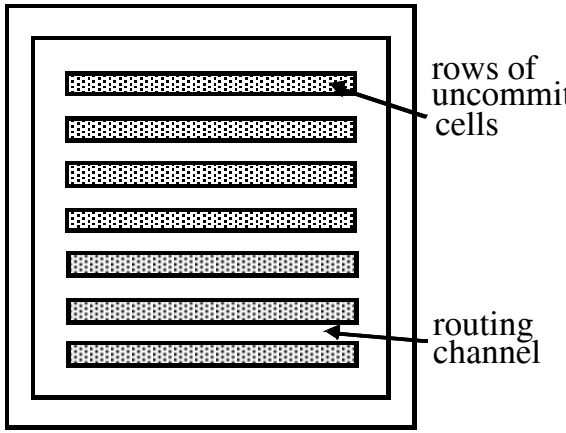
Gate-Array Based Design: Sea-of-Gates

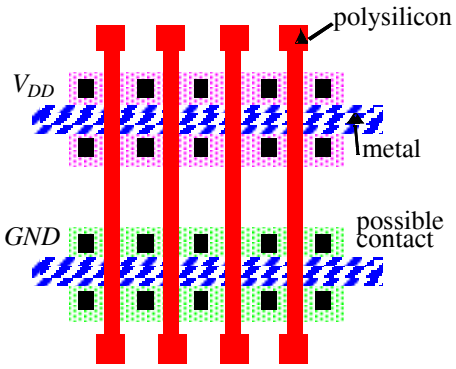
- In a gate array, the transistors level masks are fully defined and the designer can not change them
- The design instead programs the wiring and vias to implement the desired function
- Gate array designs are slower than cell-based designs but the implementation time is faster as less time must be spent in the factory
- RTL-based methods and synthesis, together with other CAD tools, are often used for gate arrays

- Examples:
  - **Chip Express**
    - Wafers built with sea of macros + 4 metal layers
    - 2 metal layers customized for application
    - Only 4 masks!
  - **Triad Semiconductor**
    - Analog and Digital Macros
    - 1 metal layer for customization (2 week turnaround)

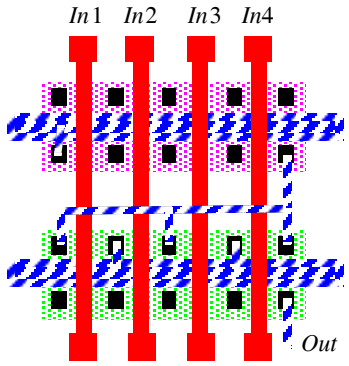


Gate-Array Based Design: Sea-of-Gates



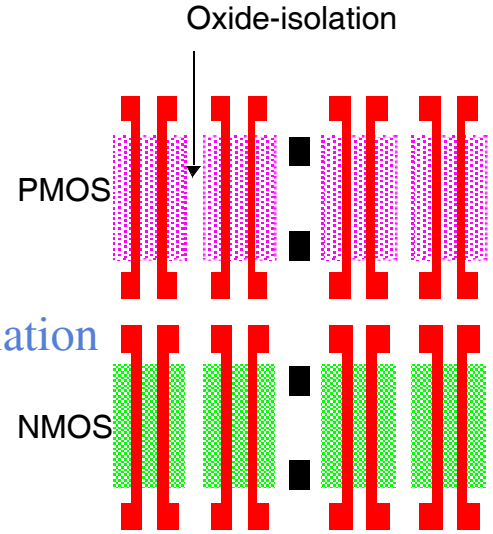


Uncommited Cell

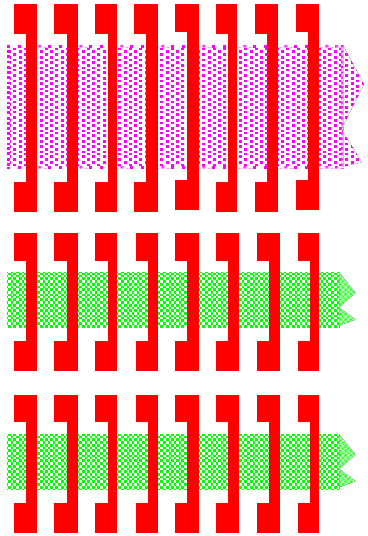


Committed Cell (4-input NOR)

Sea-of-Gates Primitive Cells



Using oxide-isolation



Using gate-isolation

Prewired Arrays

Classification of prewired arrays (or field-programmable devices):

- **Based on Programming Technique**
  - Fuse-based (program-once)
  - Non-volatile EPROM based
  - RAM based
- **Programmable Logic Style**
  - Array-Based
  - Look-up Table
- **Programmable Interconnect Style**
  - Channel-routing
  - Mesh networks

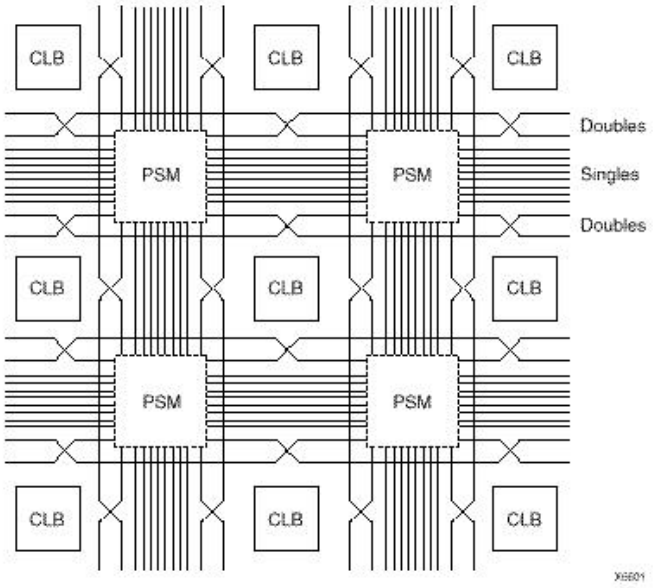
Programmable Logic Devices (PLDs and FPGAs)

- FPGA= Field Programmable Gate Array
- Are off-the-shelf ICs that can be programmed by the user to capture the logic
- There are no custom mask layers so final design implementation is a few hours instead of a few weeks
- Simple PLDs are used for simple functions.
- FPGAs are increasingly displacing standard cell designs
- Capable of capturing 100,000+ designed gates
- High power consumption
- High per-unit cost
- FPGAs are also slow (< 100 MHz)

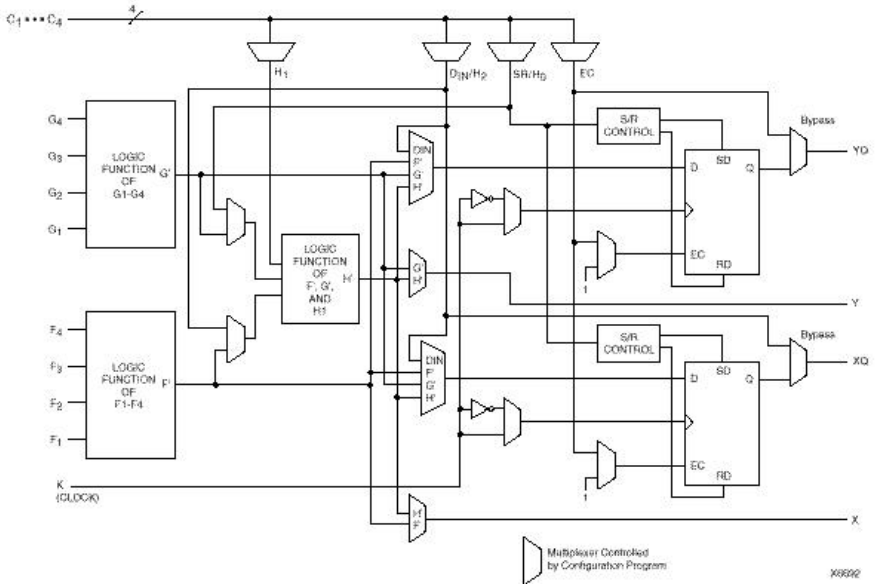
FPGAs (Contd.)

- **Sample internal architecture:**
  - Store logic in look-up table (RAM)
  - Programmable interconnect

Programmable Interconnect Array

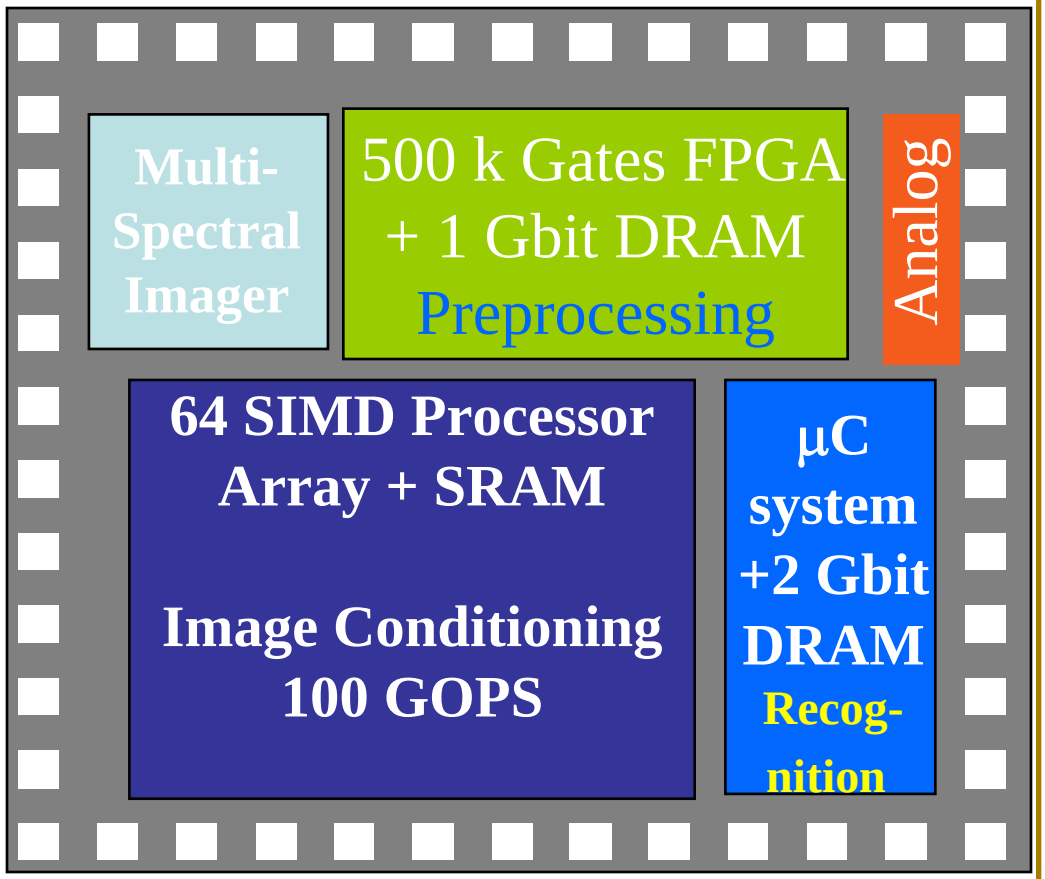


Configurable Logic Block (CLB):



Design Complexity

- **System-on-a-Chip**
  - Embedded applications where cost, performance, and energy are the real issues!
  - DSP and control intensive
  - Mixed-mode
  - Combines programmable and application-specific modules
  - Software plays crucial role

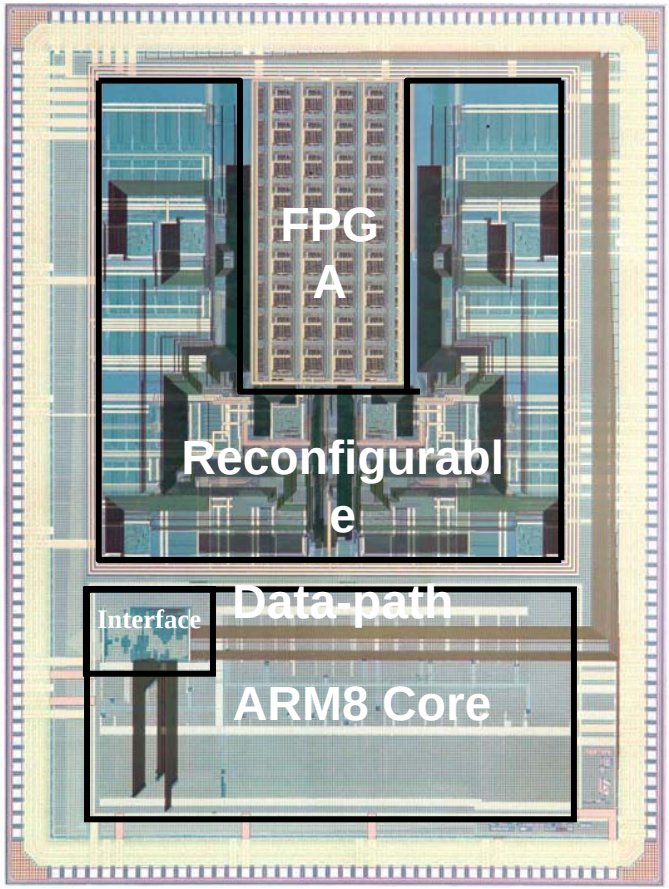


Addressing Design Complexity

- **Reuse existing components/designs**
  - Standard Cells
  - IP Blocks
  - Architecture
  - IC
- **IP can be broadly classified as**
  - Hard IP
    - Defined at the mask layout level in a particular process
  - Firm IP
    - Will normally have a specific or generic gate net-list
  - Soft IP
    - Defined at the RTL level
- **Platform-based Design**
  - H/W architectures, S/W modules, programmable components, network architecture

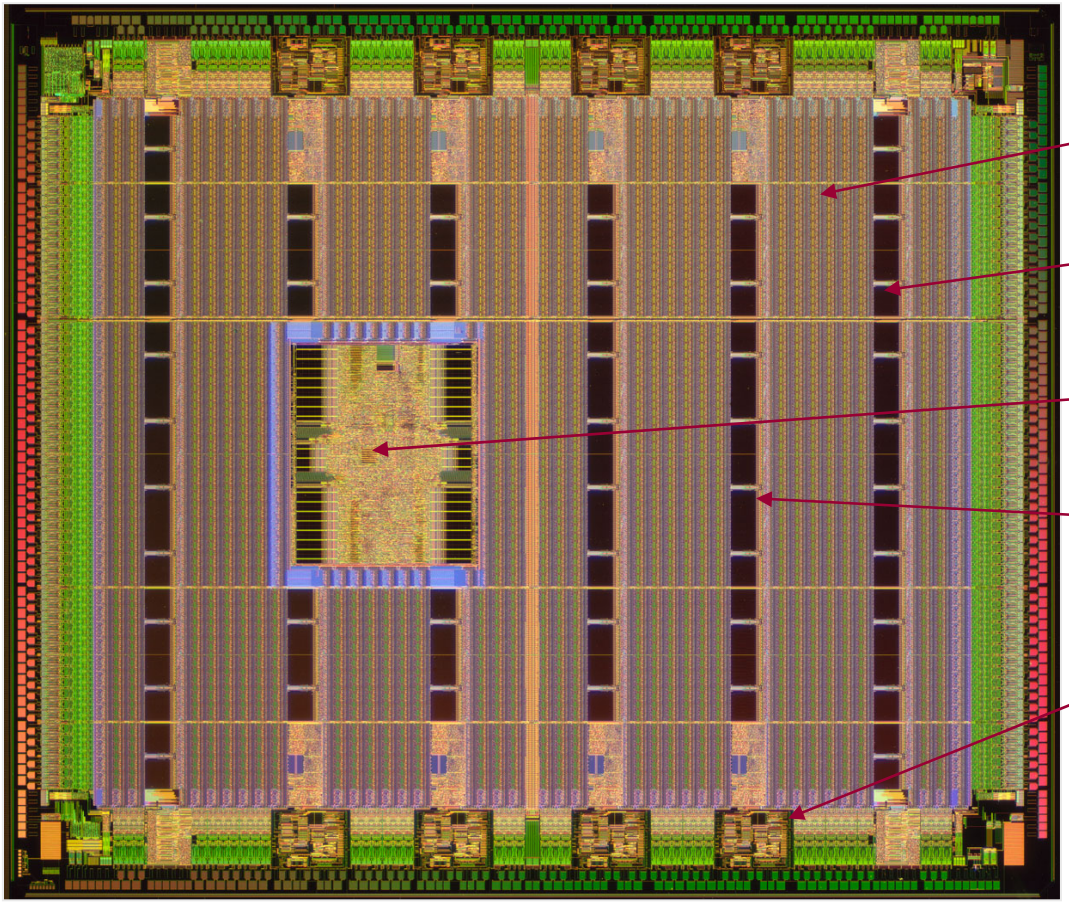
Hybird Implementation Platforms

- Example platform for wireless applications: Berkeley Pleiades Processor



Hybird Implementation Platforms

- **Example platform for wireless applications: Xilinx Vertex-II Pro**



Courtesy Xilinx

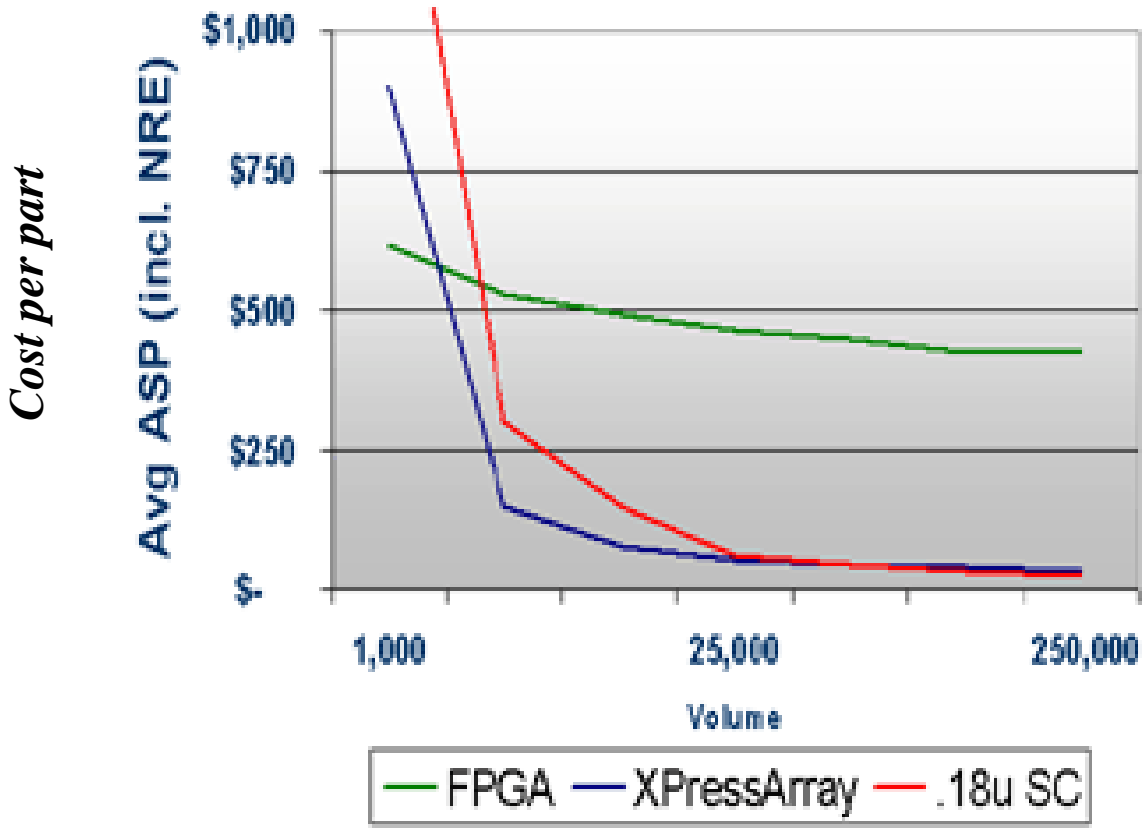
Comparison of Design Methods

Table 8.4 Comparison of CMOS design methods

Design Method	Non-recurring Engineering	Unit Cost	Power Dissipation	Complexity of Imple-mentation	Time to Market	Perfor-mance	Flexibility
Microprocessor/ DSP	low	medium	high	low	low	low	high
PLA	low	medium	medium	low	low	medium	low
FPGA	low	high	medium	medium	low	medium	high
Gate Array/SOG	medium	medium	low	medium	medium	medium	medium
Cell Based	high	low	low	high	high	high	low
Custom Design	high	low	low	high	high	very high	low
Platform Based	high	low	low	high	high	high	medium

Total Cost Calculation

- Example



ASICs and FPGAs

- **Market currently dominated by standard cell ASICs and FPGAs**
  - Ideally standard cell designs would be used for higher volume applications that justify the NRE
- **Many consider FPGAs separate from ASICs**

  Why?
  - Different level of design skills required, especially in "back end" (place and route or physical design)
  - Reduced level of verification required before "sending to factory"
    - Again reduces sophistication required of team
  - Low-cost (barrier) of entry
    - Often different, lower cost Design Automation (CAD) tools
  - Lower performance
- **However, front-end design (RTL coding) is virtually identical for each implementation style**
- **Sometimes FPGA done first and standard cell ASIC done later**

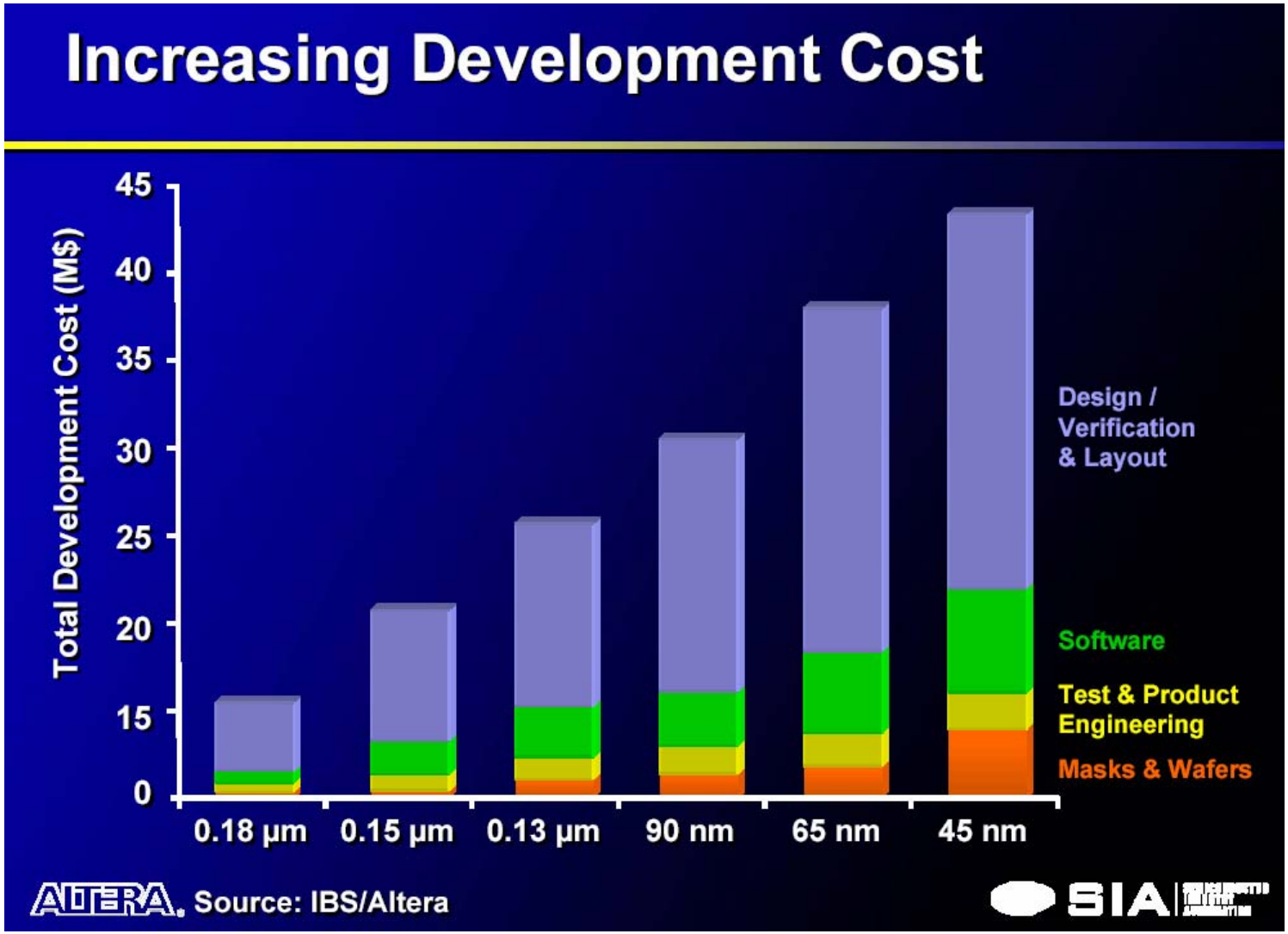
Future Issues

- **Increased cost of custom fab**
  - First chip run will cost over $2M for 90 nm
  - Multiproject wafers
- **Increased cost of design**
  - Must be addressing > $1B market to justify a new chip run
- **Globalization**
  - Time-to-market and other competitive issues

Future Issues (Contd.)

- **Trends**
  - Increased use of FPGA and Gate Arrays
  - Increased use of 'platform' solutions
    - Multi-core embedded CPU + ASIC accelerators
    - Configurable systems
    - Existing designs ('IP or Intellectual Property')
  - Increased use of SystemVerilog, SystemC and other system modeling tools
  - Complexity shifting from design to logical and performance verification
    - Logical verification = function; Performance = speed
  - Cost to first silicon getting so high that the total addressable market must be very large and product risk low

Design Cost



Adapted from: Dr. Paul D Franzon, www.ece.ncsu.edu/erl/faculty/paulf.html

Summary

- Over the next ten years, product growth will be driven by:
  - Underlying technology push
  - High demand for graphics, multimedia and wireless connectivity
  - Insidious insertion of electronics and computers into our everyday lives
- Many of the resulting products will require specialized silicon chips to meet performance (speed/size/weight/power/cost) demands - ASICs
- To match this product need, the capability of a silicon CMOS chip will continue doubling every 2-3 years until after 2015.
  - To sell a product at $300-$1,000, it can only include one high value chip
    - Thus product performance is determined by the performance of that one chip
  - AND talk about planned obsolescence!!
- ASIC styles include full custom (for analog) and RTL-based design: Cell based (semi-custom), Gate Array or FPGA implementation
- ASIC design methodology includes logic, timing, and physical design
  - Unfortunately, design productivity is not keeping up with chip performance growth