CMOS VLSI IC Design

- A decent understanding of all tasks required to design and fabricate a chip takes years of experience

Commonly used keywords

- INTEGRATED CIRCUIT (IC)
  many transistors on one chip

- VERY LARGE SCALE INTEGRATION (VLSI)
  very many transistors (> 10000 gates) on one chip

- COMPLEMENTARY METAL OXIDE SEMICONDUCTOR (CMOS) TECHNOLOGY
  cheap, high integration density, low power

Integrated Circuits

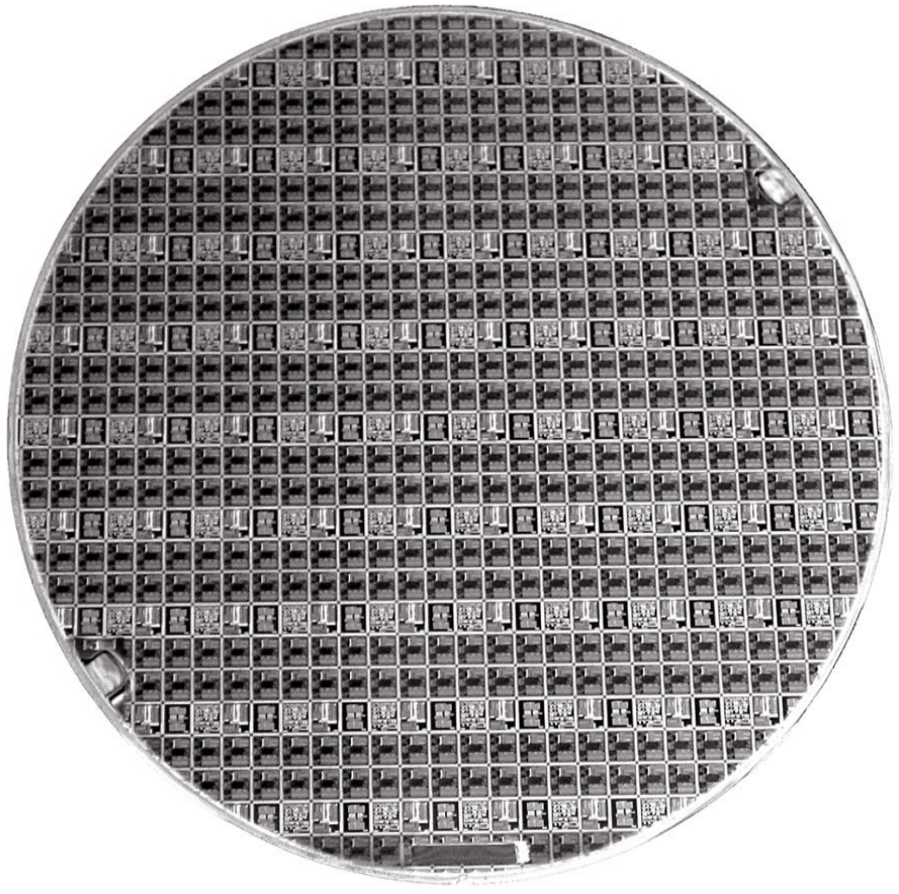


FIG 1.70 Processed 8-inch wafer

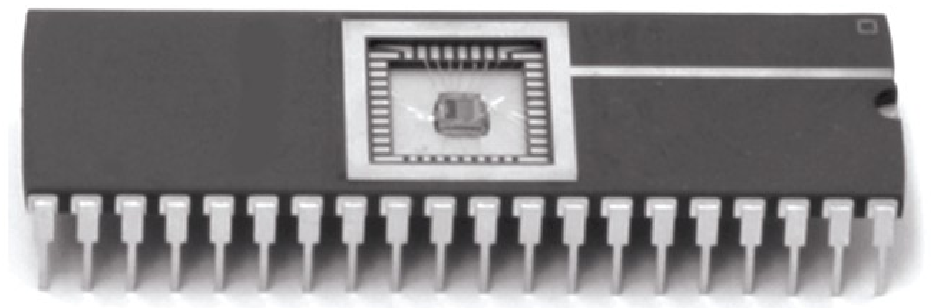
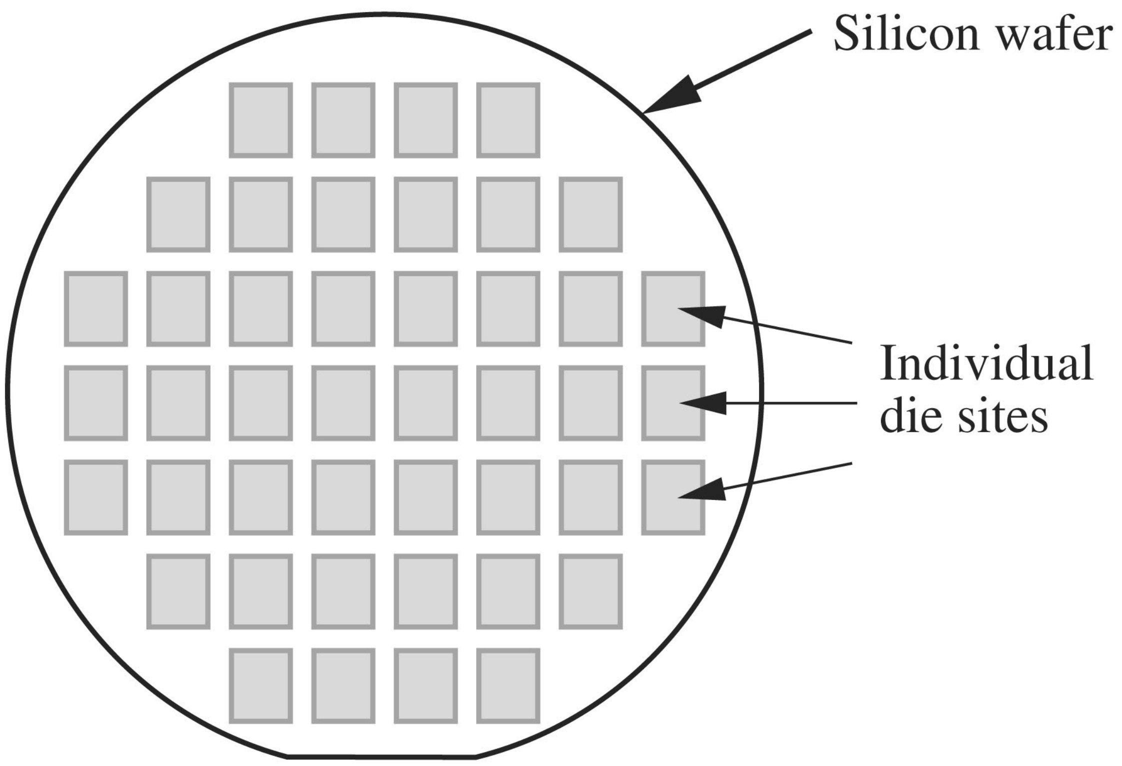


FIG 1.71 Chip in a 40-pin dual-inline package

The Silicon Wafer

Figure 2.12:
Silicon wafer with individual die sites



Packaging the Chip

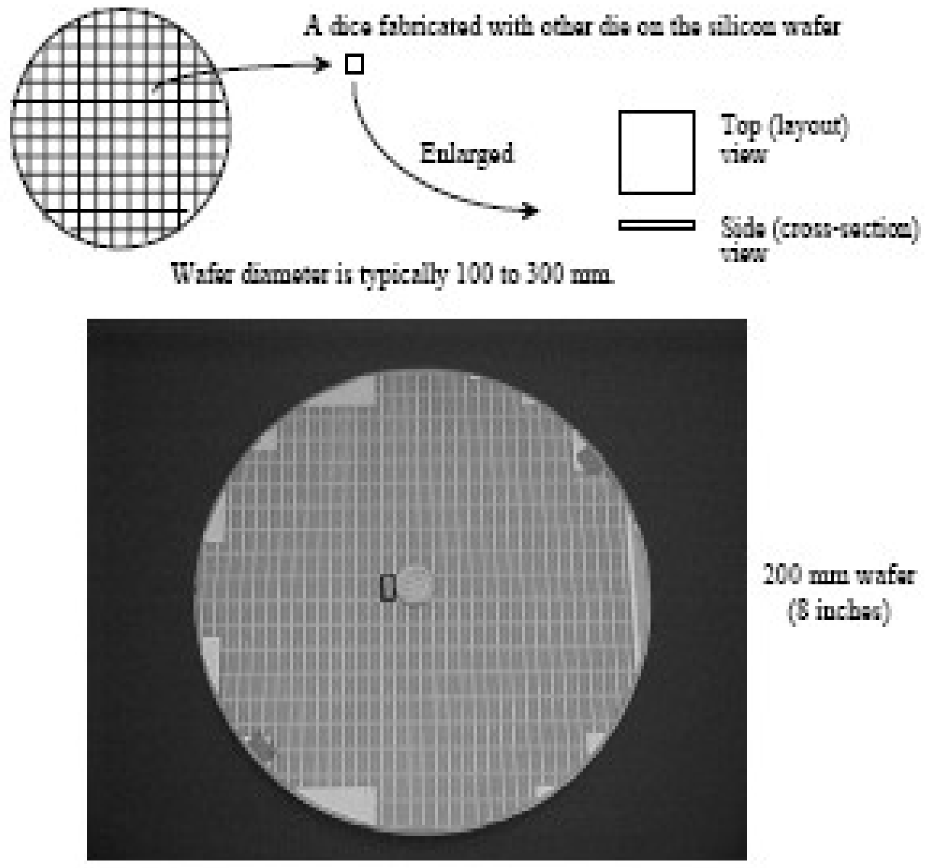

A dice fabricated with other die on the silicon wafer
Enlarged
Top (layout) view
Side (cross-section) view
Wafer diameter is typically 100 to 300 mm.
200 mm wafer (8 inches)

Figure 1.2 CMOS integrated circuits are fabricated on and in a silicon wafer.

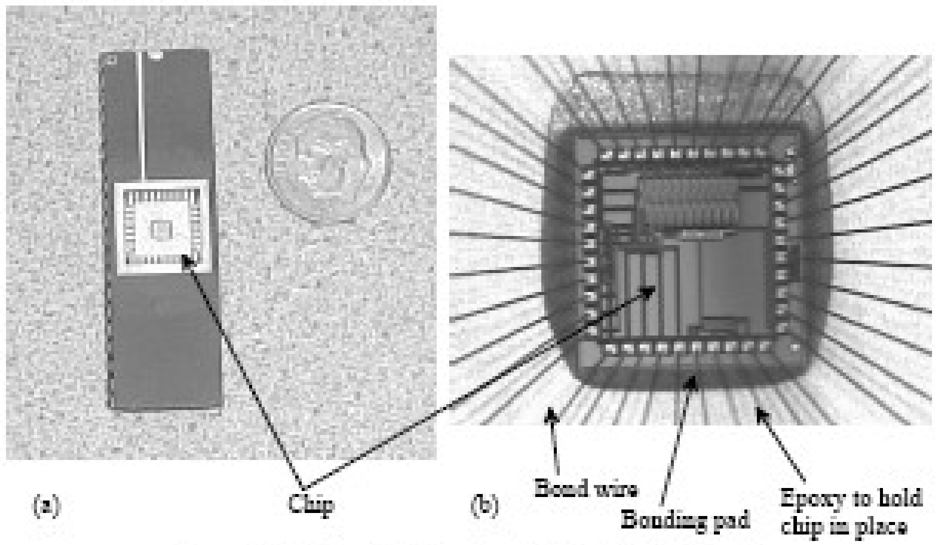


Figure 1.3 How a chip is packaged (a) and (b) a closer view.

Common Packages

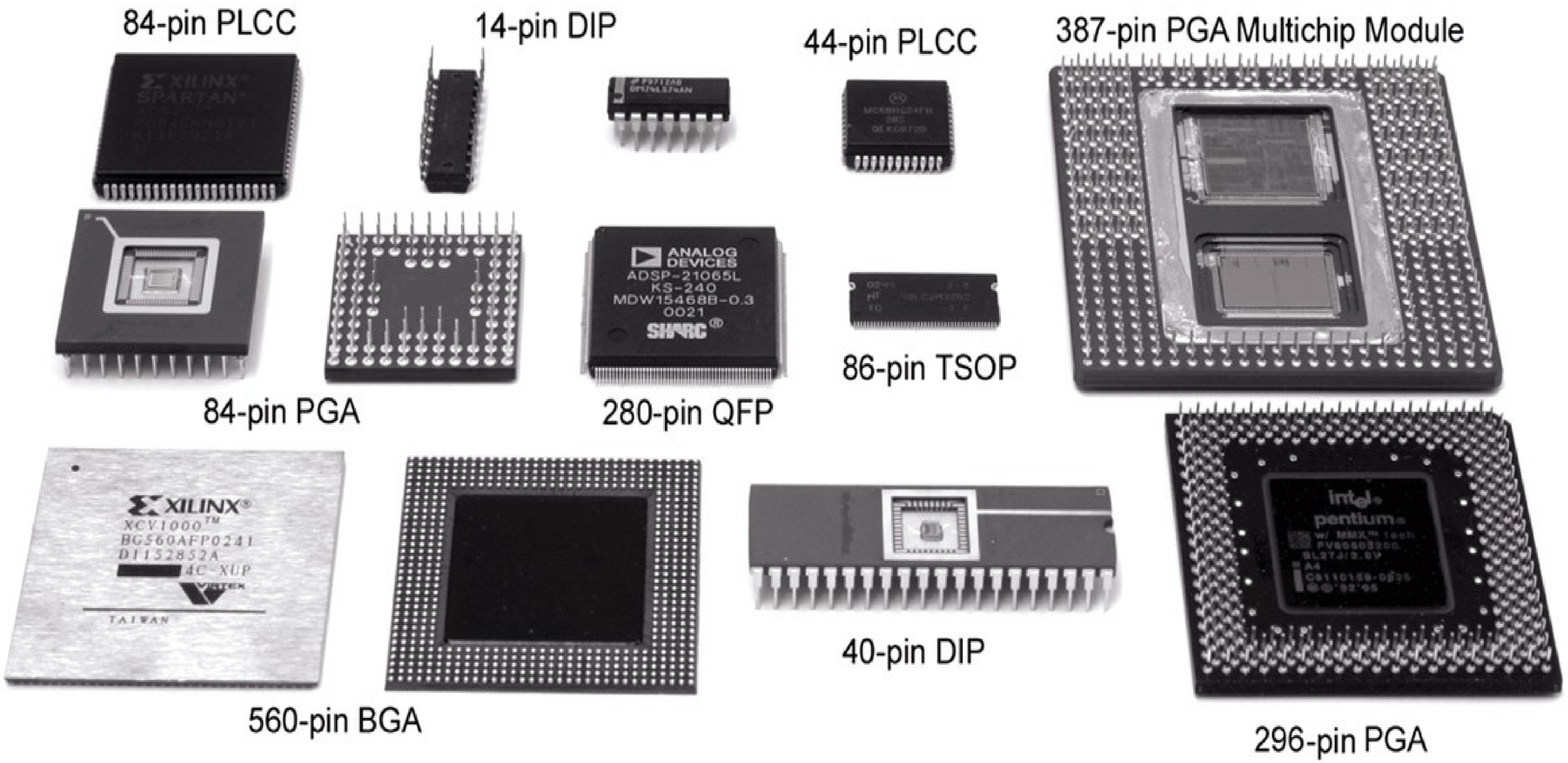


FIG 12.1 Integrated circuit packages. © 2003 Harvey Mudd College. Reprinted with permission.

Impact of ICs industry

Integrated Circuits enabled today's way of life

10^{18} transistors manufactured in 2003

(100 million for every human on the planet)

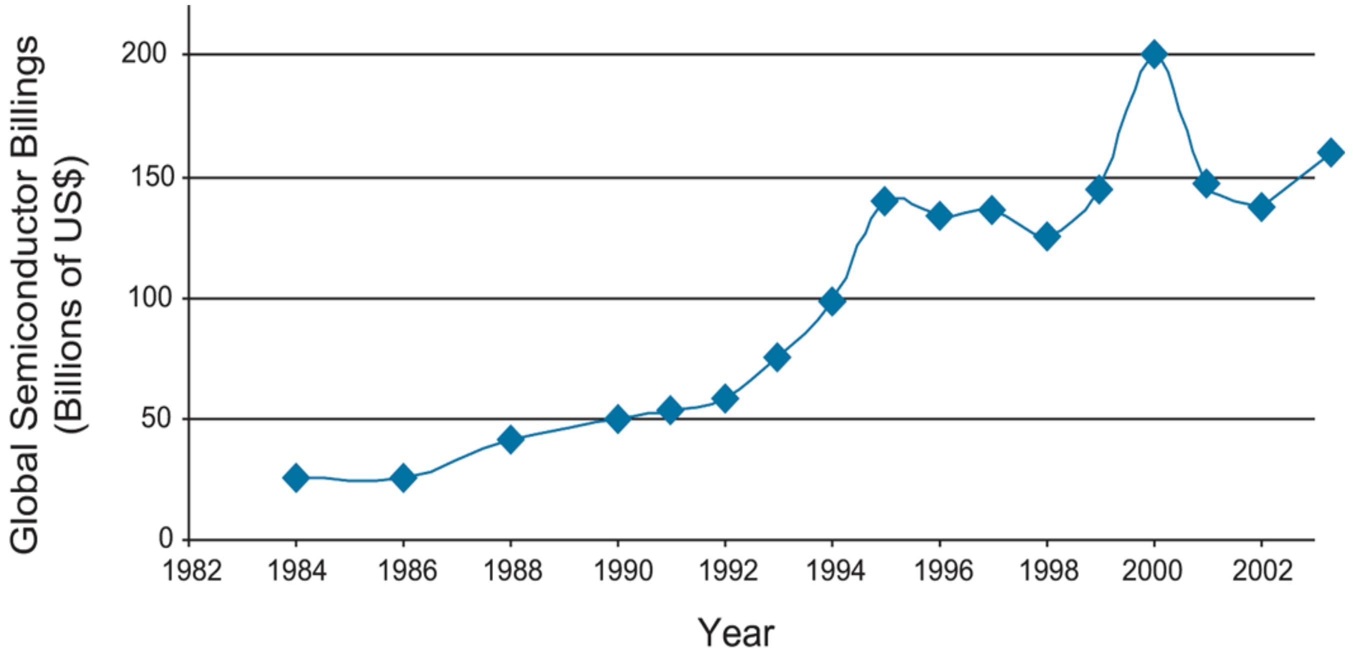
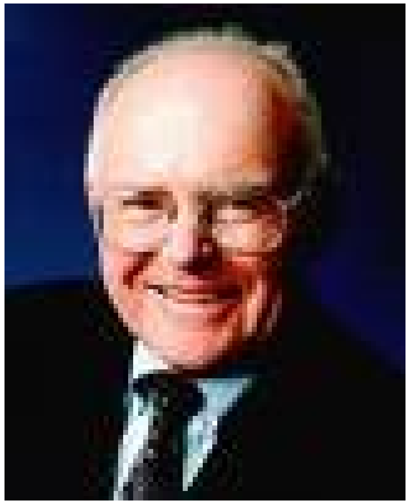


FIG 1.1 Size of worldwide semiconductor market

Source: Semiconductor Industry Association.

Moore's Law



- In 1963 Gordon Moore predicted that as a result of continuous miniaturization transistor count would double every 18 months
- 53% compound annual growth rate over 45 years (No other technology has grown so fast so long)

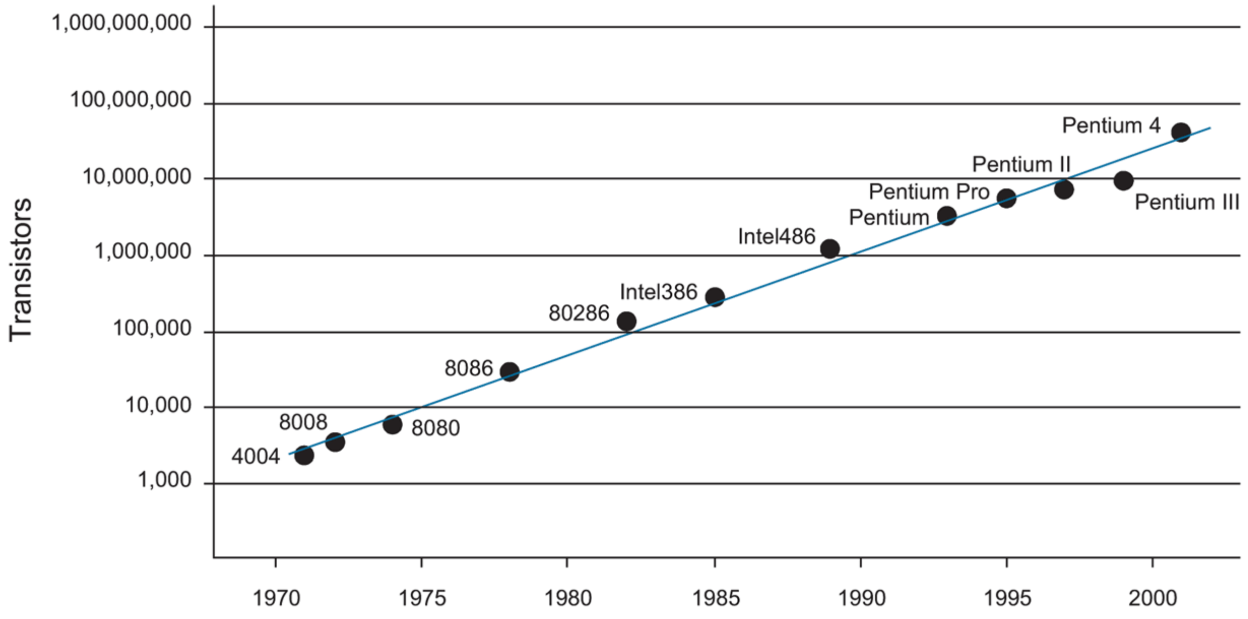


FIG 1.4 Transistors in Intel microprocessors [Intel03]

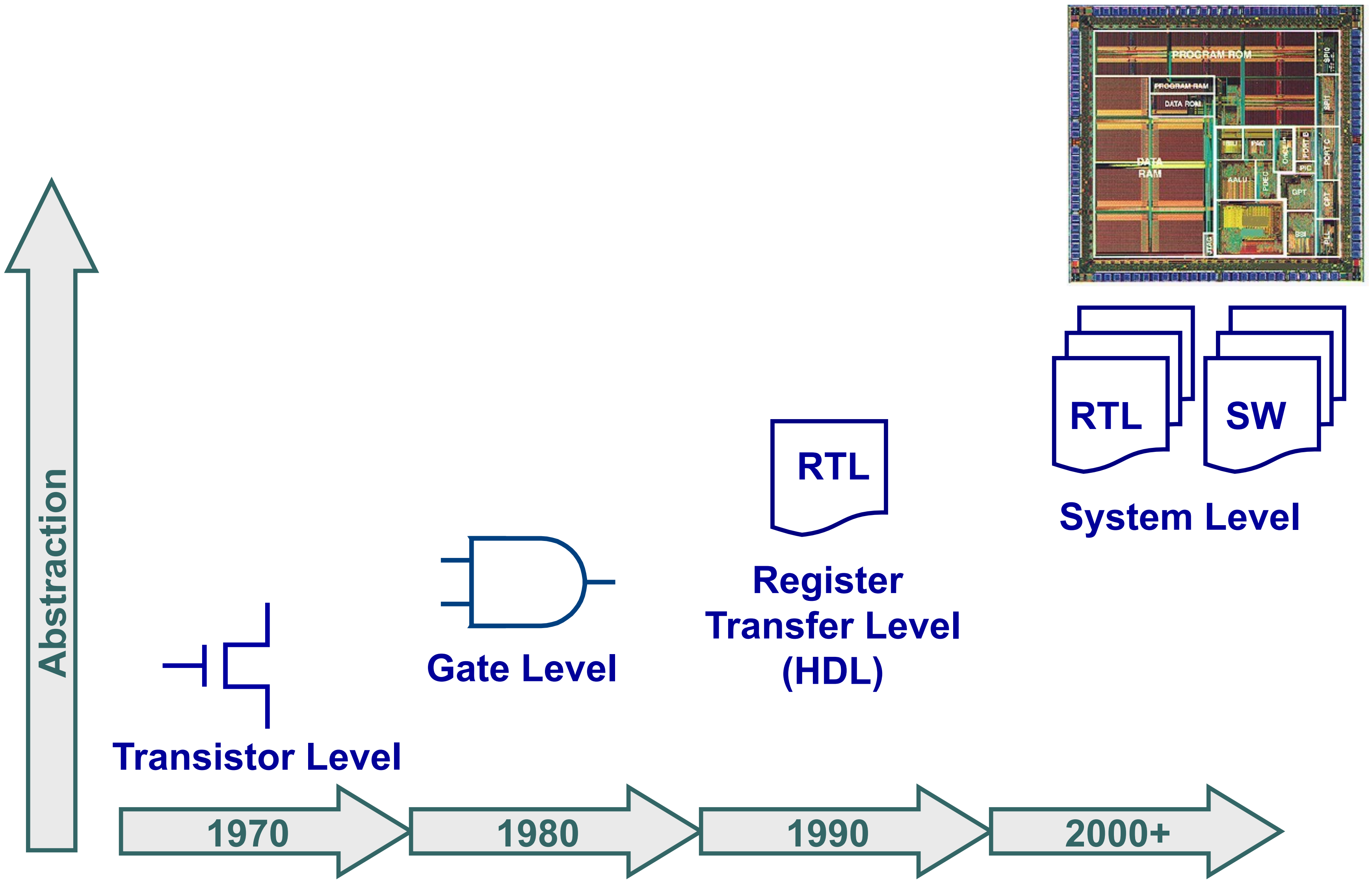
Transistors have become:
- smaller
- faster
- consume less power
- cheaper to manufacture

Challenges

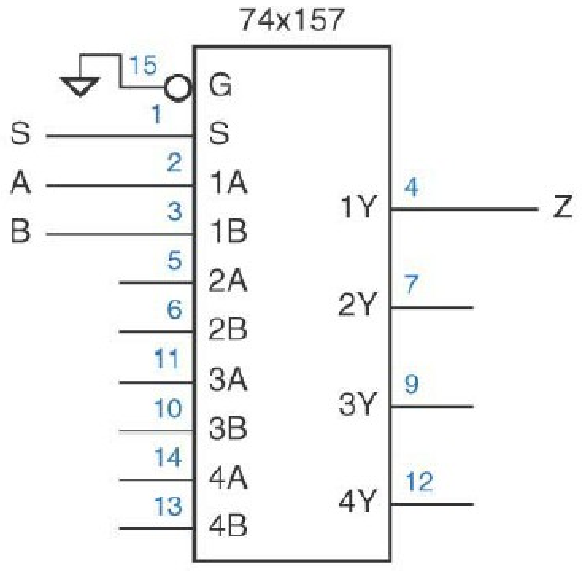
- The greatest challenge in modern VLSI design is managing **system complexity**

- Strategies used to cope with Complexity
  - Abstraction
  - Structured Design Approach
  - Design Flow

Design Abstractions

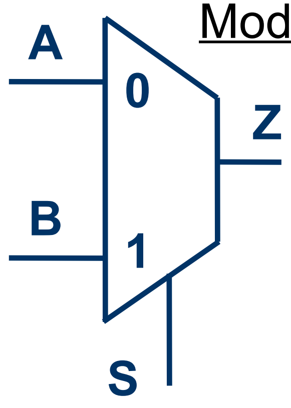


Examples of design abstractions (1)



MSI Building Block

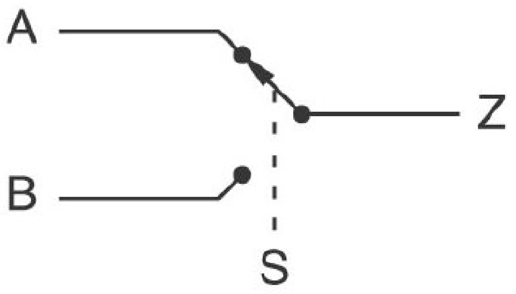
Module Level



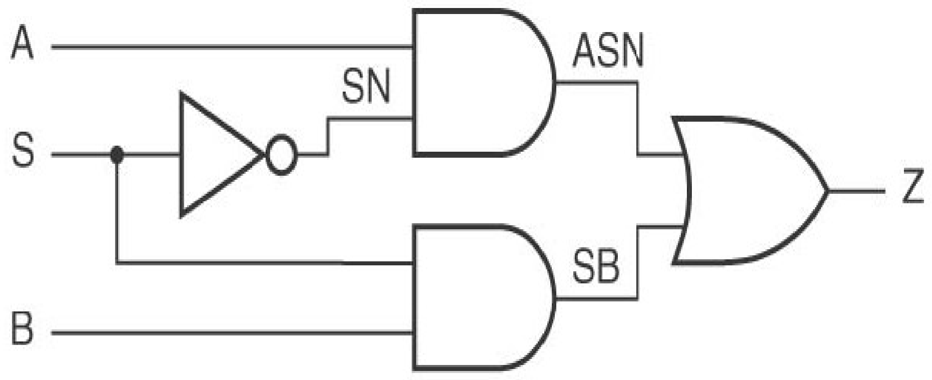
Truth Table

S	A	B	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Switch Level



Gate Level

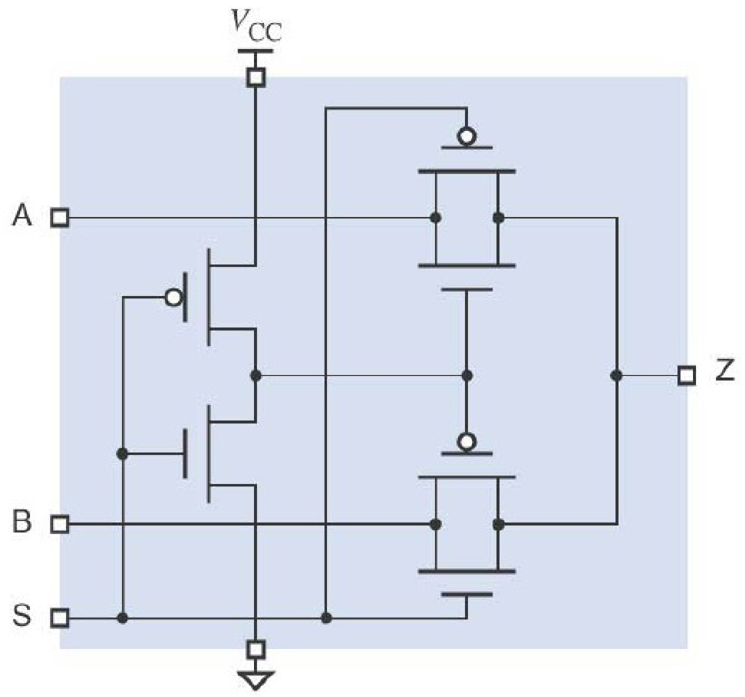


Logic Level

$$Z = A\,S' + B\,S$$

Examples of design abstractions (2)

MOS transistors Level



Register Transfer Level VHDL

```
library IEEE;
use IEEE.std_logic_1164.all;

entity Vchap1mux is
    port ( A, B, S: in  STD_LOGIC;
           Z:       out STD_LOGIC );
end Vchap1mux;

architecture Vchap1mux_arch of Vchap1mux is
begin
  Z <= A when S = '0' else B;
end Vchap1mux_arch;
```

Structural Level VHDL

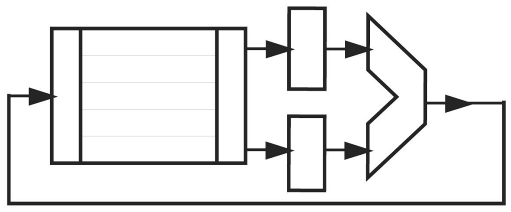
```
architecture Vchap1mux_gate_arch of Vchap1mux is
signal SN, ASN, SB: STD_LOGIC;
  -- required component declarations have been omitted
  --   for brevity in this example.
begin
  U1: port map INV (S, SN);
  U2: port map AND2 (A, SN, ASN);
  U3: port map AND2 (S, B, SB);
  U4: port map OR2 (ASN, SB, Z);
end Vchap1mux_gate_arch;
```

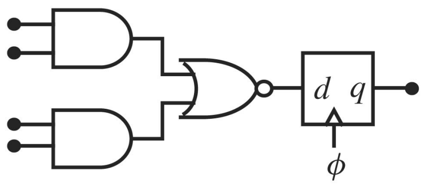
Structured Design

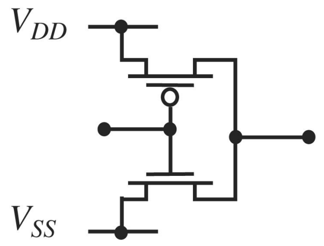
- **Hierarchy**
  - Divide and Conquer paradigm
- **Modularity**
  - Well-defined interfaces allow modules to be treated as black boxes
- **Regularity**
  - It makes easier to reuse blocks

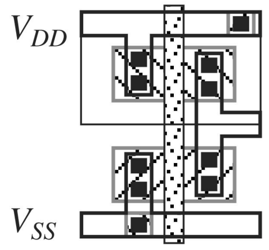
- Standard cell libraries are a very good example of modularity and regularity

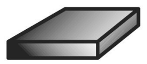
IC Design Hierarchy

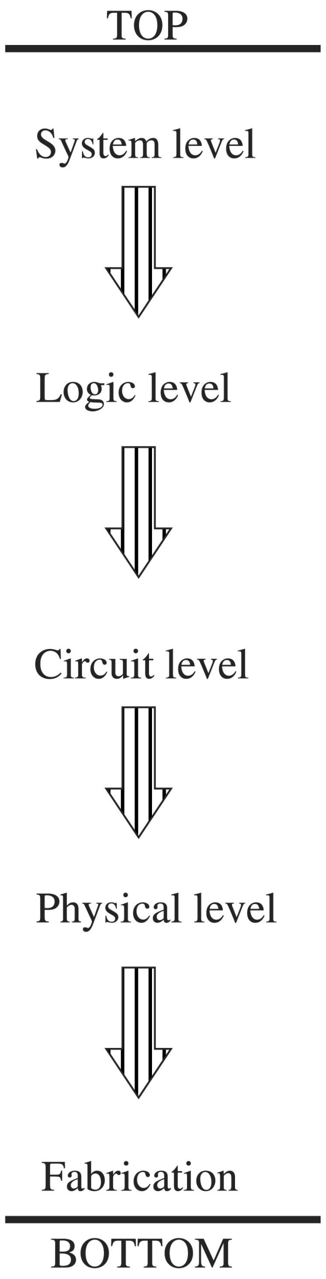


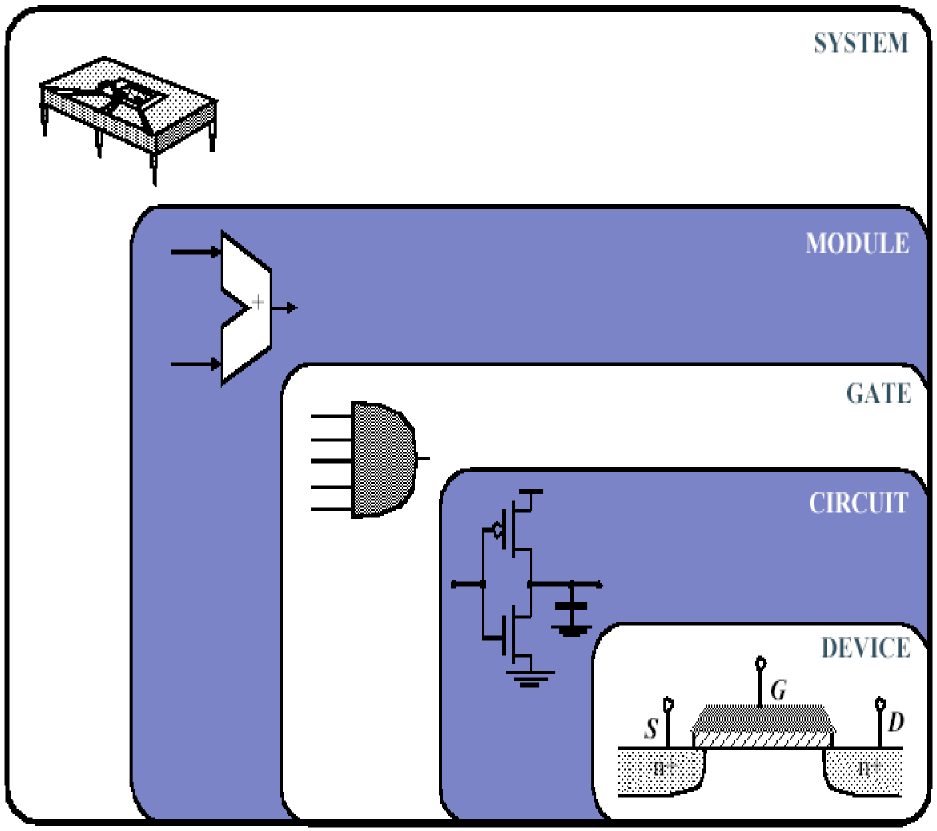












Standard cells

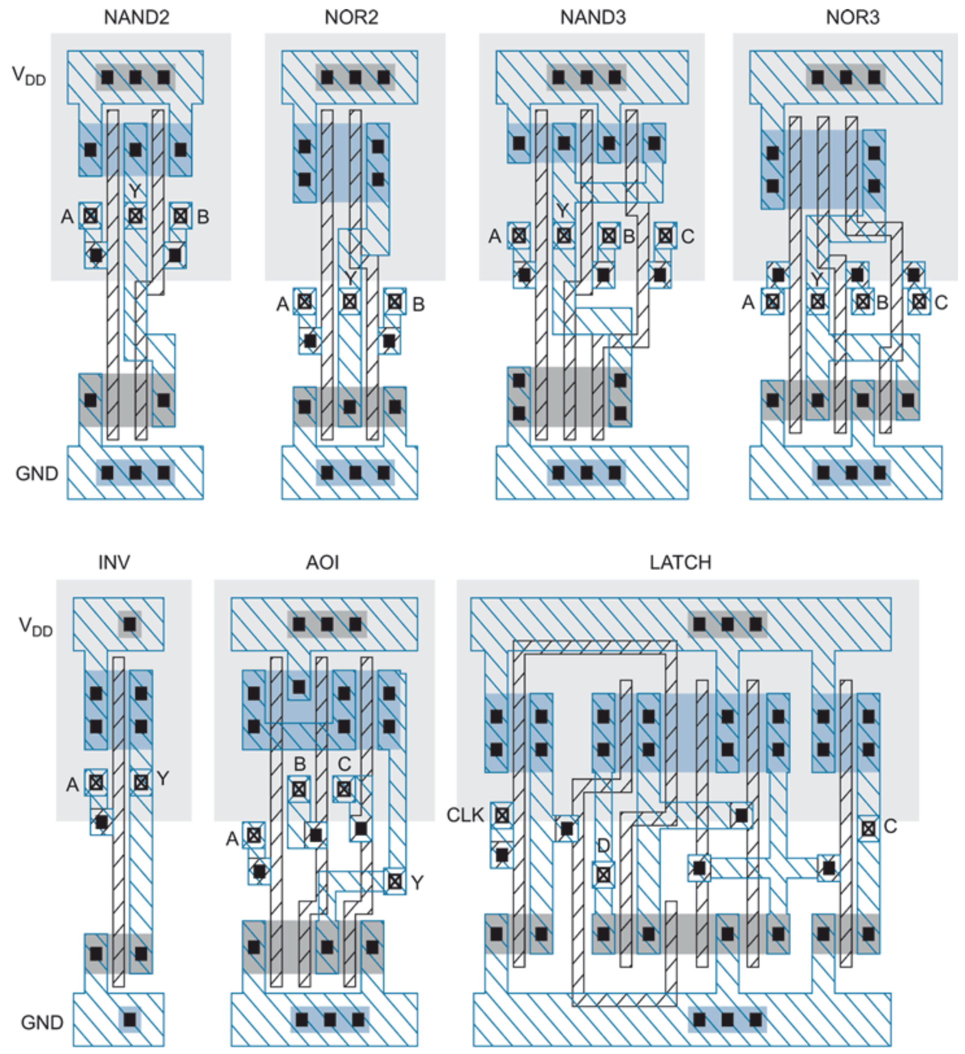
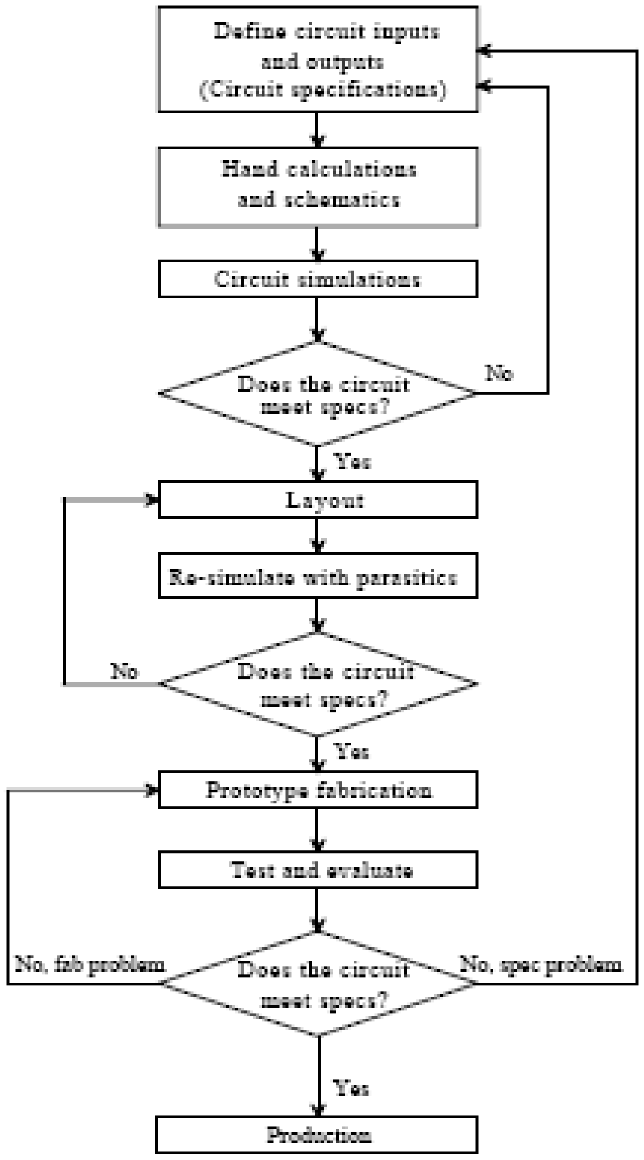


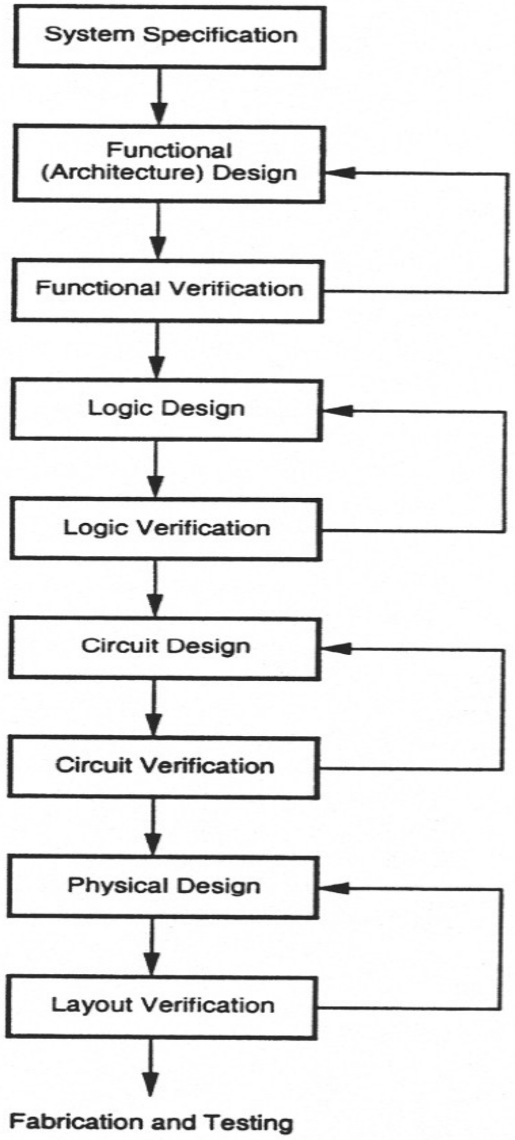
FIG 1.62 Simple standard cell library. Color version on inside front cover.

Simplified IC Design Flows

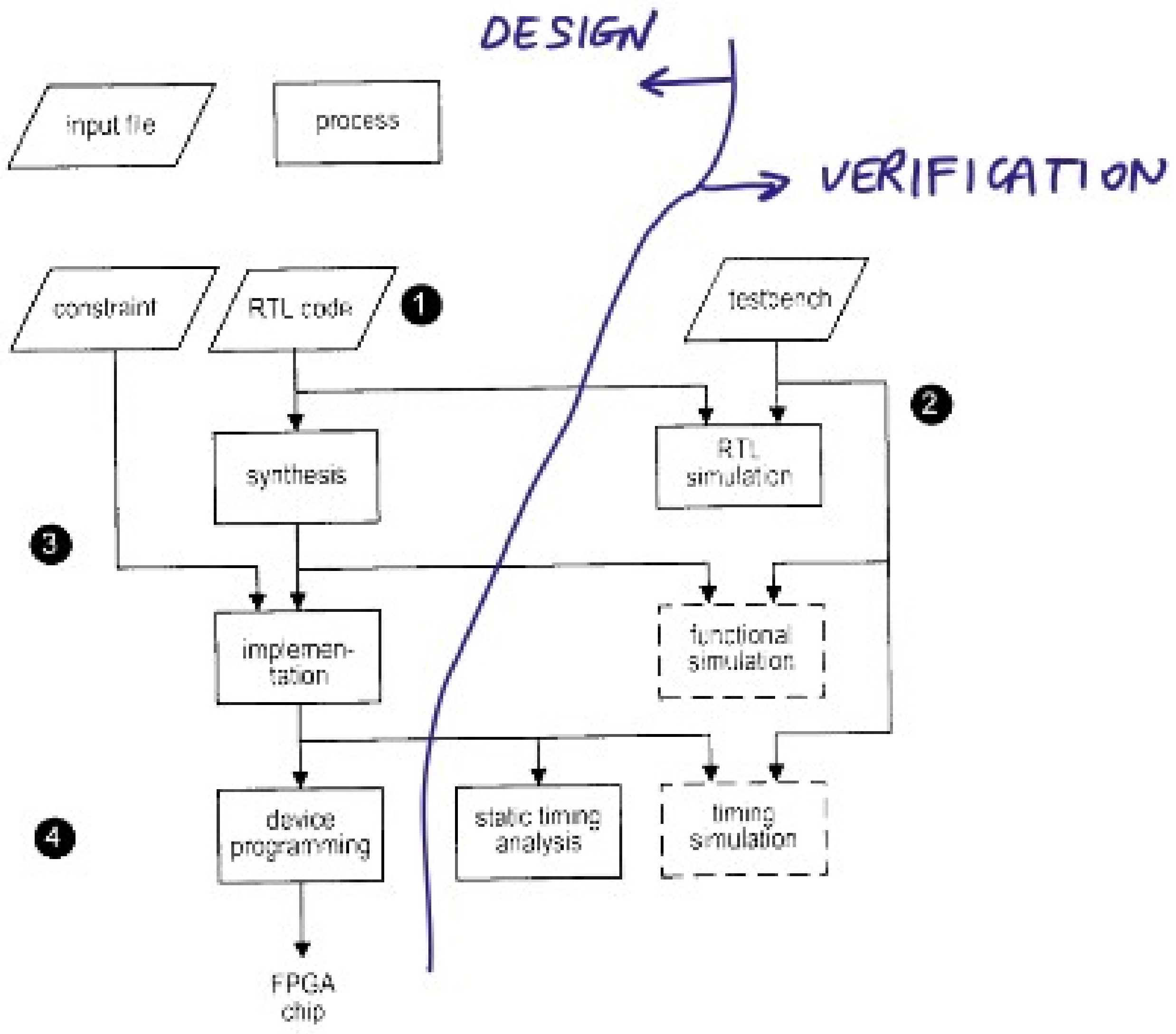


Define circuit inputs and outputs (Circuit specifications)
Hand calculations and schematics
Circuit simulations
Does the circuit meet specs?
No
Yes
Layout
Re-simulate with parasitics
Does the circuit meet specs?
No
Yes
Prototype fabrication
Test and evaluate
Does the circuit meet specs?
No, fab problem
No, spec problem
Yes
Production

Figure 1.1 Flowchart for the CMOS IC design process.



Another Simplified IC Design Flow



Implementation:
- Translation (merge all design files into a single netlist)
- Device mapping
- P&R

Device Programming:
- Generation conf. file
- Download conf. file into device

RTL code (Verilog)

```
assign cout = (a&b) | (a&c) | (b&c);
```

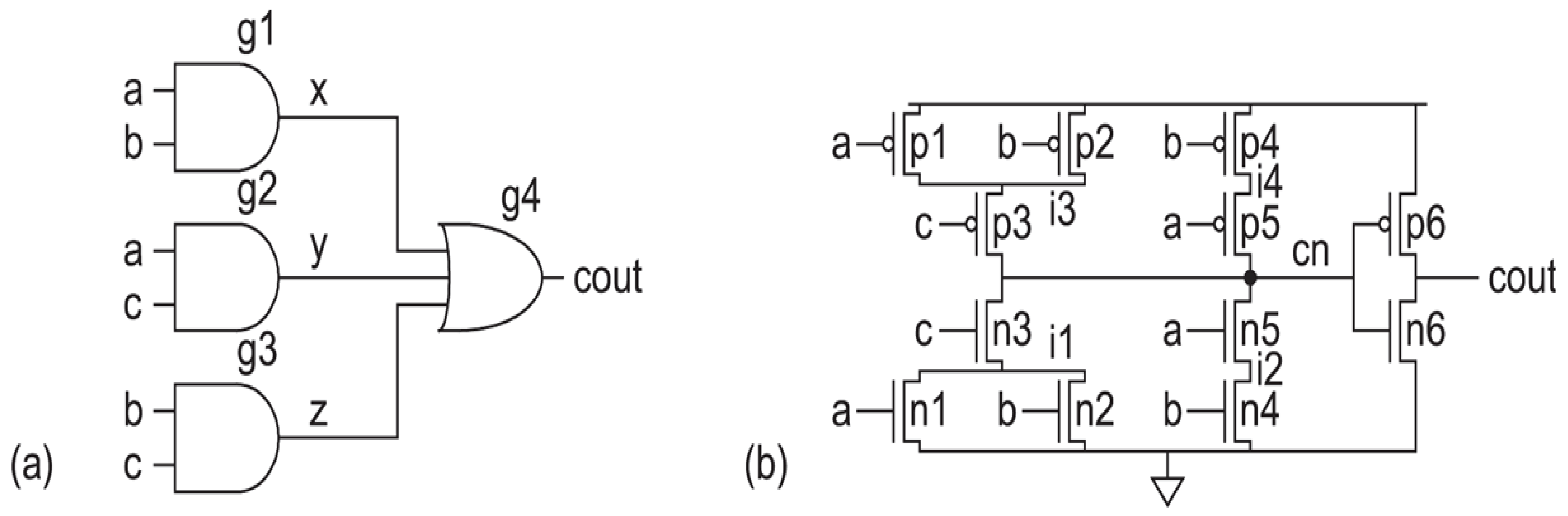


FIG 1.59 Carry subcircuit

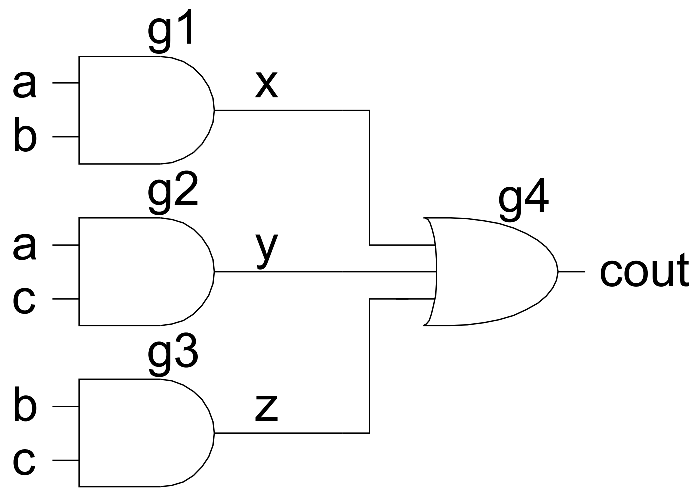
Gate level netlist (Verilog)

```
module carry(input  a, b, c,
             output cout)

   wire   x, y, z;

   and g1(x, a, b);
   and g2(y, a, c);
   and g3(z, b, c);
   or  g4(cout, x, y, z);

endmodule
```



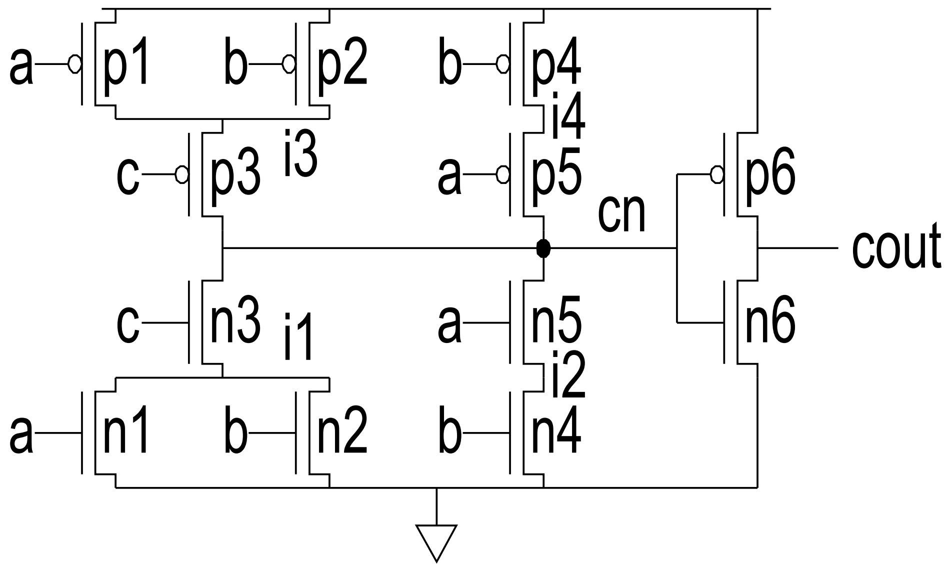
Transistor level netlist (Verilog)

```
module carry(input  a, b, c, output cout)

    wire     i1, i2, i3, i4, cn;

    tranif1 n1(i1, 0, a);
    tranif1 n2(i1, 0, b);
    tranif1 n3(cn, i1, c);
    tranif1 n4(i2, 0, b);
    tranif1 n5(cn, i2, a);
    tranif0 p1(i3, 1, a);
    tranif0 p2(i3, 1, b);
    tranif0 p3(cn, i3, c);
    tranif0 p4(i4, 1, b);
    tranif0 p5(cn, i4, a);
    tranif1 n6(cout, 0, cn);
    tranif0 p6(cout, 1, cn);

endmodule
```



SPICE netlist

```
.SUBCKT CARRY A B C COUT VDD GND
MN1 I1 A GND GND NMOS W=1U L=0.18U AD=0.3P AS=0.5P
MN2 I1 B GND GND NMOS W=1U L=0.18U AD=0.3P AS=0.5P
MN3 CN C I1 GND NMOS W=1U L=0.18U AD=0.5P AS=0.5P
MN4 I2 B GND GND NMOS W=1U L=0.18U AD=0.15P AS=0.5P
MN5 CN A I2 GND NMOS W=1U L=0.18U AD=0.5P AS=0.15P
MP1 I3 A VDD VDD PMOS W=2U L=0.18U AD=0.6P AS=1 P
MP2 I3 B VDD VDD PMOS W=2U L=0.18U AD=0.6P AS=1P
MP3 CN C I3 VDD PMOS W=2U L=0.18U AD=1P AS=1P
MP4 I4 B VDD VDD PMOS W=2U L=0.18U AD=0.3P AS=1P
MP5 CN A I4 VDD PMOS W=2U L=0.18U AD=1P AS=0.3P
MN6 COUT CN GND GND NMOS W=2U L=0.18U AD=1P AS=1P
MP6 COUT CN VDD VDD PMOS W=4U L=0.18U AD=2P AS=2P
CI1 I1 GND 2FF
CI3 I3 GND 3FF
CA A GND 4FF
CB B GND 4FF
CC C GND 2FF
CCN CN GND 4FF
CCOUT COUT GND 2FF
.ENDS
```

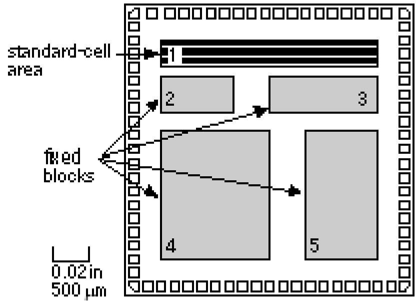
Types of ICs

- ASSP
- ASIC

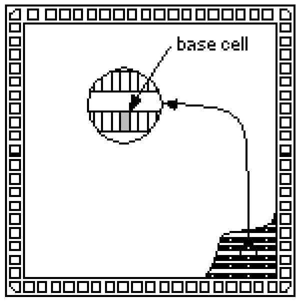
Use

- Full-custom
- Semi-custom
  - Cell Based
  - Gate Arrays
- Programmable
  - CPLD and FPGA

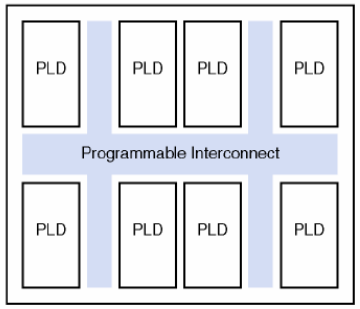
Design Style



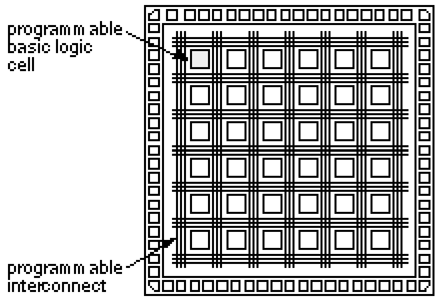
Cell based



Gate Array

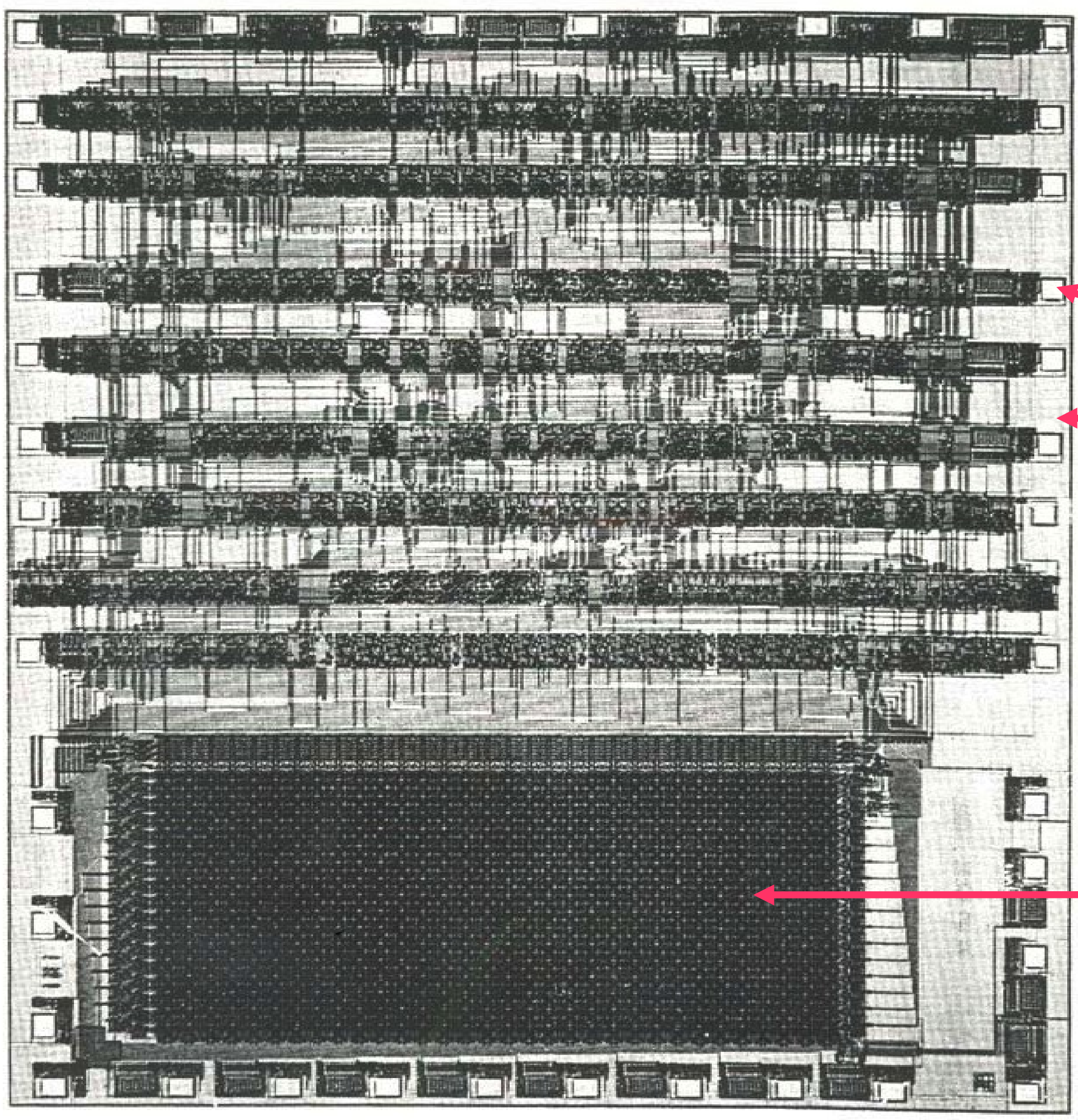


CPLD



FPGA

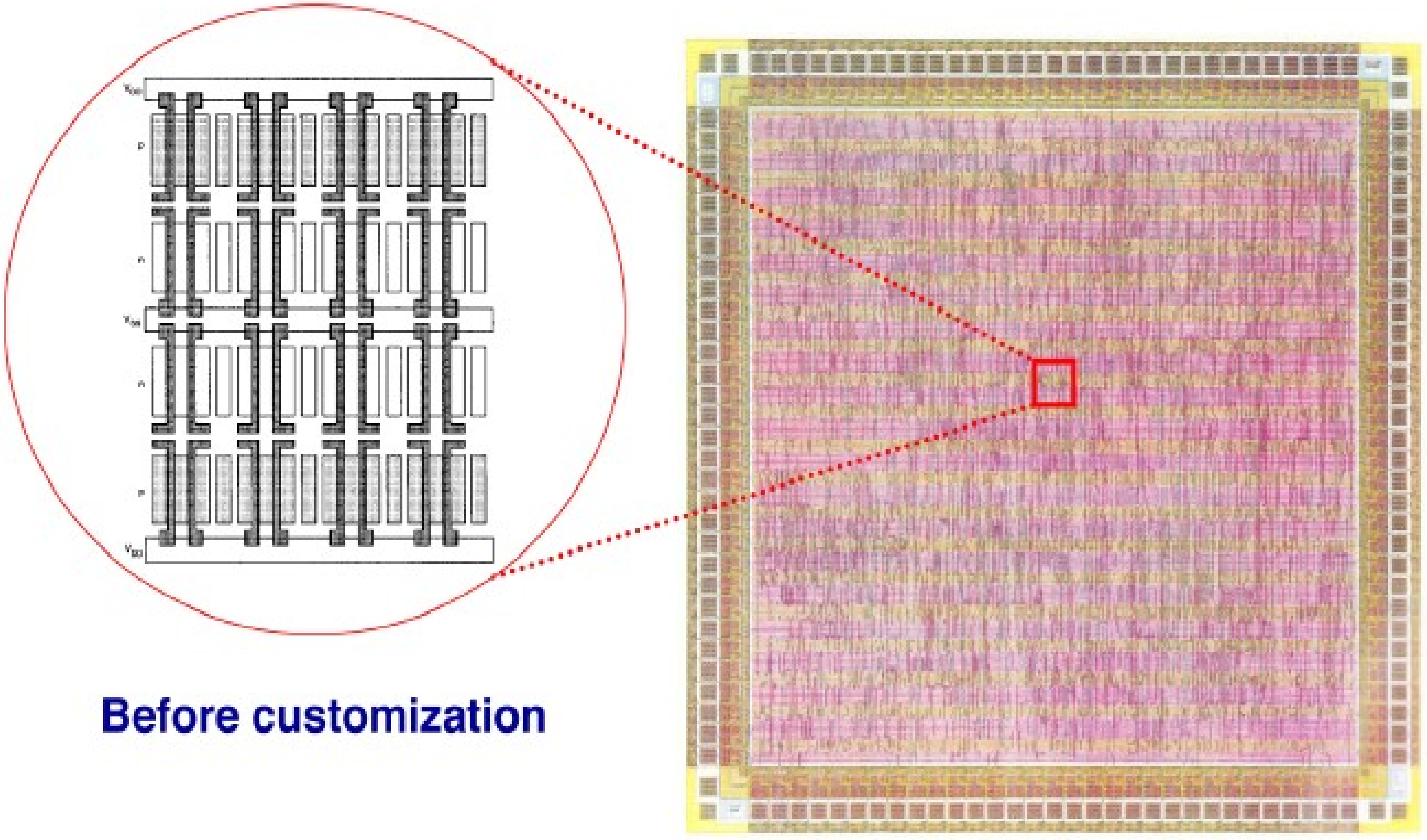
Standard Cells



Rows of standard cells with routing channels between them

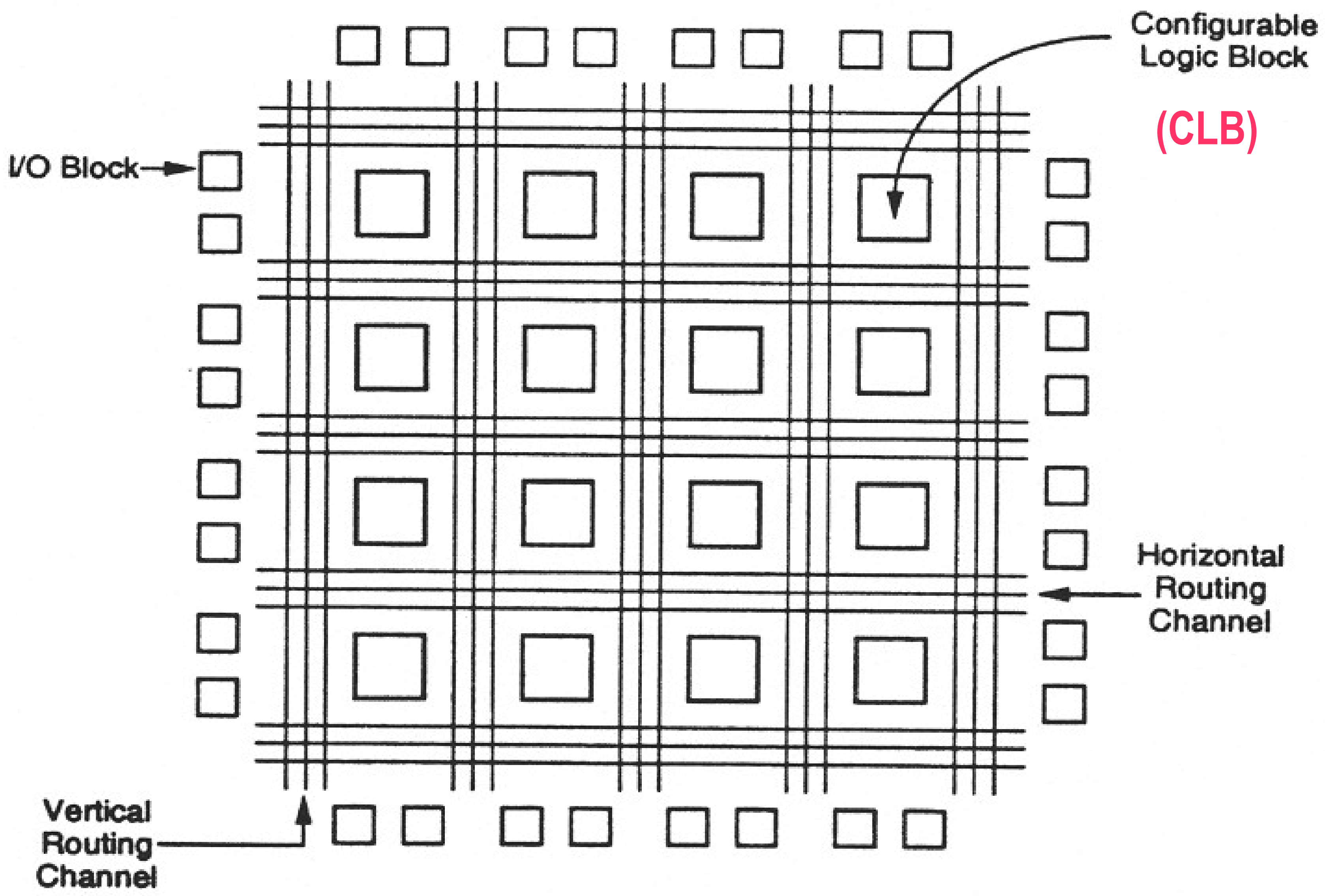
Memory array

Gate Arrays

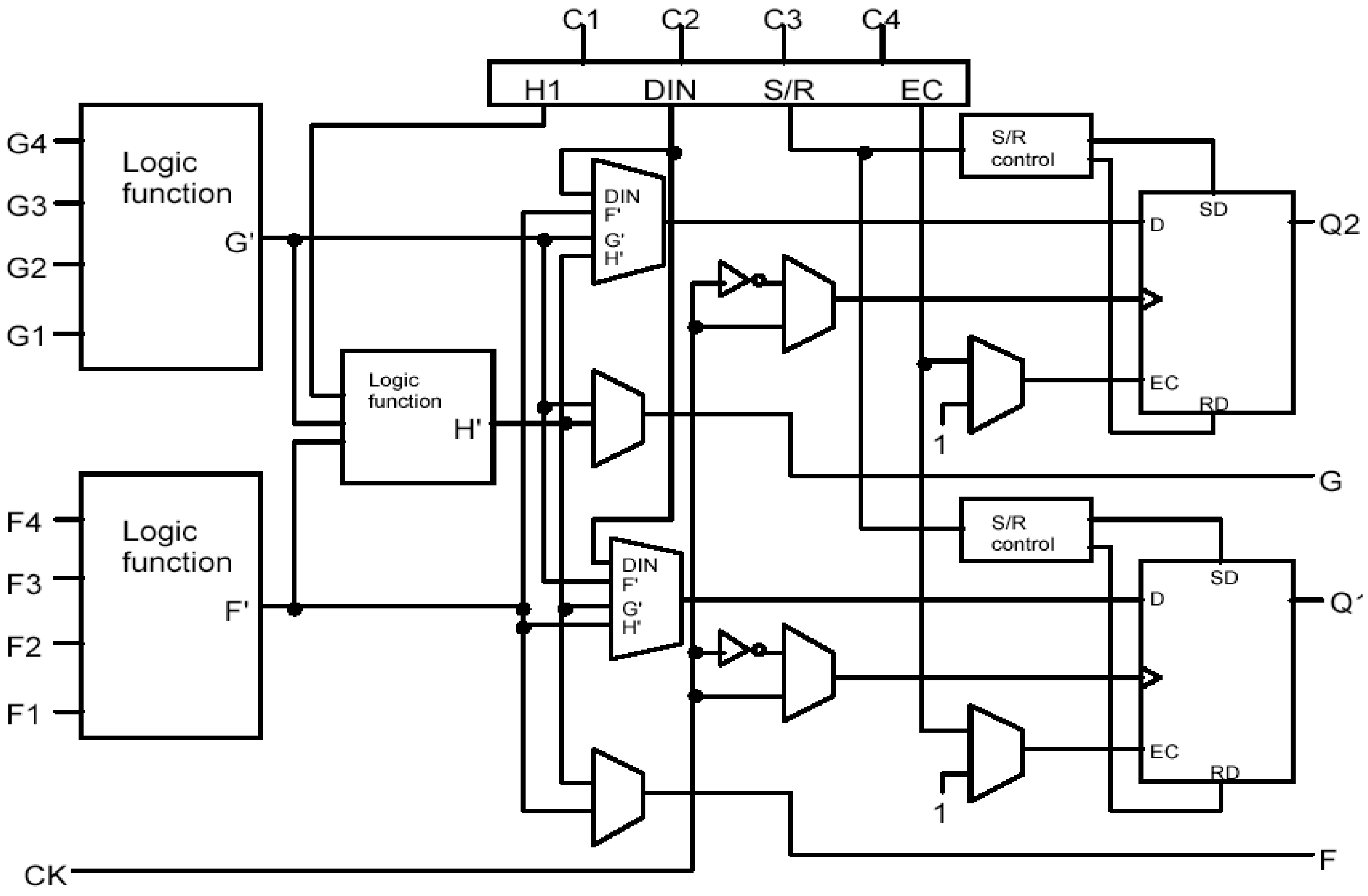


Before customization

Field Programmable Gate Array



Internal Structure of a CLB



Various on chip structures

- Random logic
- Data paths
- Arrays
- Analog
- Input/output (I/O)

Example of random Logic

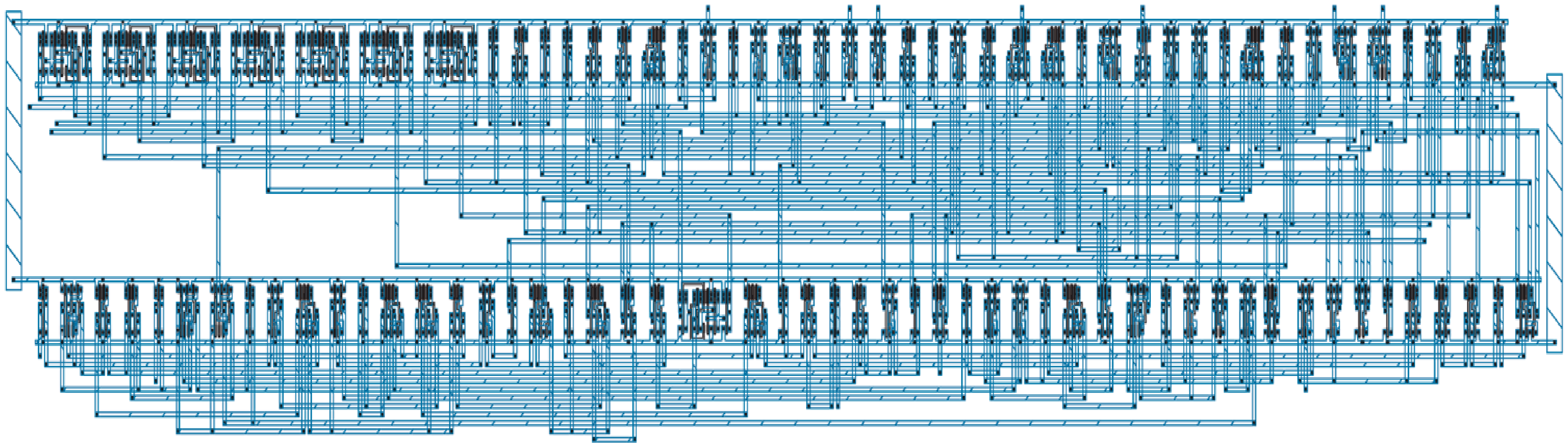


FIG 1.63 MIPS controller layout

- Synthesized MIPS controller

Another example of random logic

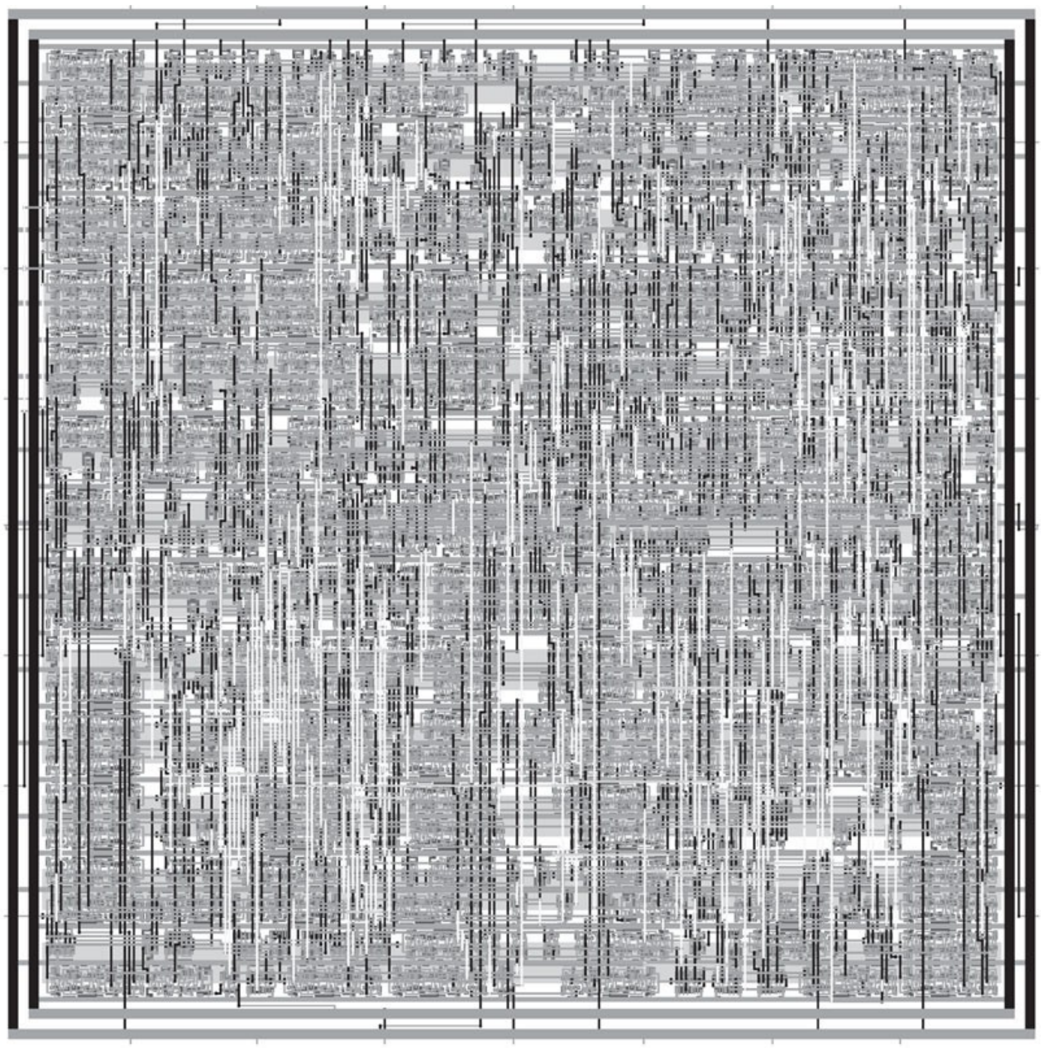


FIG 1.64 Synthesized MIPS processor

- Synthesized MIPS

Example of data path

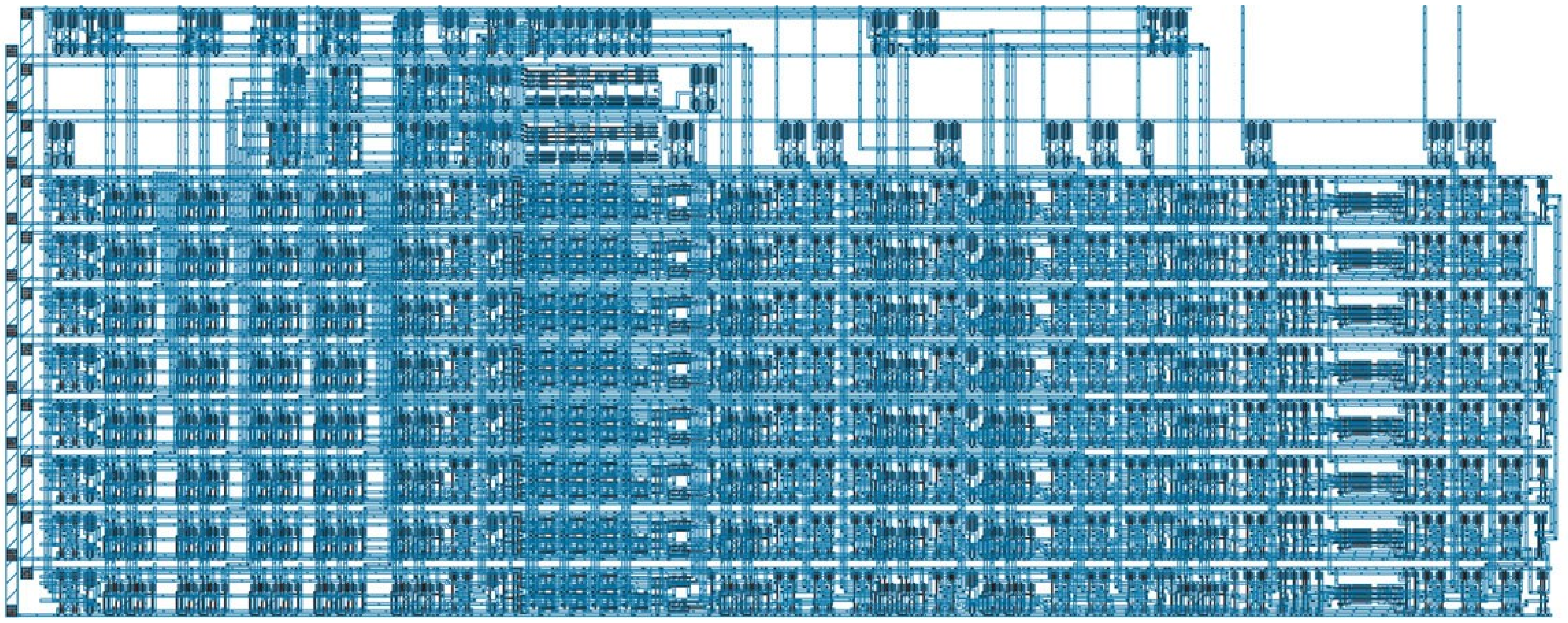
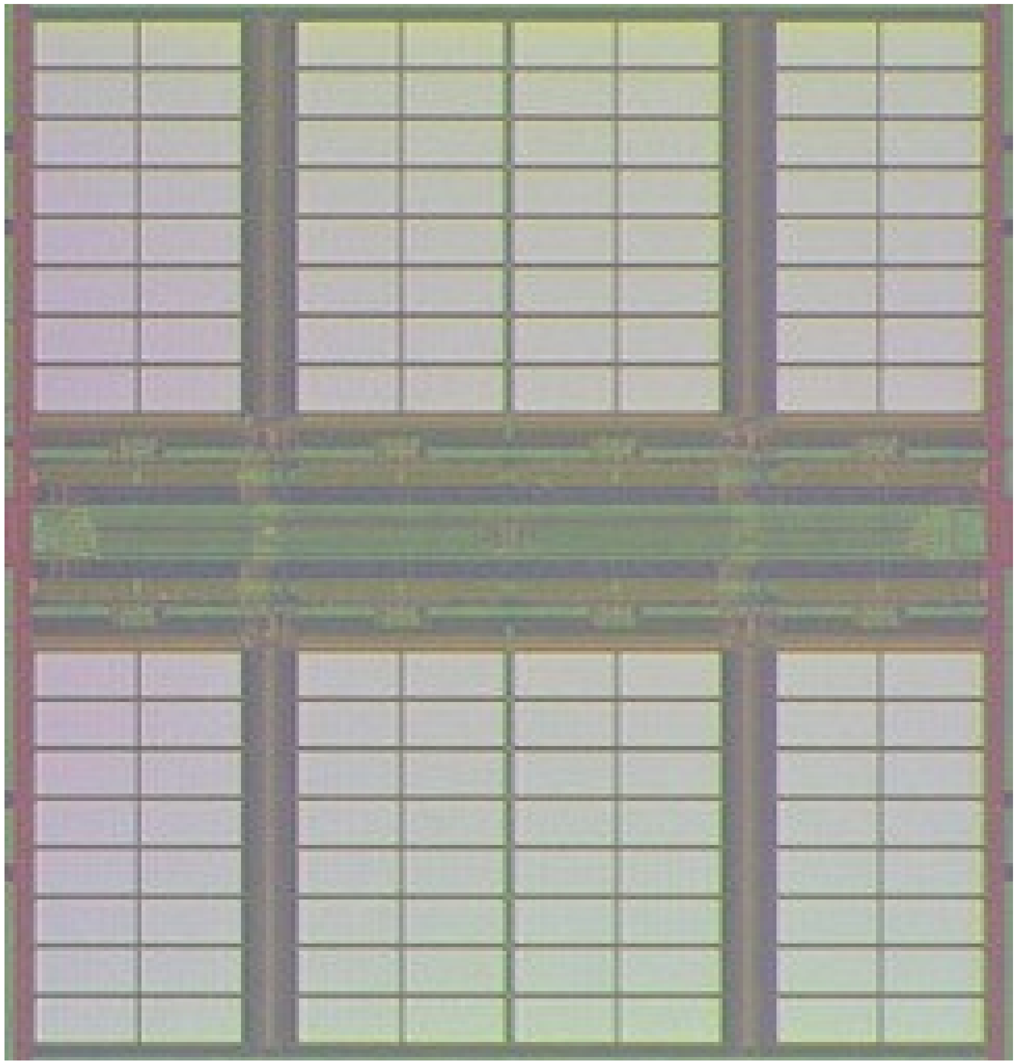


FIG 1.66 MIPS datapath layout

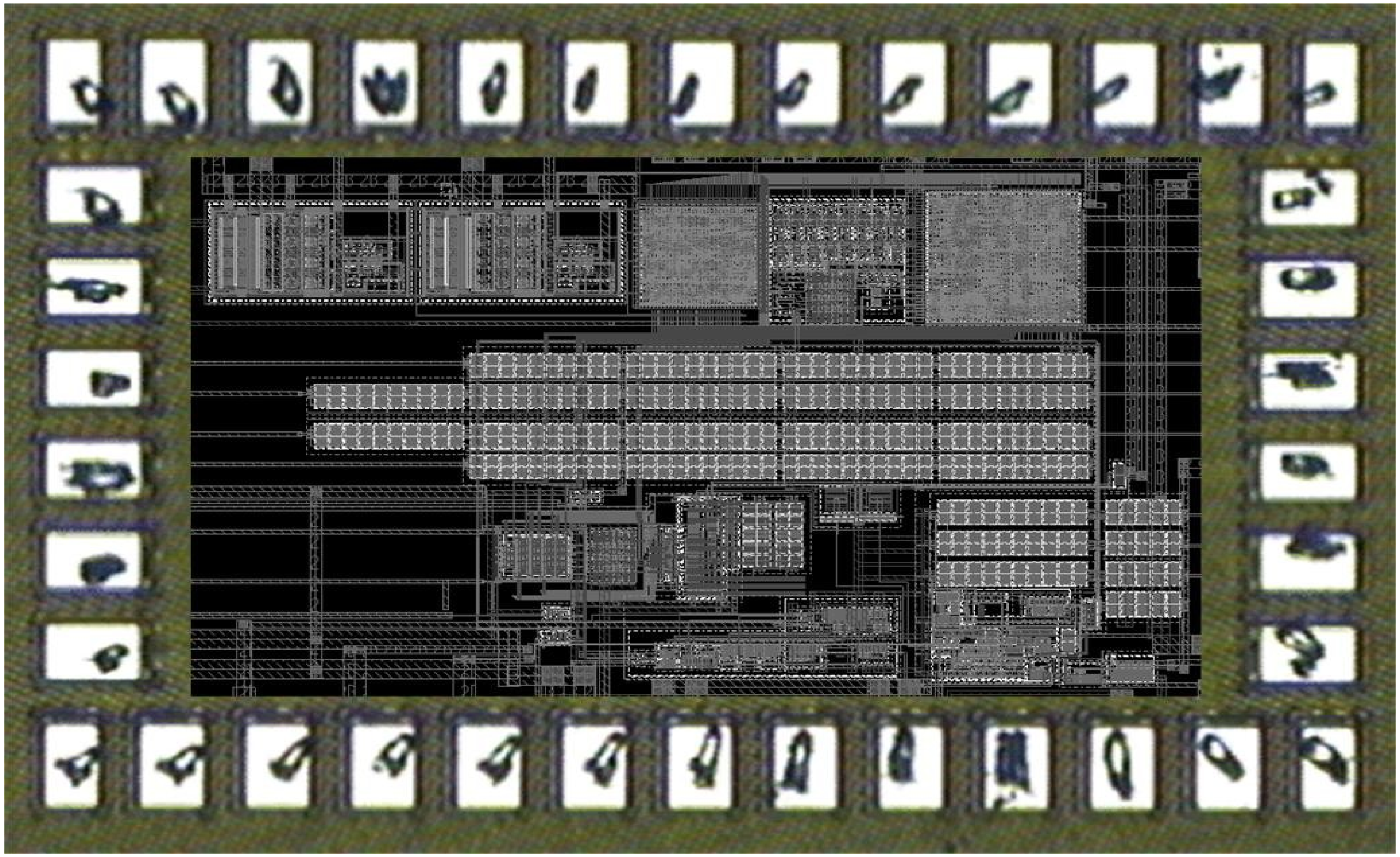
- Hand-Crafted MIPS datapath

Example of Array



- SRAM chip

Example of Analog structure



- Charge Pump Phase-Locked Loop

Example of I/O

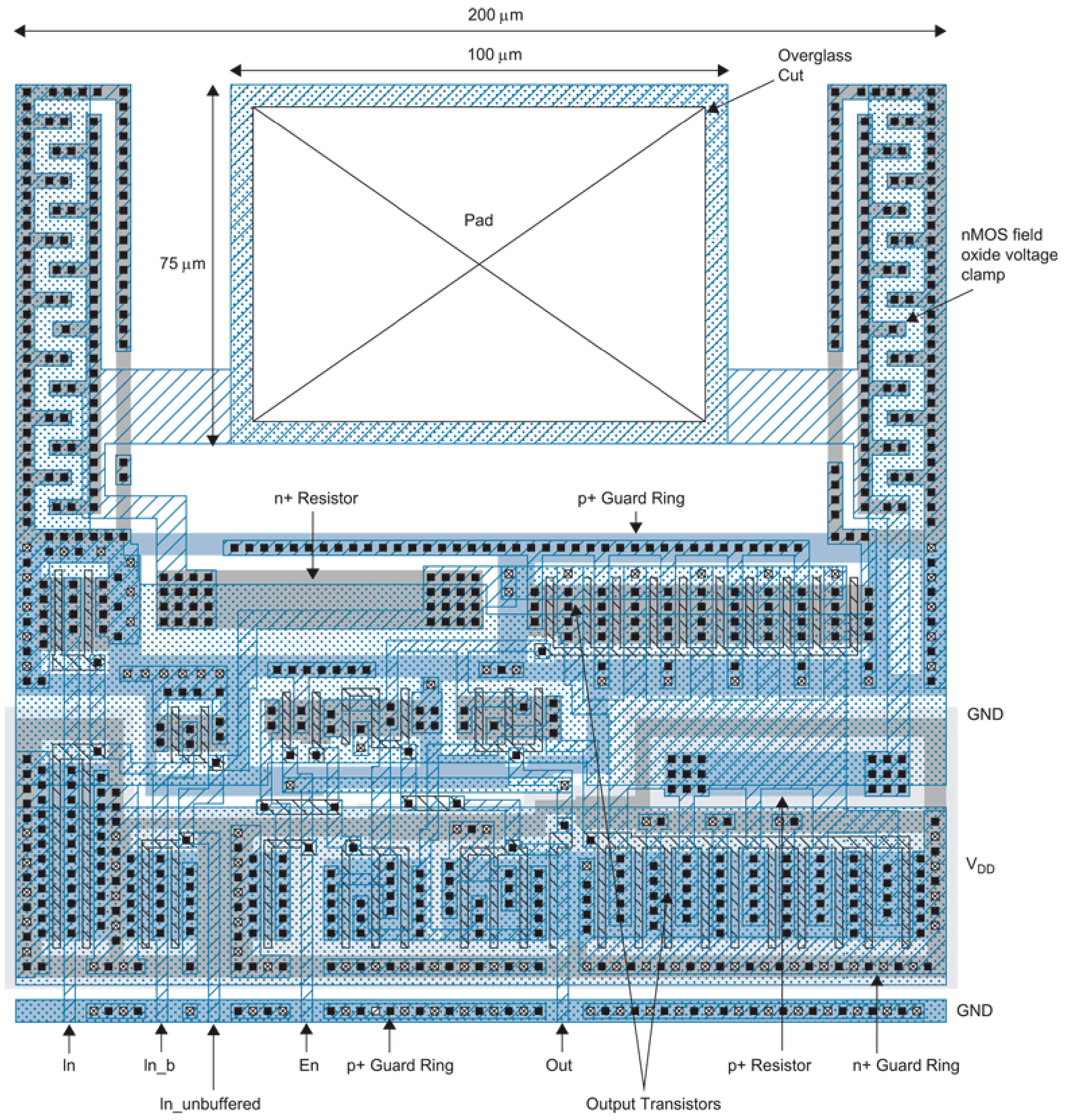


FIG 12.23 MOSIS 1.6 μm bidirectional pad. Color version on inside front cover.

Bidirectional I/O PAD circuit

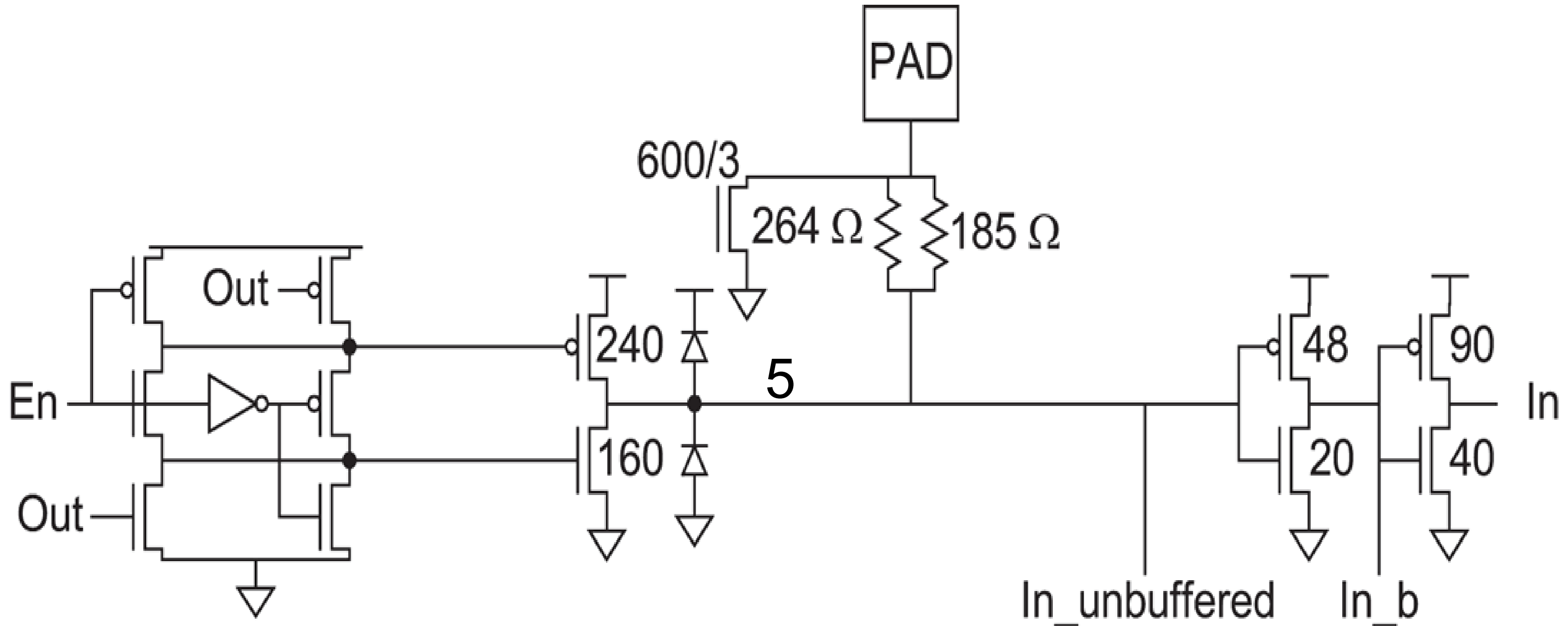


FIG 12.24 MOSIS bidirectional pad schematic

Physical Design

- Floorplanning and area estimation
- Standard Cell Based Layout
  - Place and Route
  - Parasitic Extraction
  - Post Layout Verification
- Data-path Based Layout
  - Slice Planning
  - Parasitic Extraction
  - Post Layout Verification

Floorplanning

- Does the design fit the chip area budgeted ?
- Estimates area of major units and defines their relative placement
- Estimate wire lengths
- Estimate wiring congestion

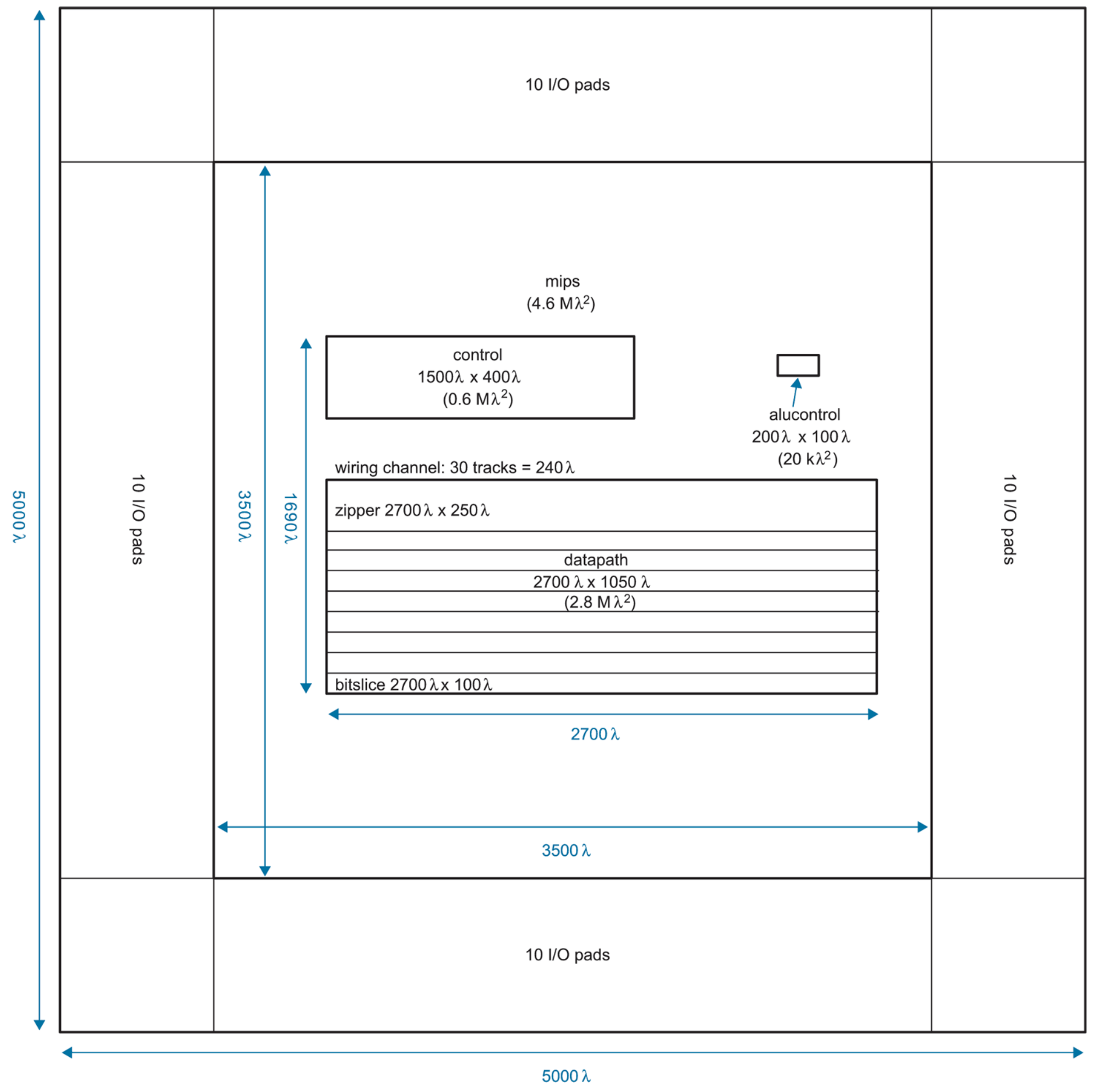


FIG 1.60 MIPS floorplan

Area Estimation

- Some cell library vendor specify cell layout densities in Kgates/mm^2

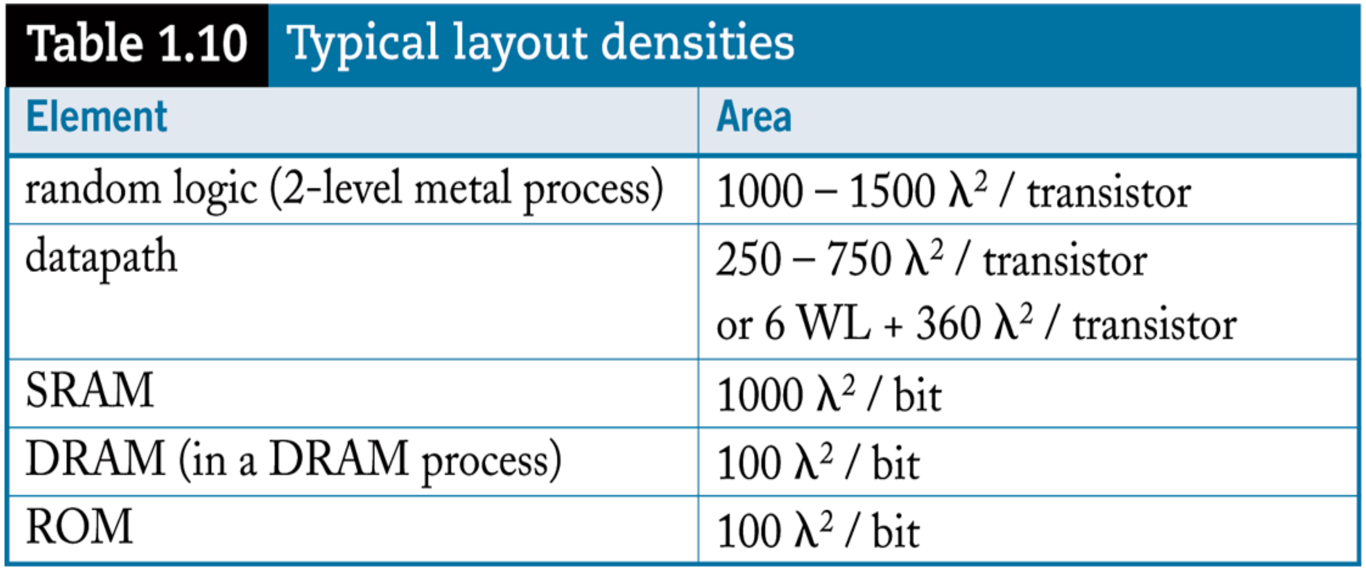


Table 1.10 Typical layout densities

Element	Area
random logic (2-level metal process)	1000 – 1500 λ^2 / transistor
datapath	250 – 750 λ^2 / transistor or 6 WL + 360 λ^2 / transistor
SRAM	1000 λ^2 / bit
DRAM (in a DRAM process)	100 λ^2 / bit
ROM	100 λ^2 / bit

- Compare to another block you already designed or estimate from transistor counts
- Budget room for large wiring tracks

Example of Layout

rail
rings

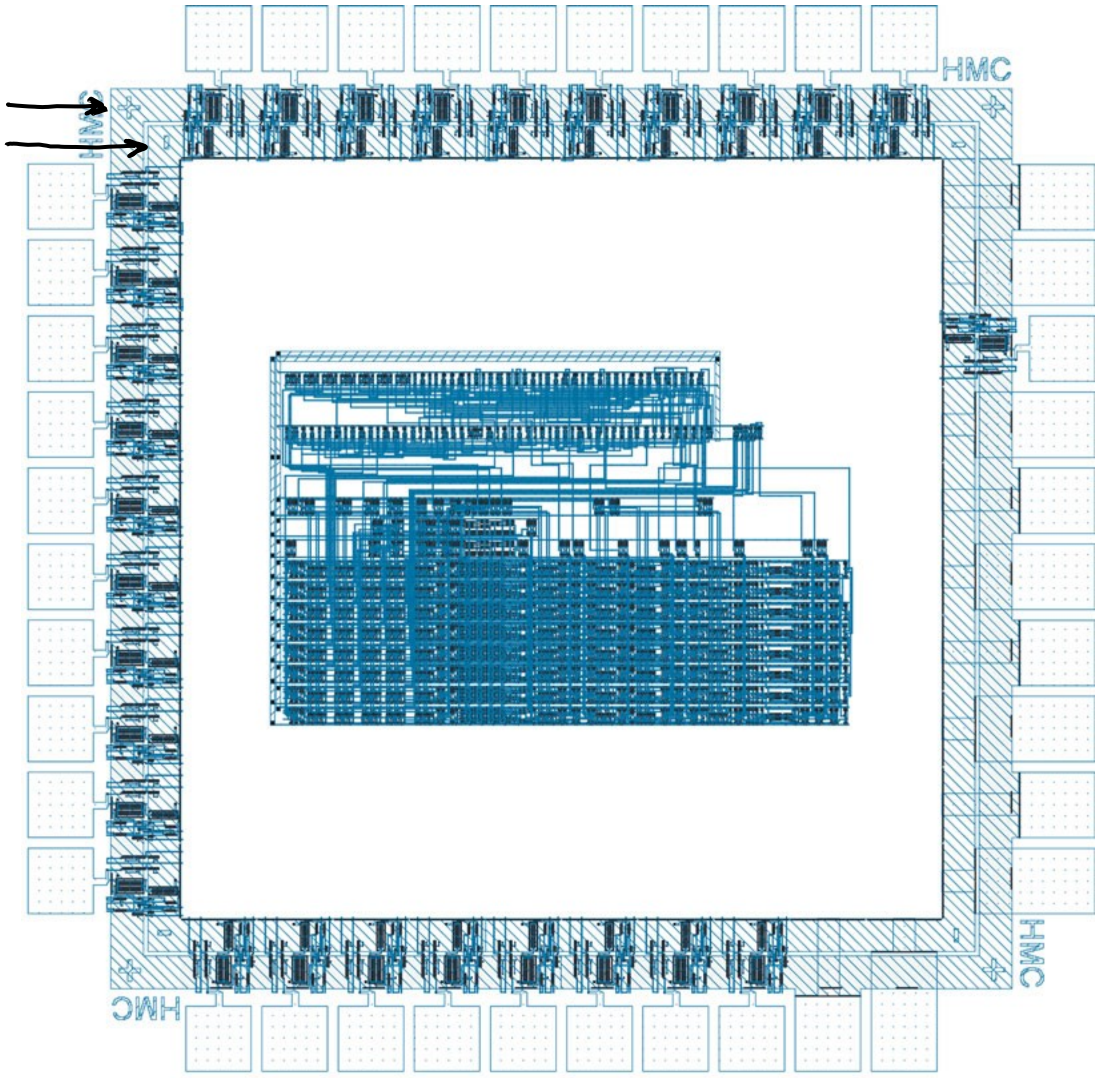


FIG 1.61 MIPS layout

- This design is pad limited (the I/O pads set the area of the chip)
- core-limited (the logic set the chip area)

CAD Tools

- Designers rely increasingly on design automation software tools to seek productivity gains and to cope with increased complexity

Typical Design Flow

Logic Design:

- Design Entry
  - Schematic capture
  - Hardware Description Languages
- Logic Synthesis ← if design entry is in HDL
- Pre layout verification
  - Functional simulation
  - Formal methods
  - Timing Analysis

Physical Design:

- Floorplanning
- Placement
- Routing
- Extraction
- Post layout verification

Verification

- Fabrication is slow & expensive
- MOSIS 0.6μm masks: $1000, 3 months
- State of art masks (130nm): $1M, 1 month
- Debugging chips is very hard
- Limited visibility into operation
- Prove design is right before building!
- System simulation & performance Assessment (C/C++
- Logic Simulation / formal verification / STA
- Circuit simulation
- Layout vs. schematic comparison (LVS)
- Design & electrical rule checks (DRC, ERC)
- Verification is > 50% of effort on most chips !

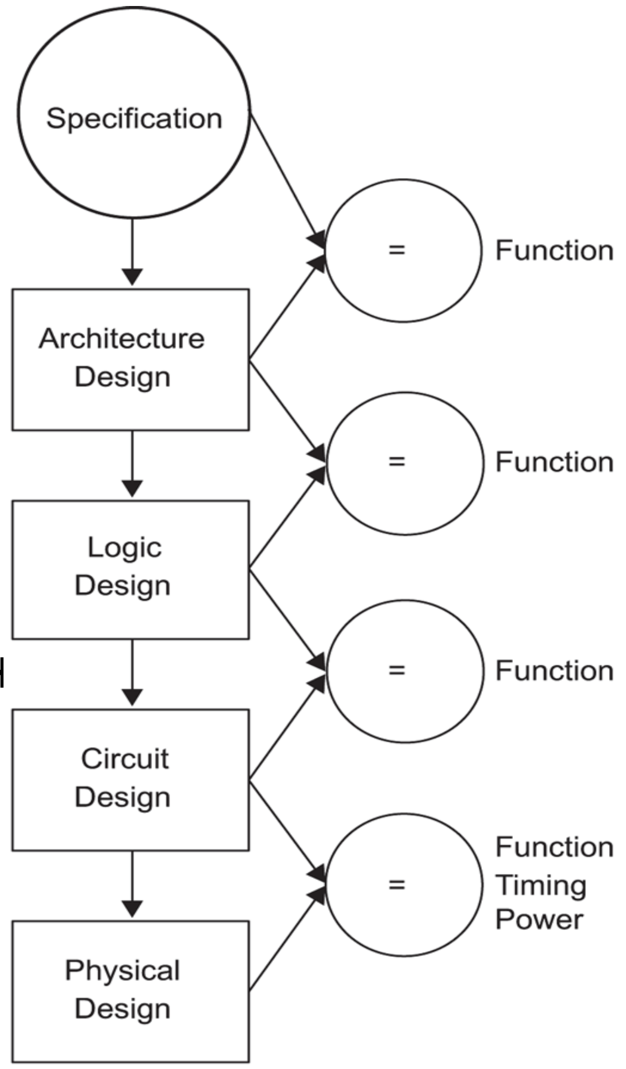


FIG 1.69 Design and verification sequence

Fabrication

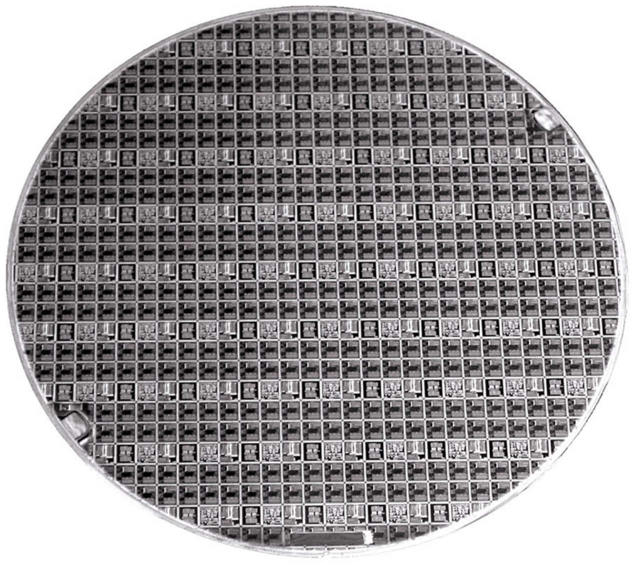


FIG 1.70 Processed 8-inch wafer

- Tapeout final layout
  - Formats for mask descriptions:
    CIF (academia) and GDS II (industry)
- Fabrication
  - 6, 8, 12” wafers (bare wafer costs $1000-$5000)
  - Optimized for throughput, not latency
    (turnaround times up to 10 weeks !)
  - Cut into individual dice
- Fabs cost billions of dollars and become obsolete in a few years
  - Fabless semiconductor companies
  - Manufacturing Companies: TSMS, UMC, IBM

Testing

- Test that chip operates as expected
  - Design errors
  - Manufacturing errors
- A single dust particle or wafer defect kills a die
  - Yields from 90% to < 10%
  - Depends on die size, maturity of process
  - Test each part before shipping to customer

Summary

- Chip Design requires a fundamental understanding of circuit and physical design
- This is true even if many chip designers spend much of their time specifying circuits with HDL and seldom look at the actual transistors

- The best way to learn VLSI design is by doing it !

MOS Transistors (POS to be picky)

- TRANS-ISTOR (=TRANSFER-RESISTOR)
- Four terminals: gate, source, drain, body (= bulk)

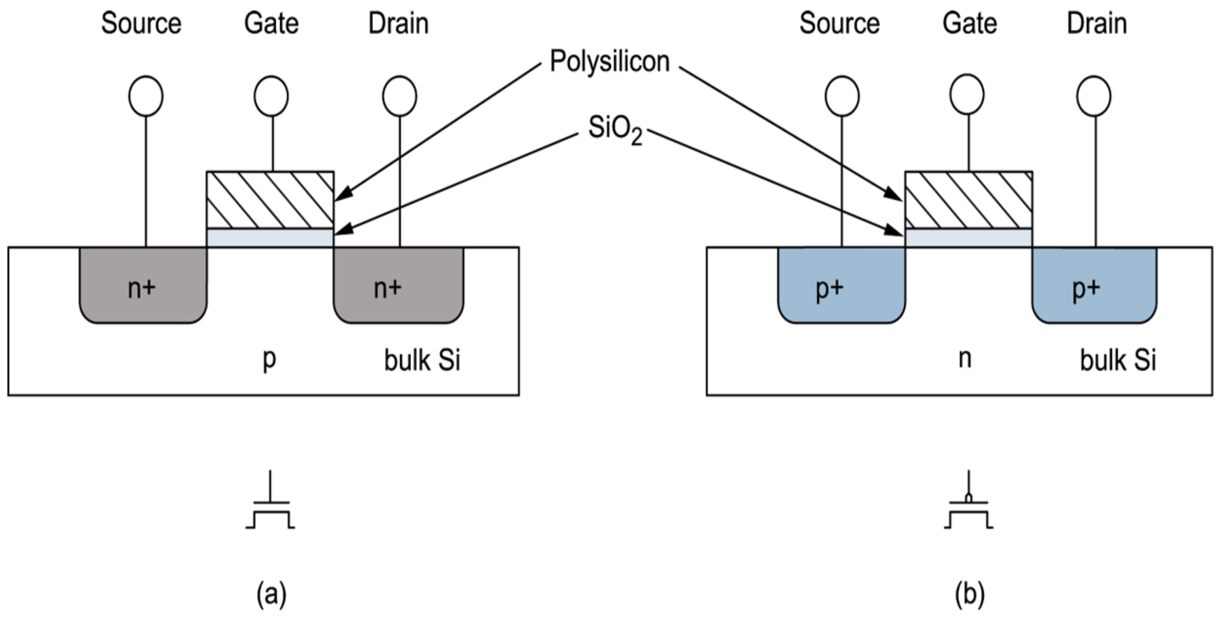


FIG 1.8 nMOS transistor (a) and pMOS transistor (b)

Our first CMOS circuit

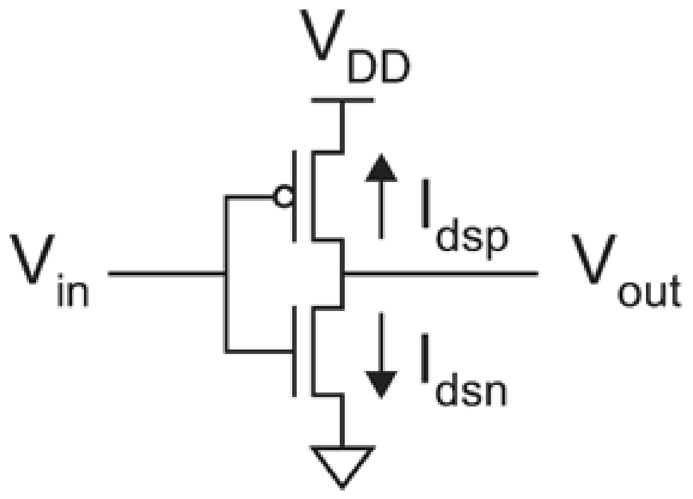


FIG 2.23 A CMOS inverter