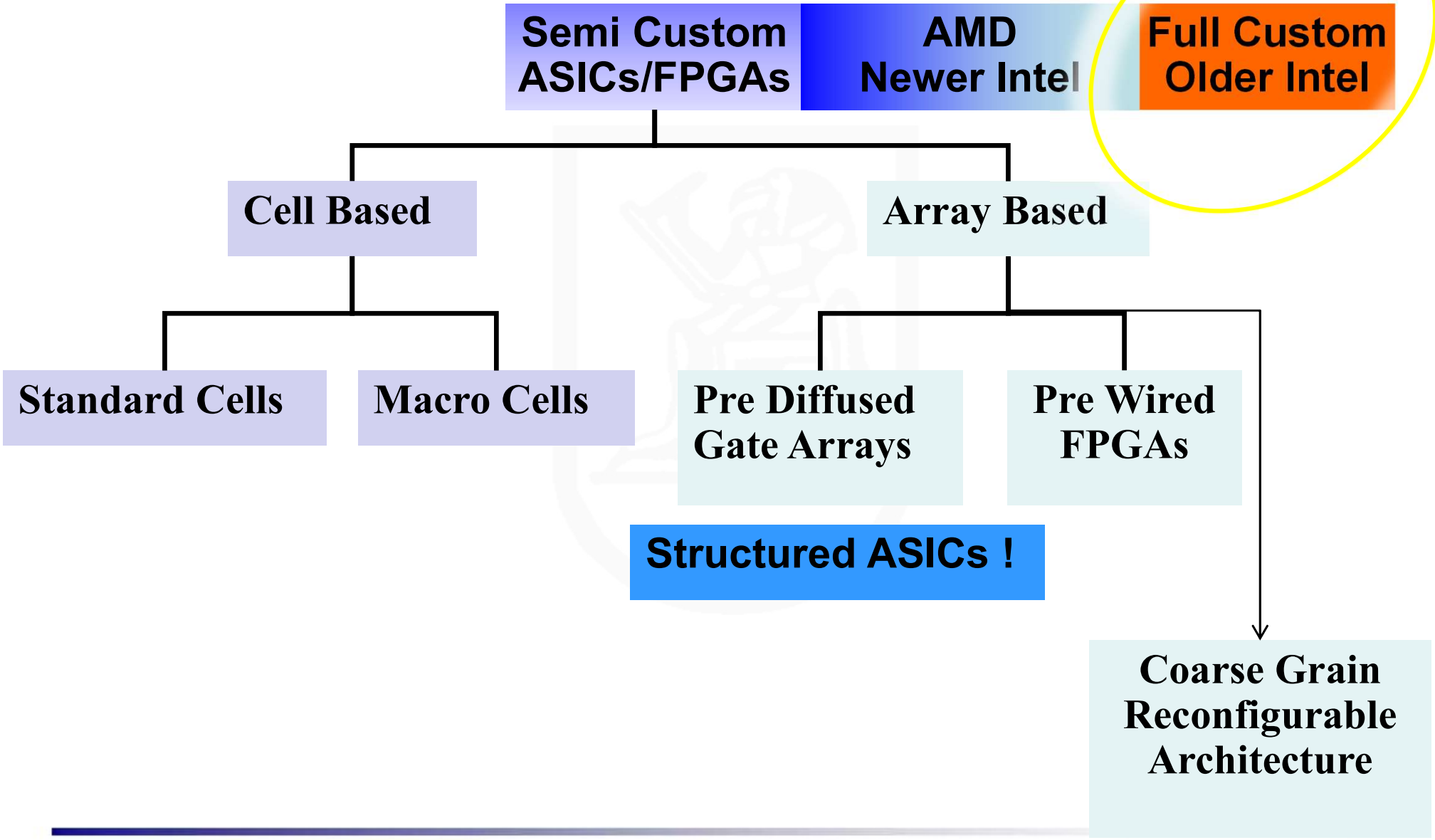
IC Design Flows

- The MOS Transistor
- Analog and Circuit Design
- Digital Logic Families
- Productivity Gap
- **Digital Design Flows**

Physical Implementation Styles



Full Custom Design

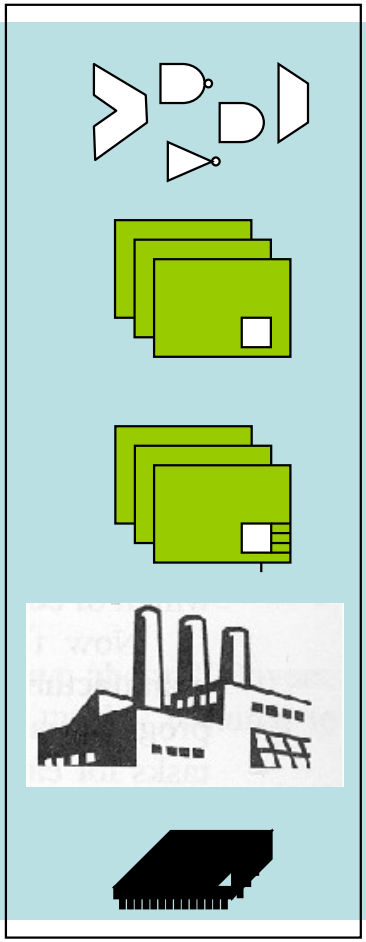
Founders of Intel

Andy Grove
Robert Noyce
Gordon Moore

with the Rubylith of 8080 microprocessor

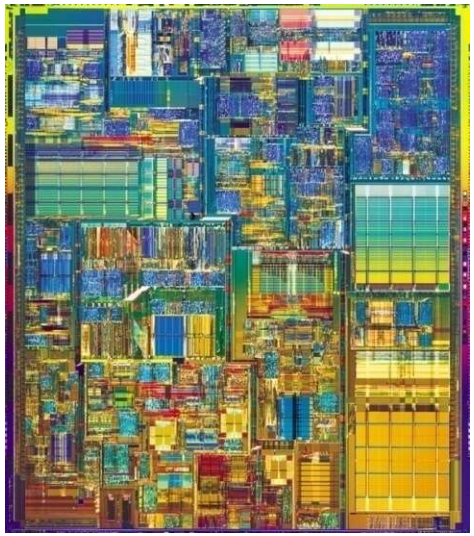
IC Design Styles

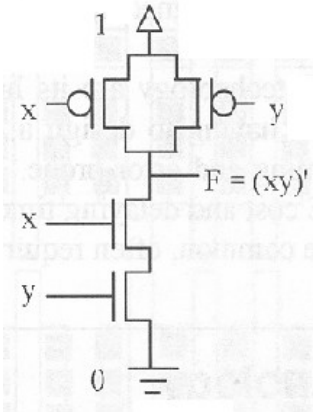
- **Full Custom**:
  - Designer creates and optimizes layouts for basic cells
  - Designer places cells resulting in masks
  - Designer provides interconnections which are translated to masks
  - Masks are sent to the fabrication plant
  - ICs ready to be tested/used

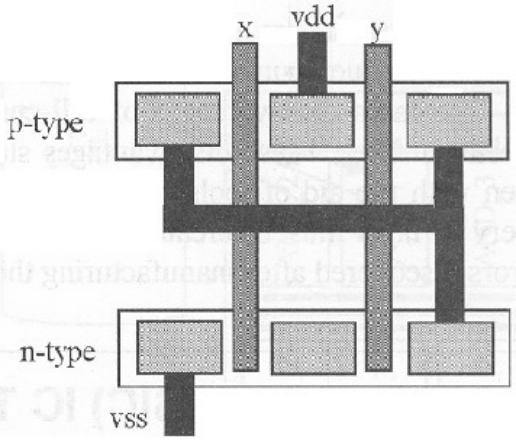


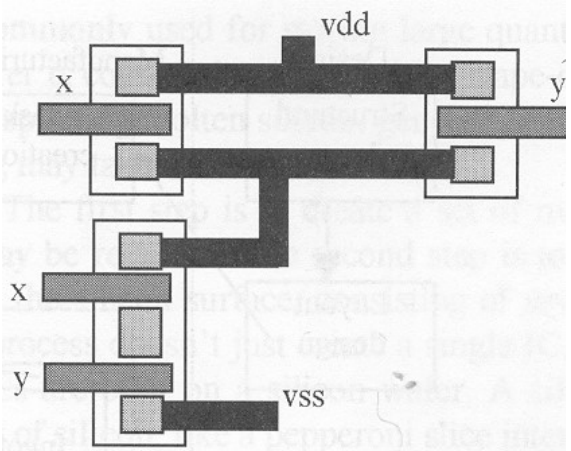
IC Design Styles

- Full custom gives best size, power, performance
- Hand design
  - Horrible time-to-market / flexibility / NRE cost…
  - Reserve for the most important units in a processor (ALU, Instruction fetch…)
- Physical design tools





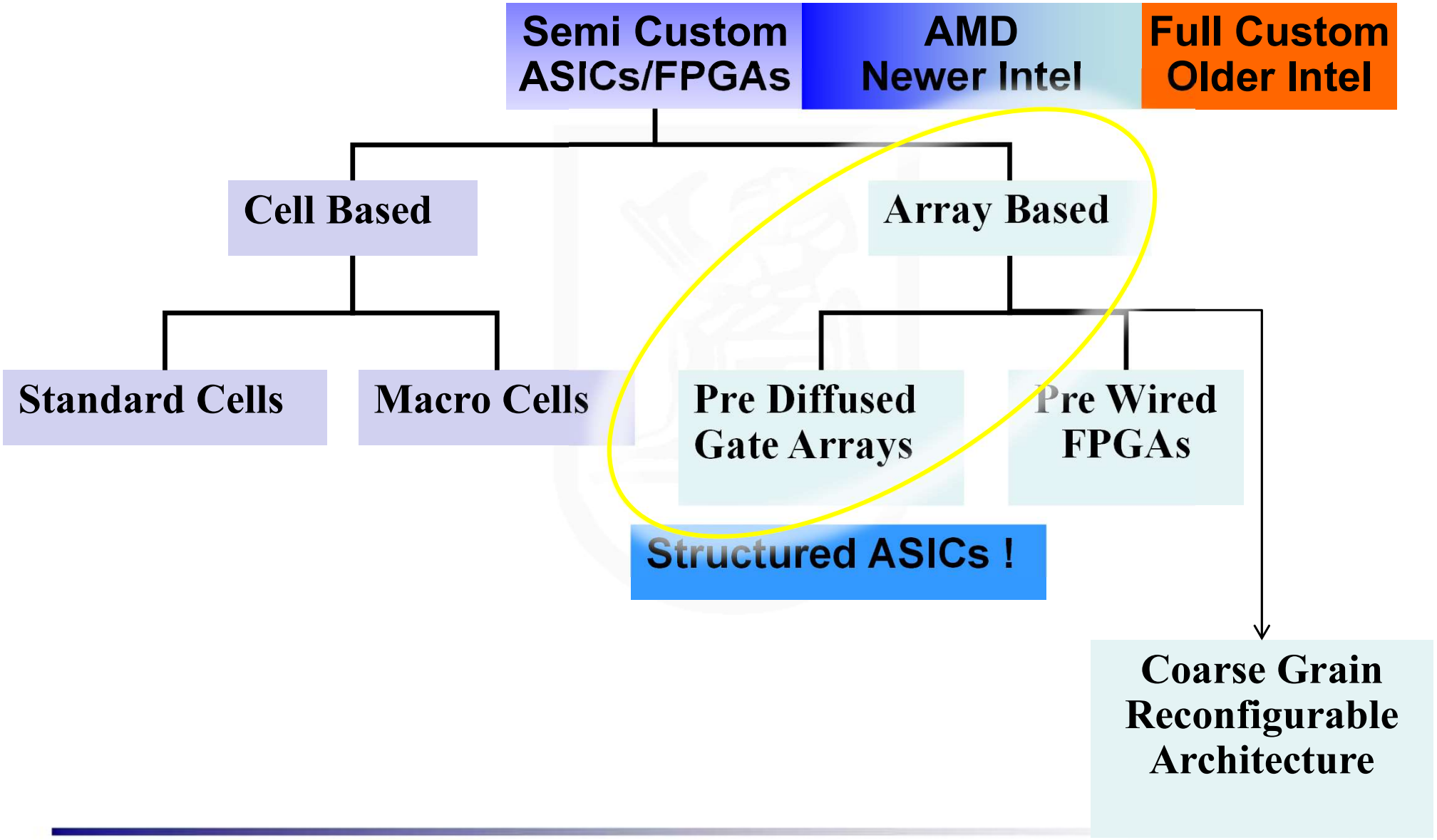




IC Design Styles

- Cost factors in full custom ICs:
  - For large-volume ICs:
    - packaging is largest cost
    - testing is second-largest cost
  - For low-volume ICs, design costs may swamp all manufacturing costs.
    - $10 million - $100 million.

Physical Implementation Styles

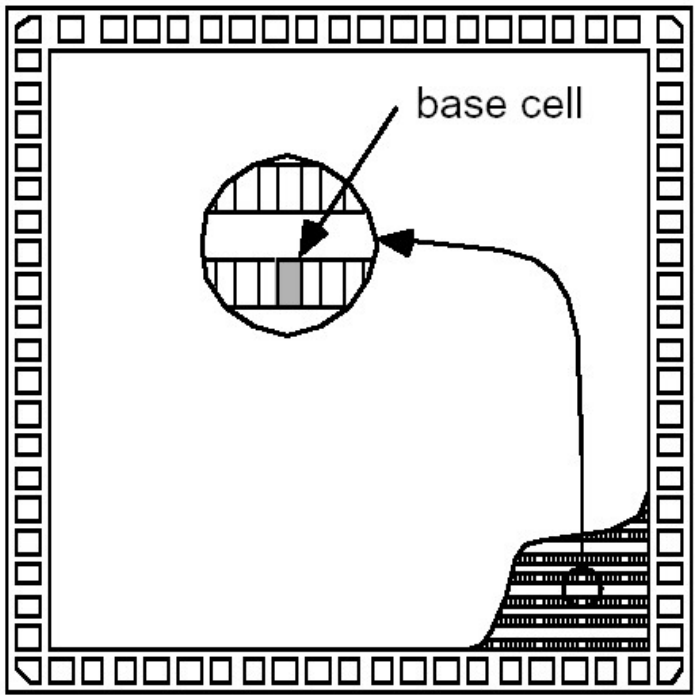


Array-Based

- A gate array chip contains prefabricated adjacent rows of PMOS and NMOS transistors
- The gate array is configured by the interconnect structure

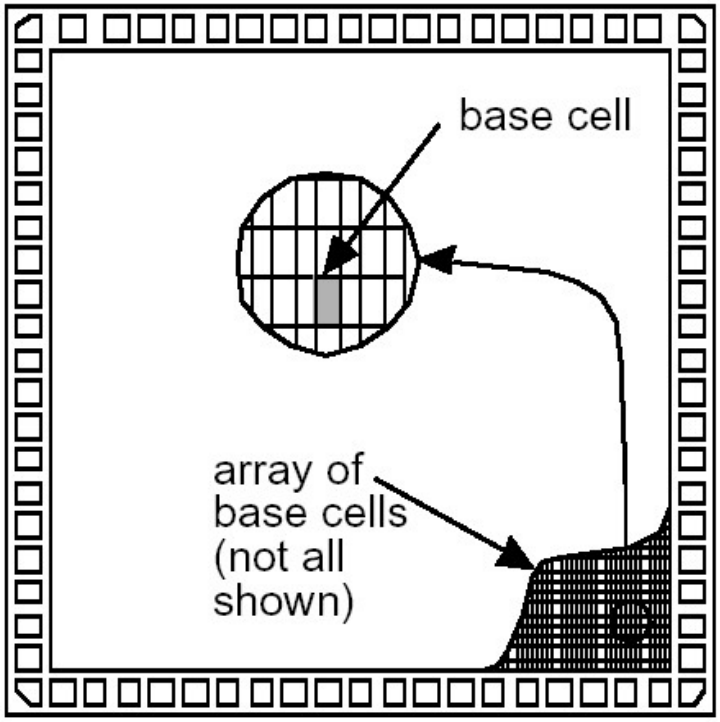
Channeled Gate Array

- **Only the interconnect is customized**
- **The interconnect uses spaces between rows of base cells**

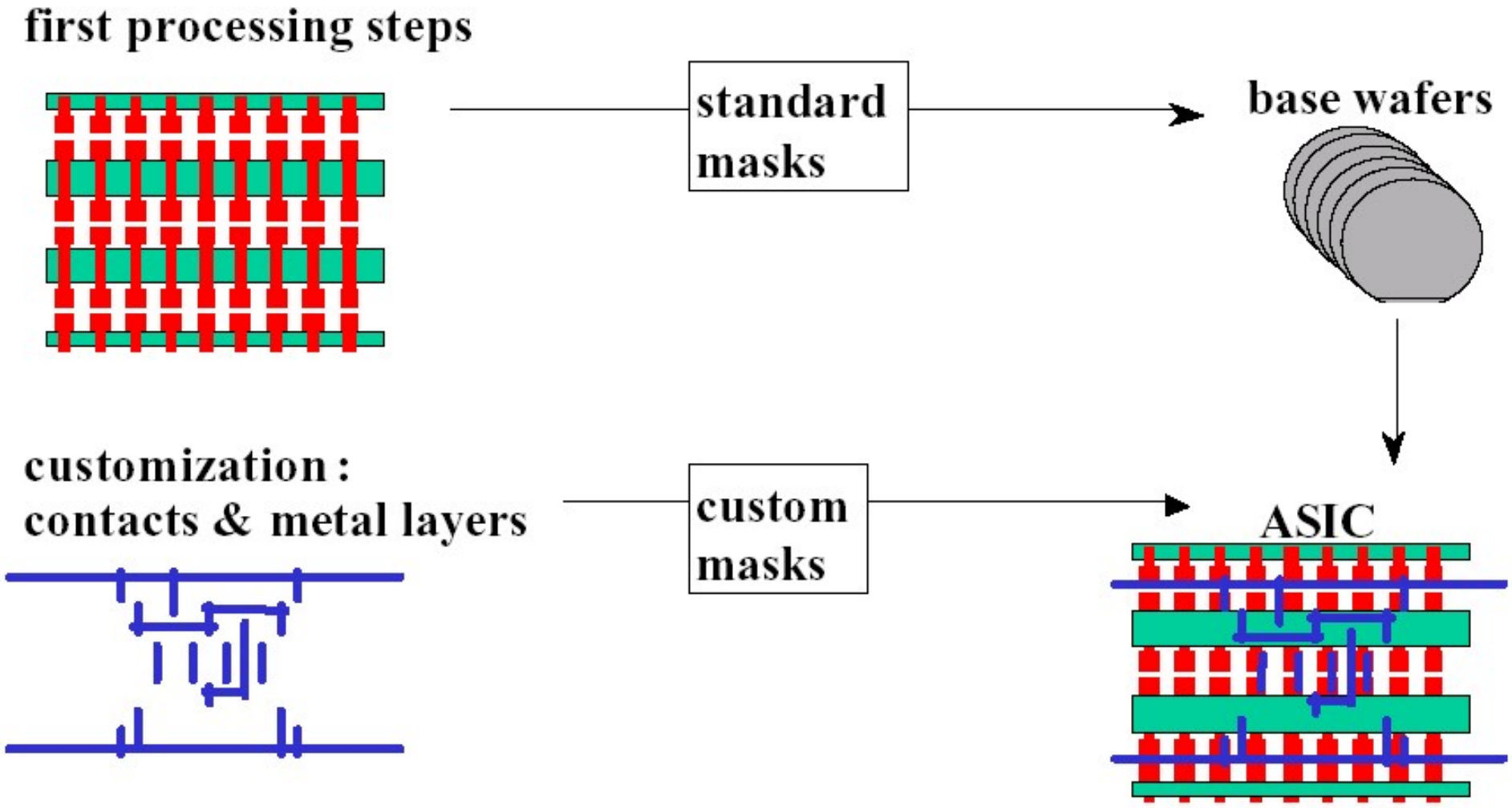


Channelless Gate Array (Sea of Gates)

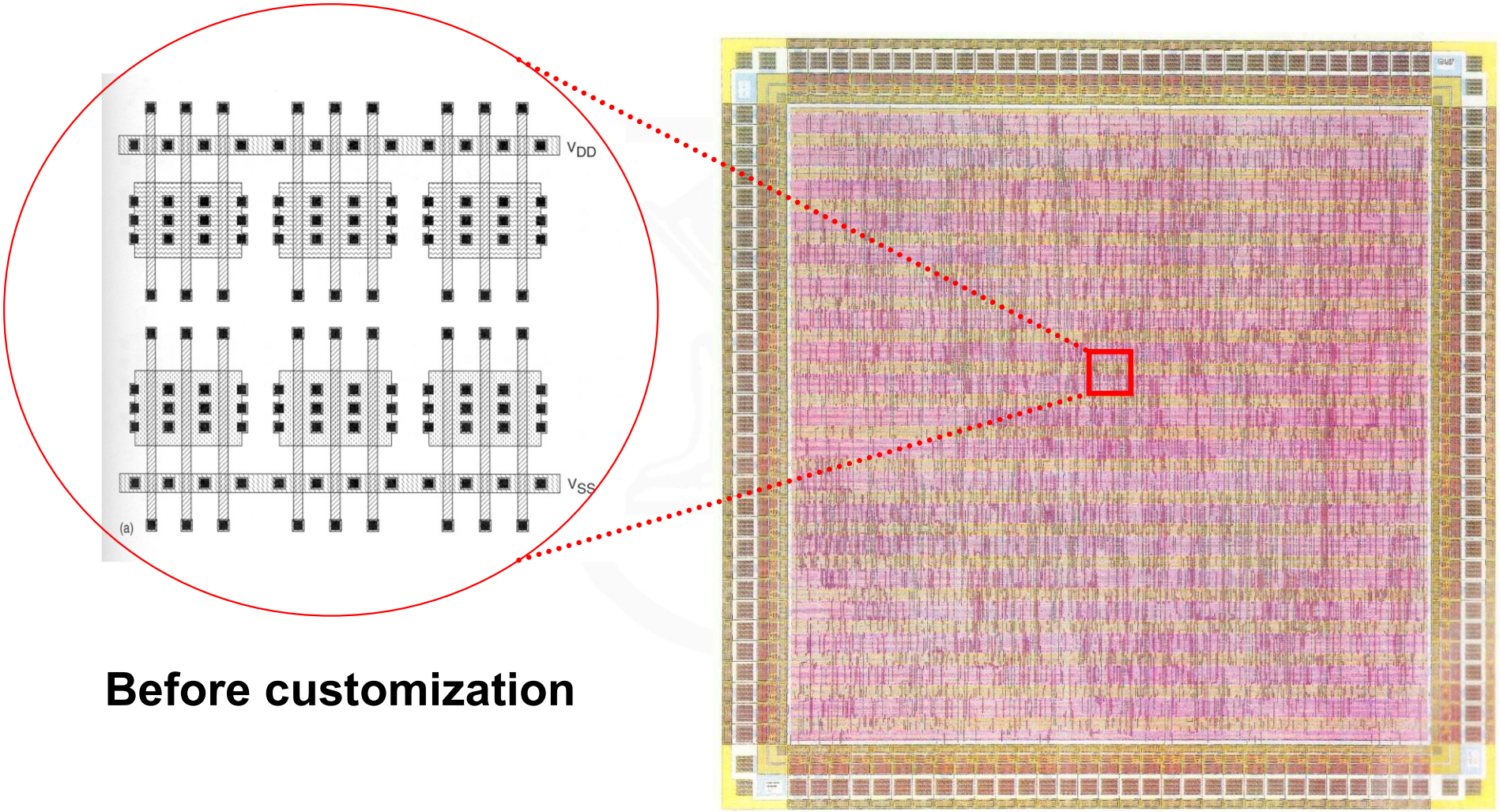
- **Only the interconnect is customized**
- **Cells are connected via unused transistors**



Mask Gate Array

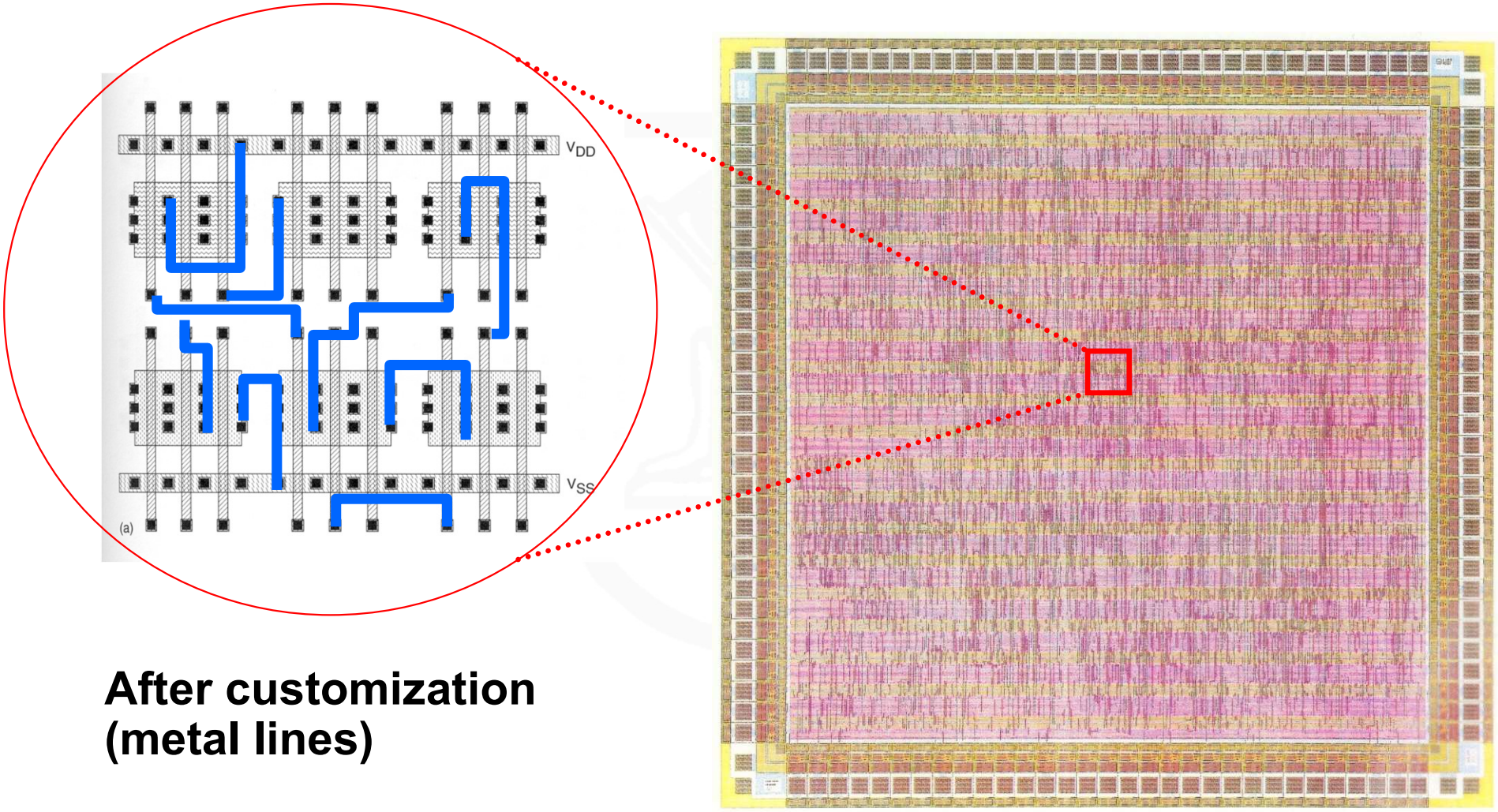


Mask Gate Array



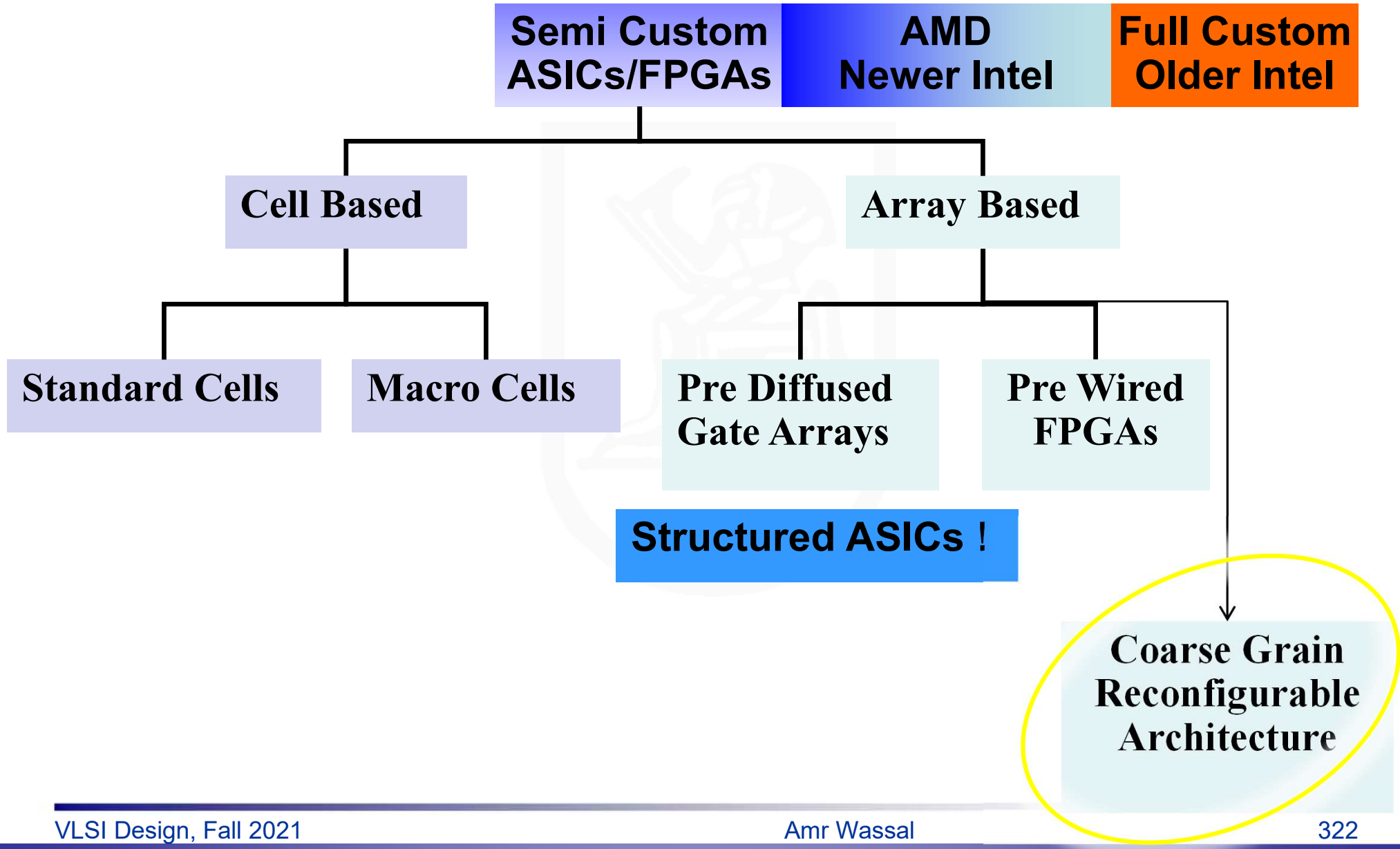
Before customization

Mask Gate Array



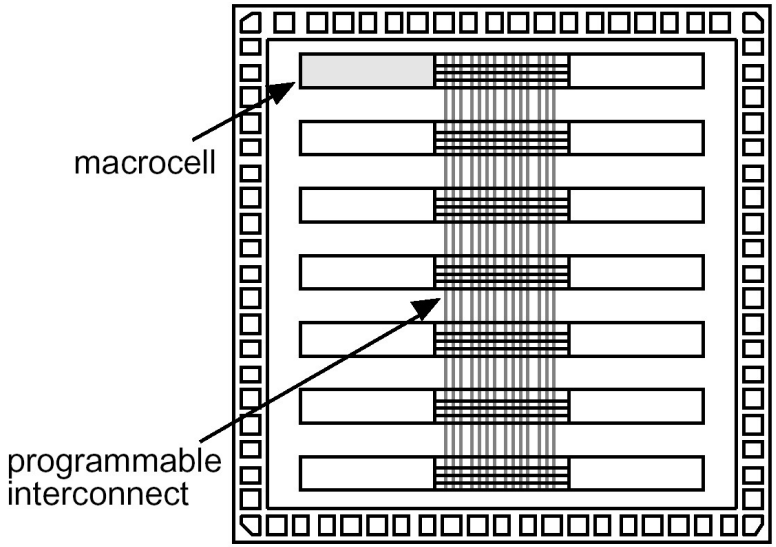
After customization (metal lines)

Physical Implementation Styles



Programmable Logic Device

- No customized mask layers or logic cells
- A single large block of interconnects
- Macrocells consist of programmable array logic followed by a flip-flop or latch

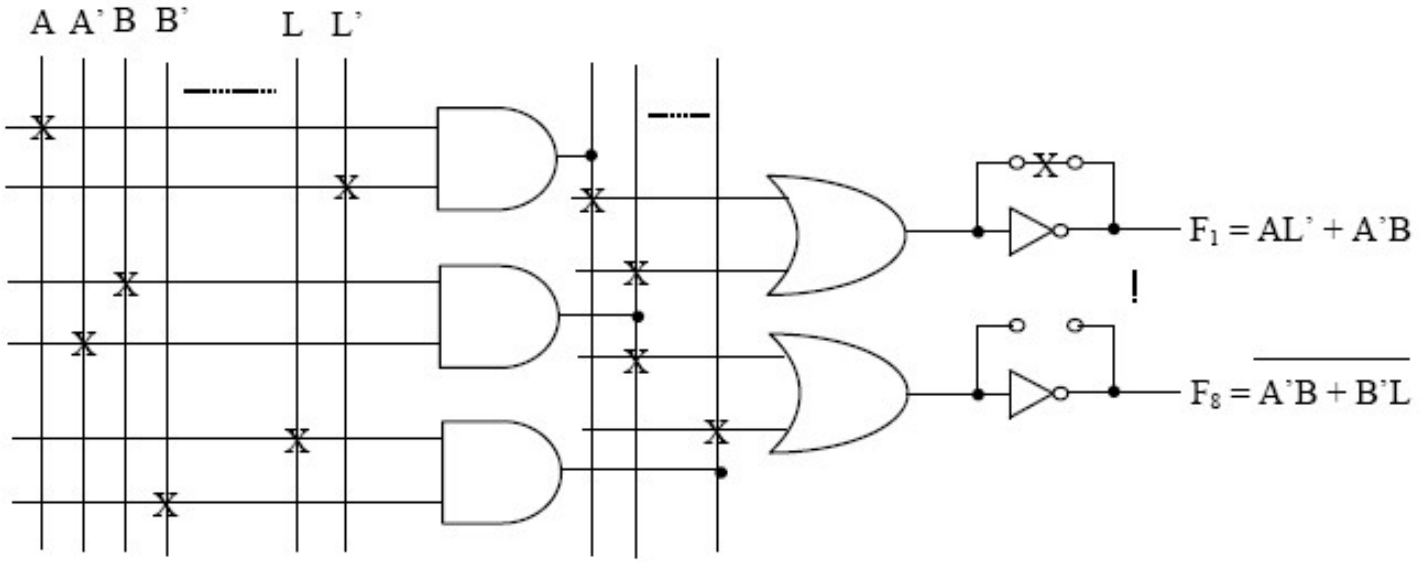


Evolution of PLDs

- **PLDs: Programmable Logic Devices**
- **Simple PLDs**
  - Programmable Logic Array (PLA)
  - Programmable Array Logic (PAL)
- **High-Capacity PLDs**
  - Complex Programmable Logic Device (CPLD)
  - Field Programmable Gate Array (FPGA)

Evolution of PLDs

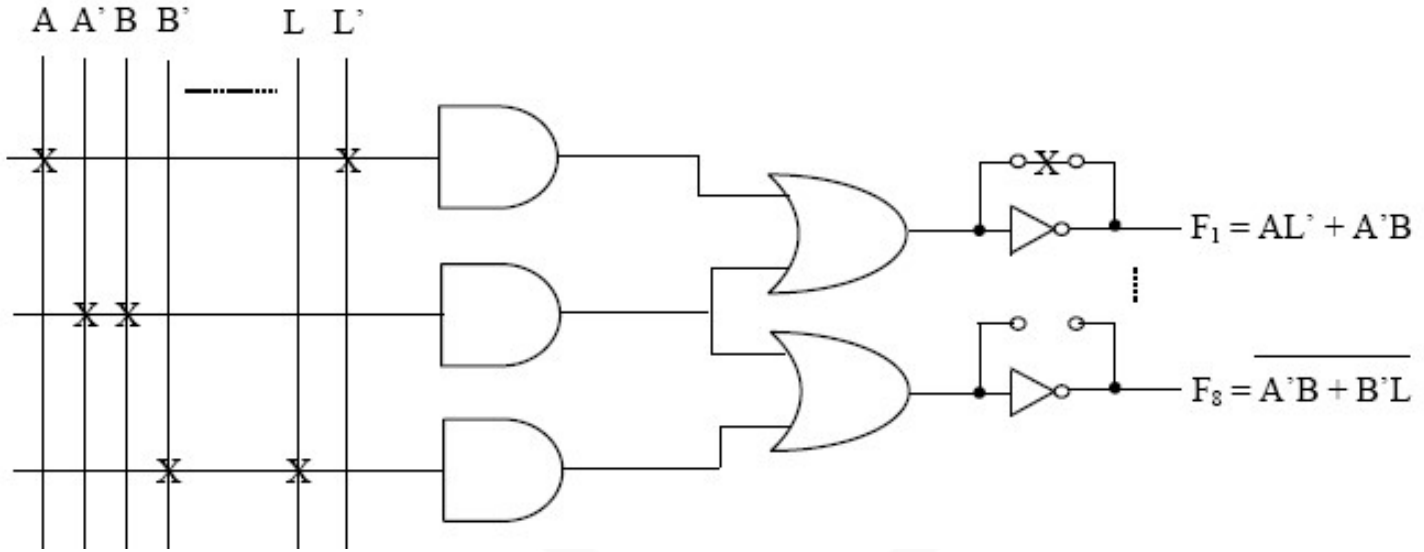
- **PLA**



  - + Well suited for implementing sum-of-products
  - - Expensive and slow (2 levels of configurable logic)

Evolution of PLDs

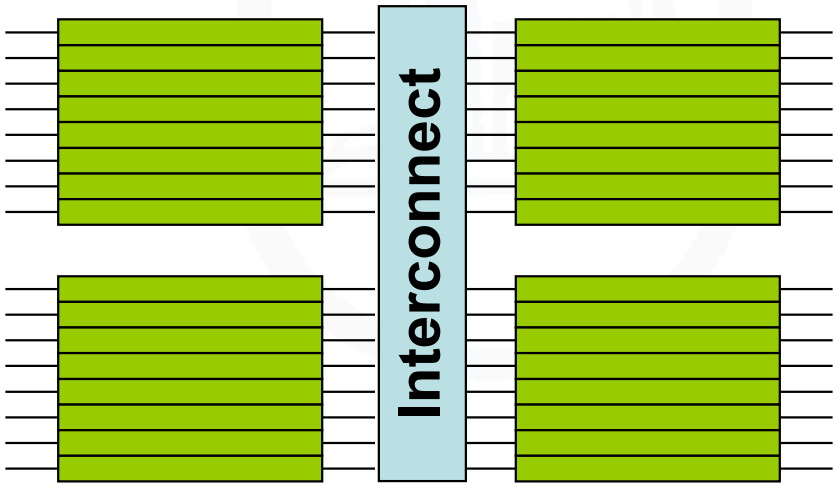
- **PAL**



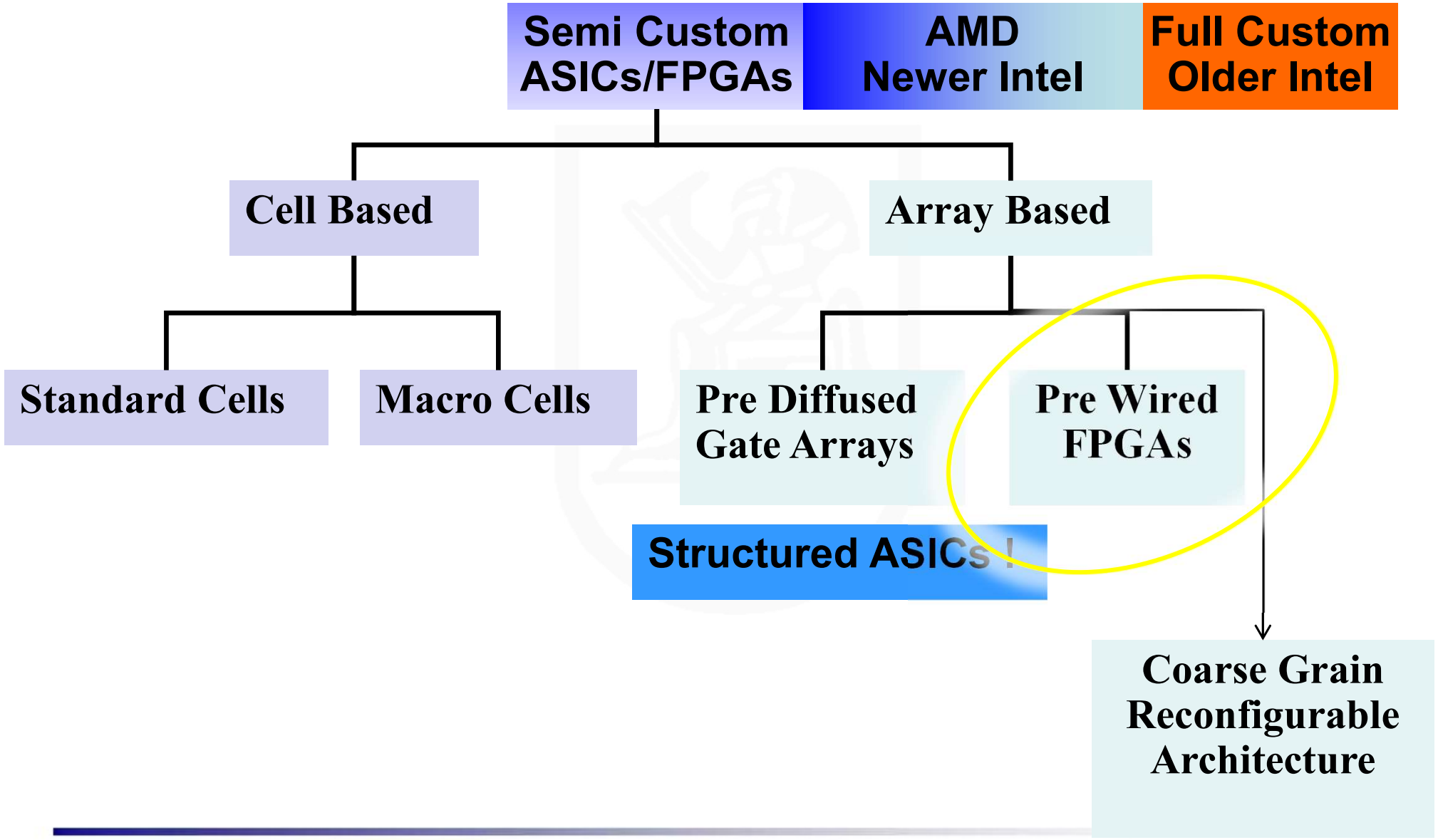
  - Single programmable plane
  - Lower cost, higher speed

Evolution of PLDs

- **CPLD**
  - An arrangement of multiple PAL-like blocks on a single chip

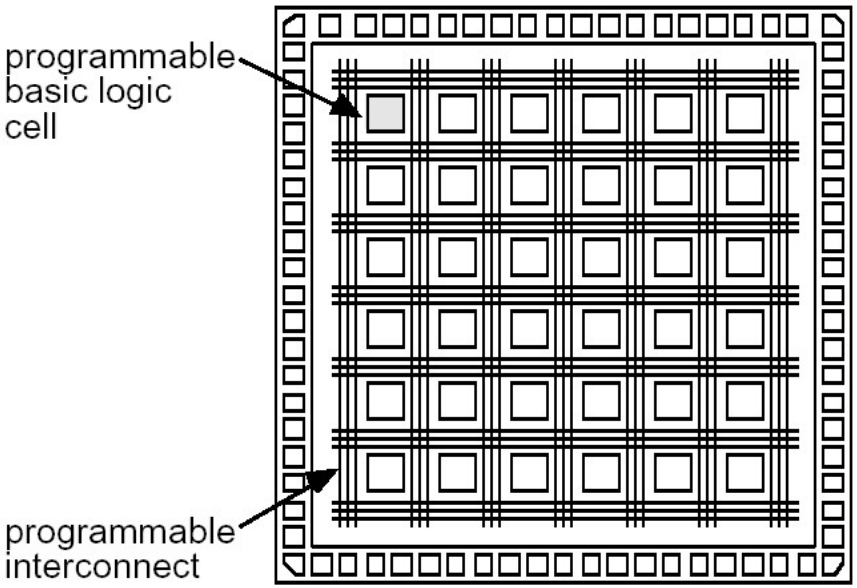


Physical Implementation Styles

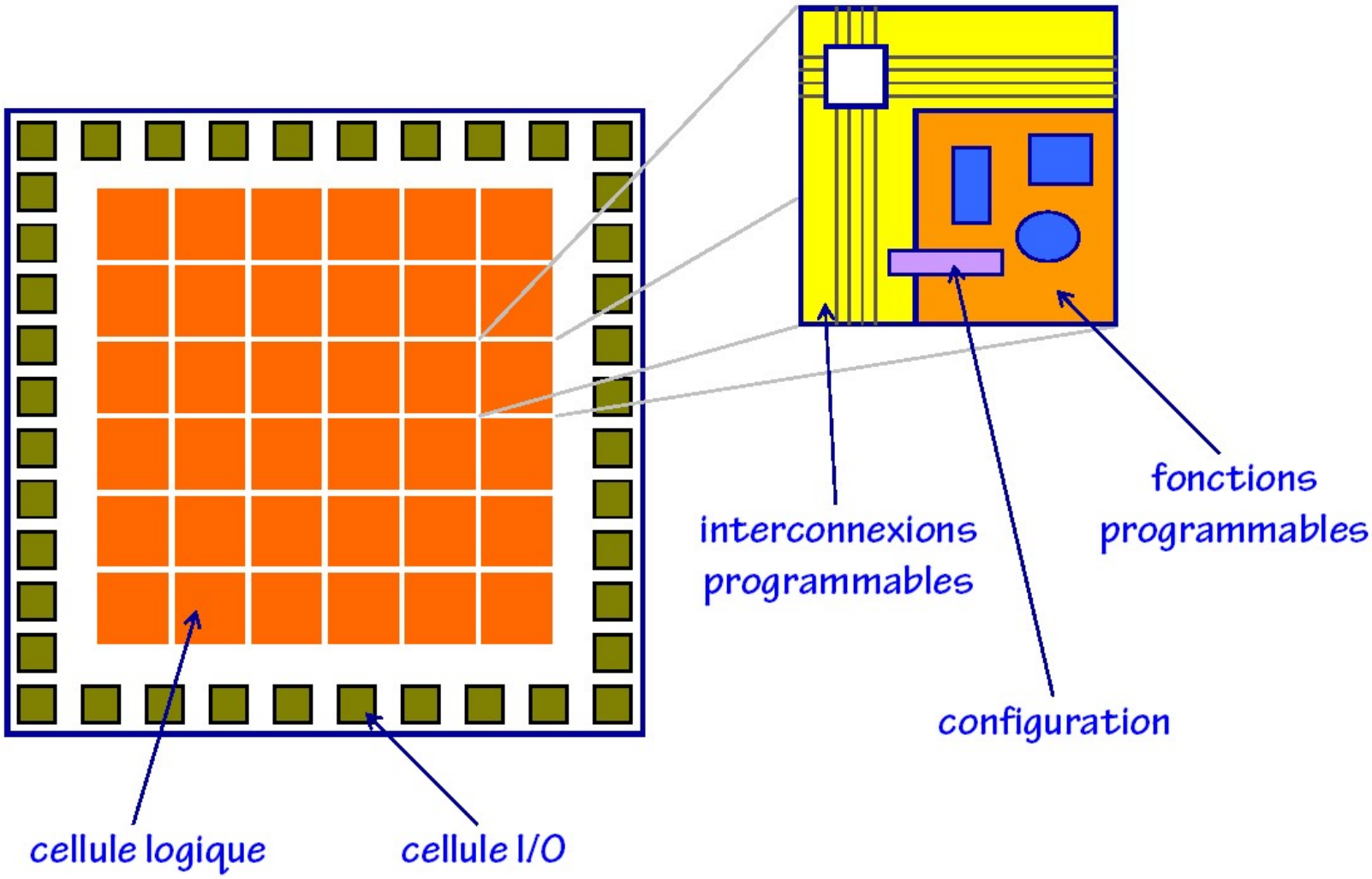


Field Programmable Gate Arrays

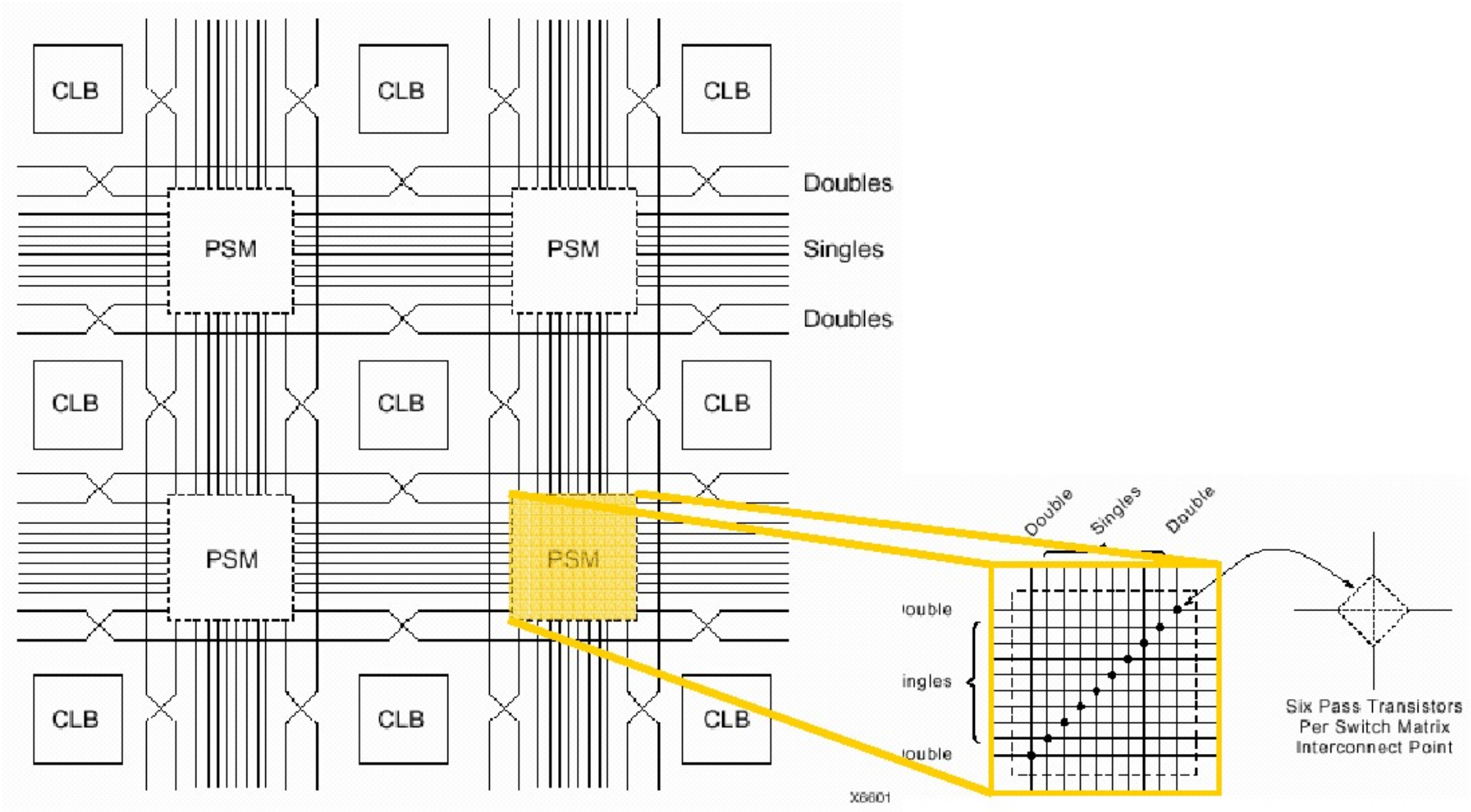
- None of the layers are customized
- Basic logic cells and interconnect can be programmed
- Basic cells can be SRAM based, Flash Memory based or fuse-based (one time programmable)



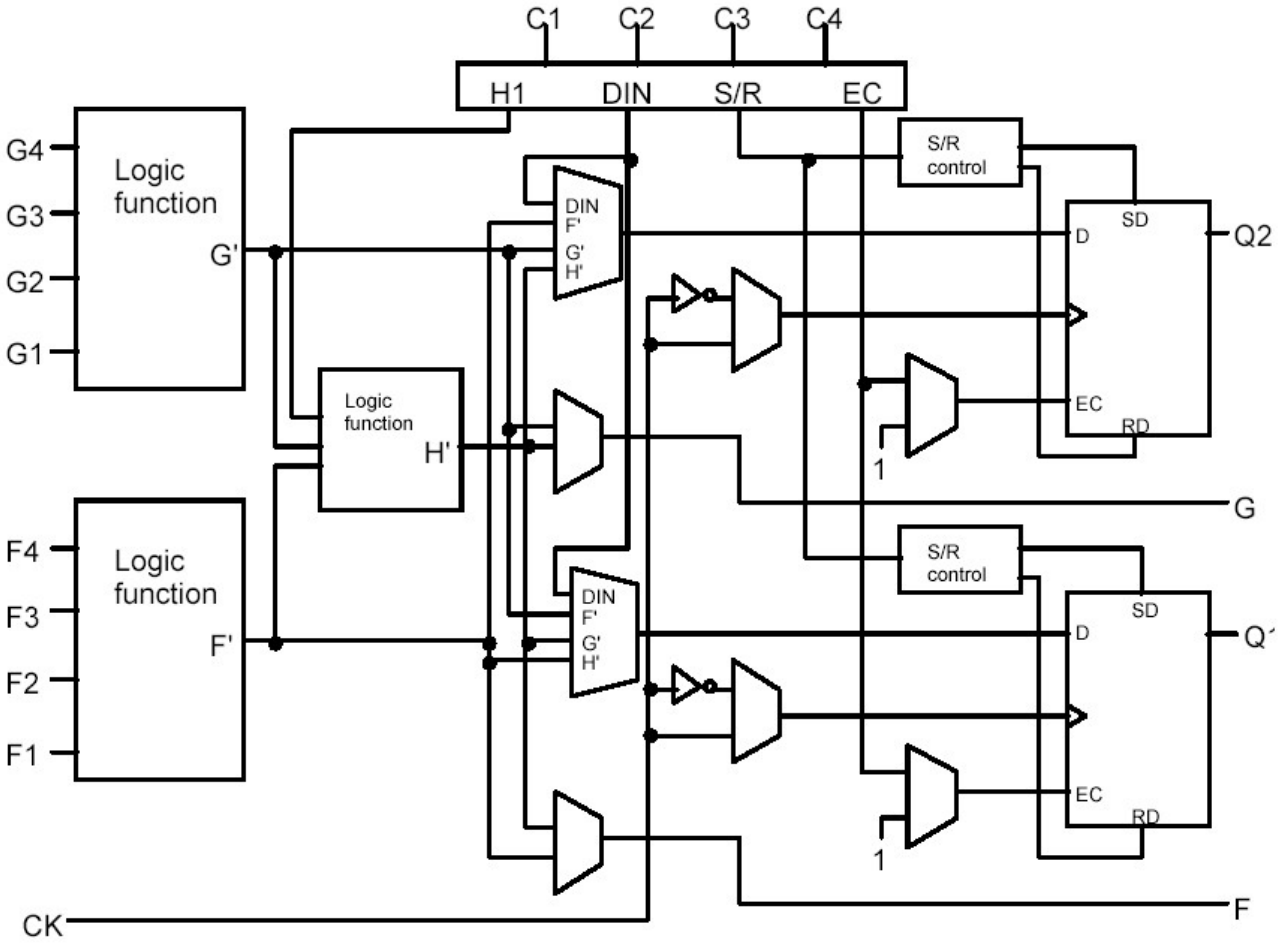
Field Programmable Gate Array



Field Programmable Gate Array



Field Programmable Gate Array



Internal structure of a CLB

Field Programmable Gate Arrays (Xilinx)

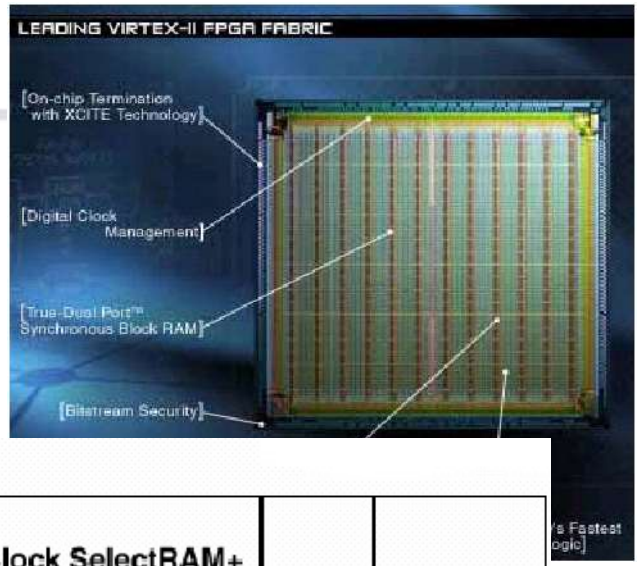


Table 1: **Virtex-II Pro FPGA Family Members**

Device	RocketIO Transceiver Blocks	PowerPC Processor Blocks	Logic Cells(1)	CLB (1 = 4 slices = max 128 bits)		18 X 18 Bit Multiplier Blocks	Block SelectRAM+		DCMs	Maximum User I/O Pads
				Slices	Max Distr RAM (Kb)		18 Kb Blocks	Max Block RAM (Kb)		
XC2VP2	4	0	3,168	1,408	44	12	12	216	4	204
XC2VP4	4	1	6,768	3,008	94	28	28	504	4	348
XC2VP7	8	1	11,088	4,928	154	44	44	792	4	396
XC2VP20	8	2	20,880	9,280	290	88	88	1,584	8	564
XC2VP30	8	2	30,816	13,696	428	136	136	2,448	8	644
XC2VP40	0(2) or 12	2	43,632	19,392	606	192	192	3,456	8	804
XC2VP50	0(2) or 16	2	53,136	23,616	738	232	232	4,176	8	852
XC2VP70	16 or 20	2	74,448	33,088	1,034	328	328	5,904	8	996
XC2VP100	0(2) or 20	2	99,216	44,096	1,378	444	444	7,992	12	1,164
XC2VP125	0(2), 20, or 24	4	125,136	55,616	1,738	556	556	10,008	12	1,200

Advantages of FPGAs

- Low tooling costs
  - No mask fabrication, packaging, testing
- Rapid turnaround
  - Fast design modification
  - Configured in minutes by the user
    "… in a high-technology environment, a 6-month delay in product delivery can cut the lifetime profit of a product by as much as 33%..."
- Low risk
  - Low cost of errors due to low initial cost and rapid turnaround
- Effective design verification

Advantages of FPGAs

- Overcomes simulation inaccuracies by using an actual part as a prototype
- Low testing costs
  - Thorough testing of each part is already done by the manufacturer
- Standard-product advantages
  - The manufacturer incurs the initial cost of moving the chip to a new process technology
  - The user gets faster and cheaper chips

Advantages of FPGAs

- FPGAs provide product life-cycle advantages
  - User can easily adapt to product life-cycle
    - Development stage
    - Start-up
    - Production
    - Rapid changes in demand (up or down)
    - Product enhancements, upgrades

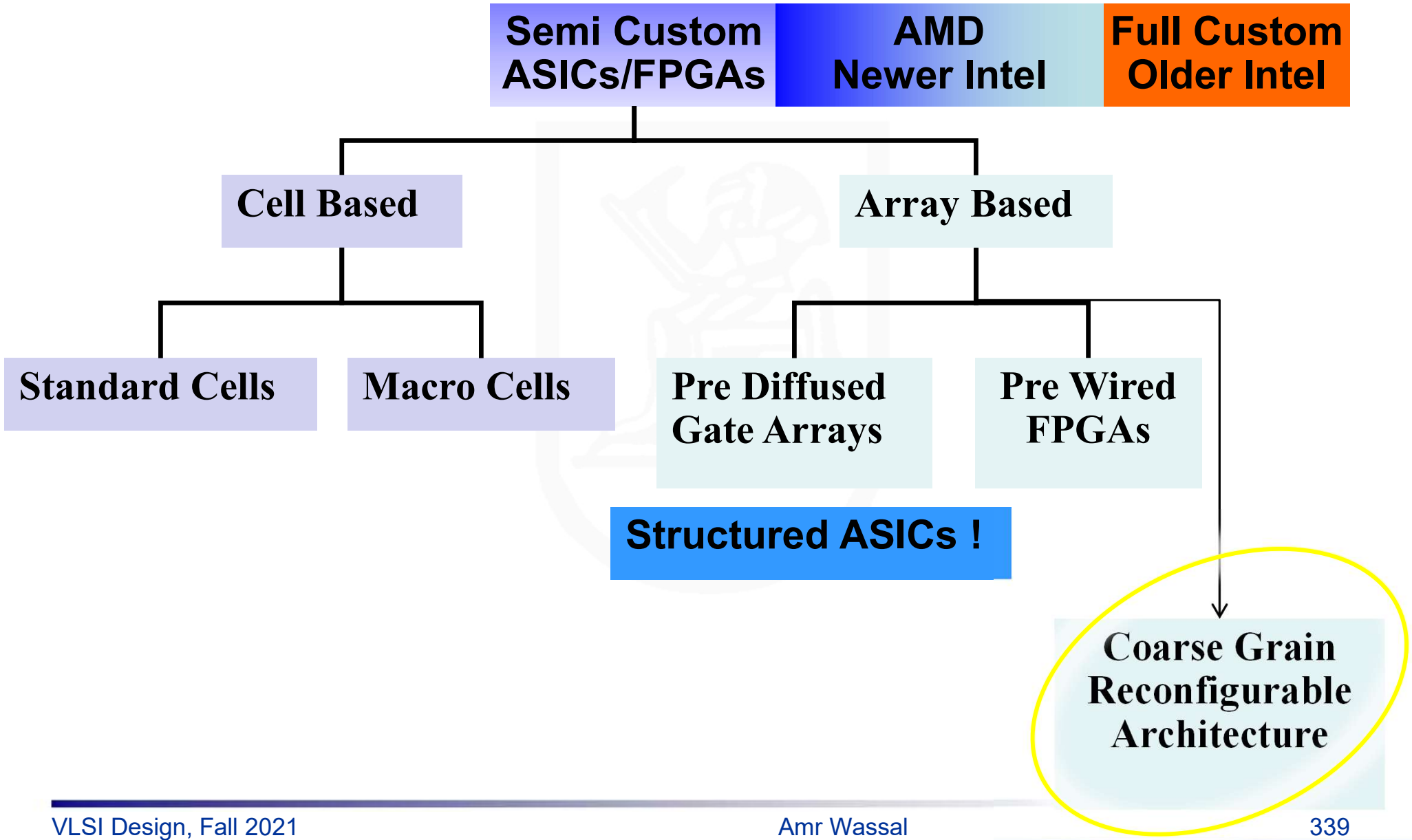
Disadvantages of FPGAs

- Overhead circuitry
  - For programmability…
- Chip size
  - About 10 times larger (for same gate capacity) than gate arrays
  - Limited gate capacity (improving)
- Chip cost
  - Low startup cost is good for low production volume, but offset by high per-chip cost

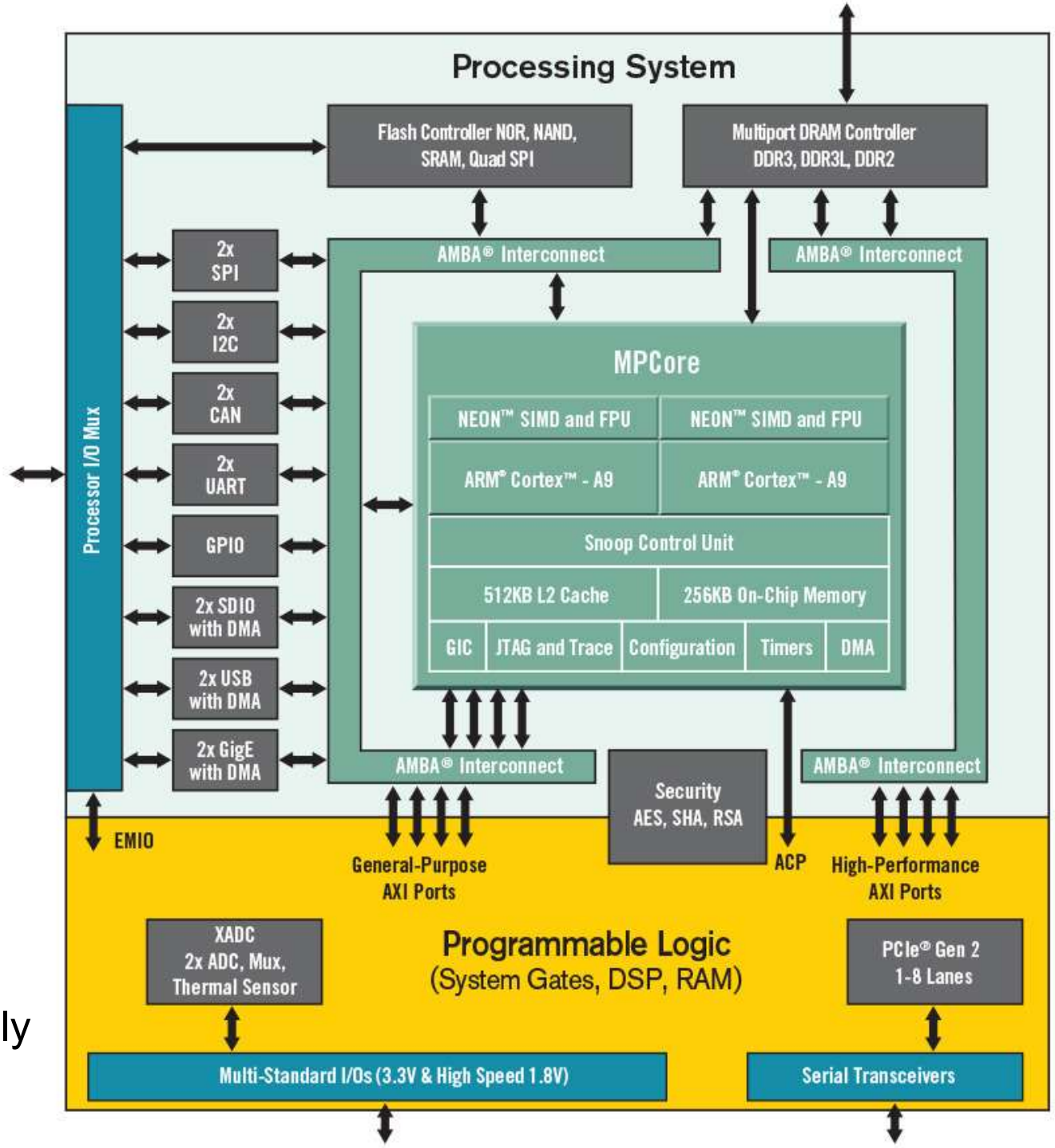
Disadvantages of FPGAs

- Speed of circuitry
  - Programming points introduce more resistance in signal paths
  - Larger area means longer wires, more resistance, more capacitance
  - FPGAs are significantly slower than gate arrays (for the same manufacturing process)
- Design methodology
  - " … too easy …"
  - "try-it-and-see-what-happens" thinking leads to sloppy designs, inferior quality products

Physical Implementation Styles



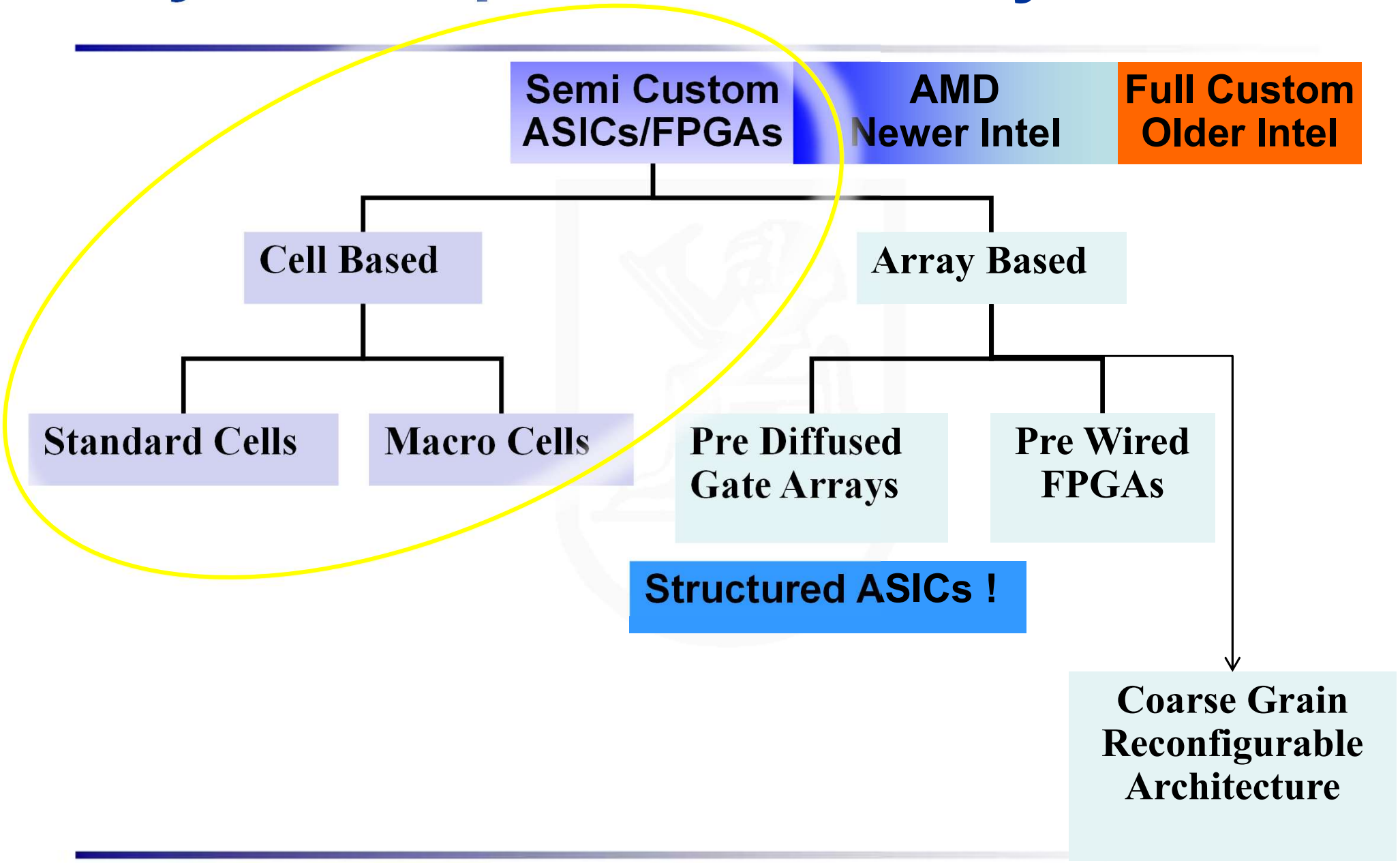
Coarse Grain Reconfigurable Architectures



Xilinx Zynq-7000
All Configurable SoC Architecture

See also the Ultrascale family

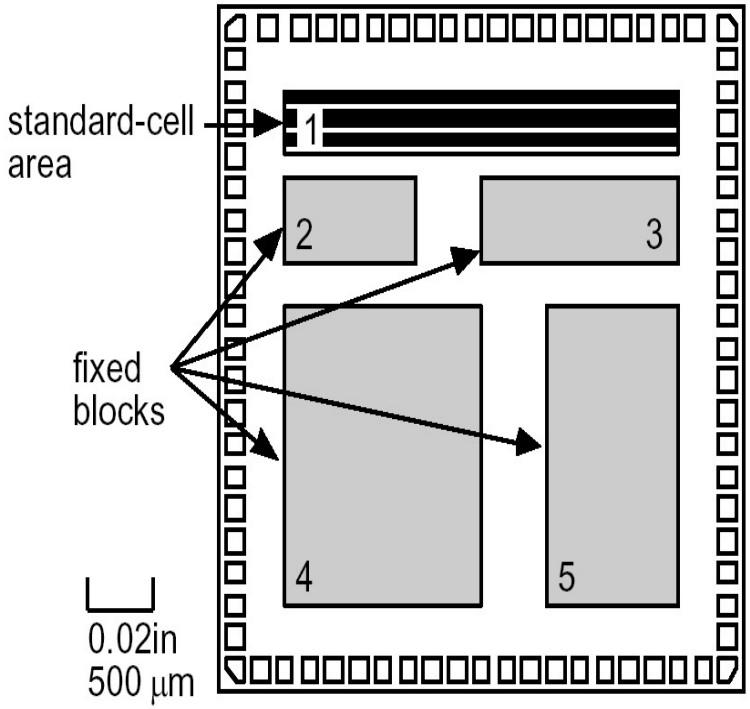
Physical Implementation Styles



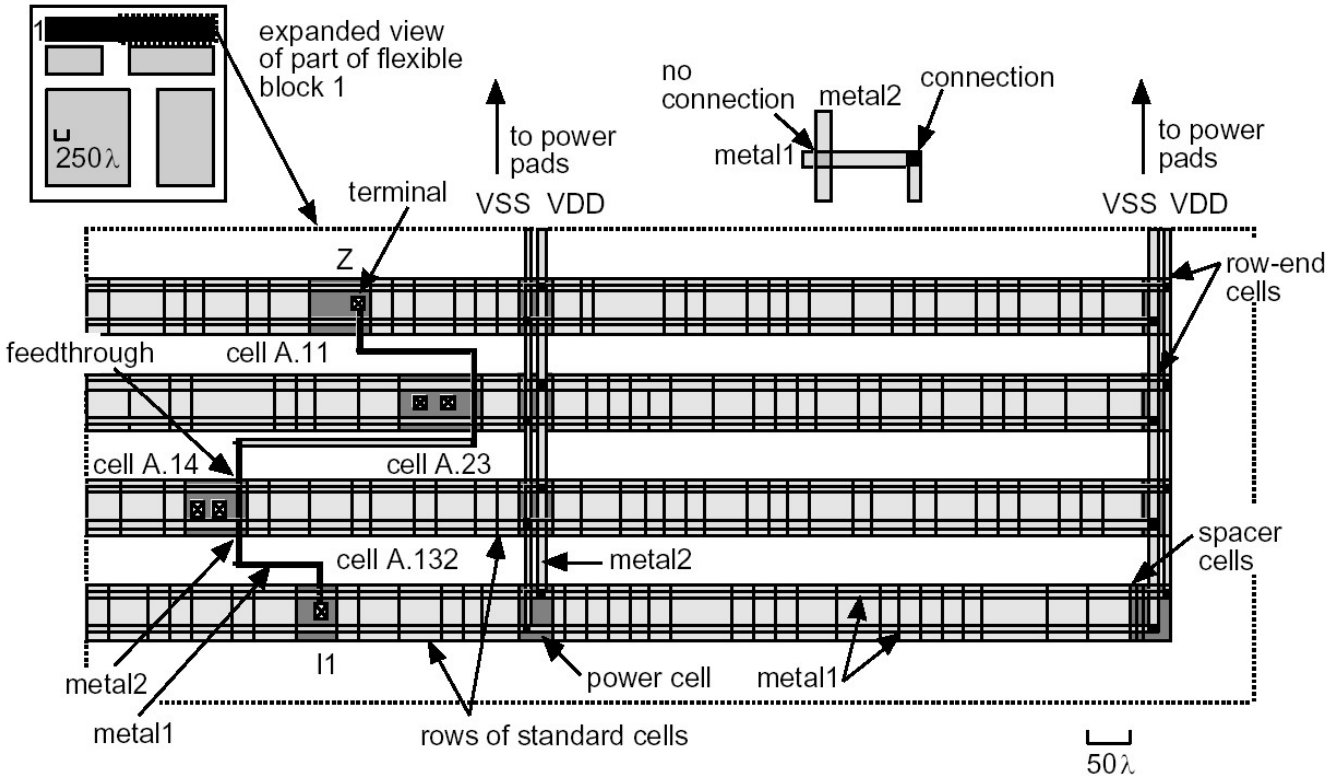
ASICs & Standard Cells

- ASICs = Application Specific Integrated Circuits
- Also, ASSPs = Application Specific Standard Products

- All mask layers are customized
- The standard cell library defines logic elements of varying complexity: SSI, MSI logic, data path blocks, memories and system-level blocks.



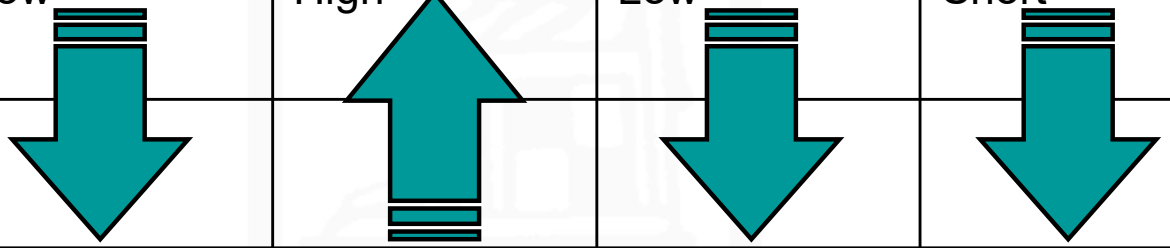
Standard Cells



- **Cells are configured in rows and have constant height and variable width**
- **Each cell is optimized for an efficient implementation**

Comparing FPGA, Gate Array, Standard Cells

	Initial Cost (NRE)	Cost per part (Production)	Performance	Fabrication Time
FPGA	Low	High	Low	Short
Gate Array				
Standard Cell	High	Low	High	Long

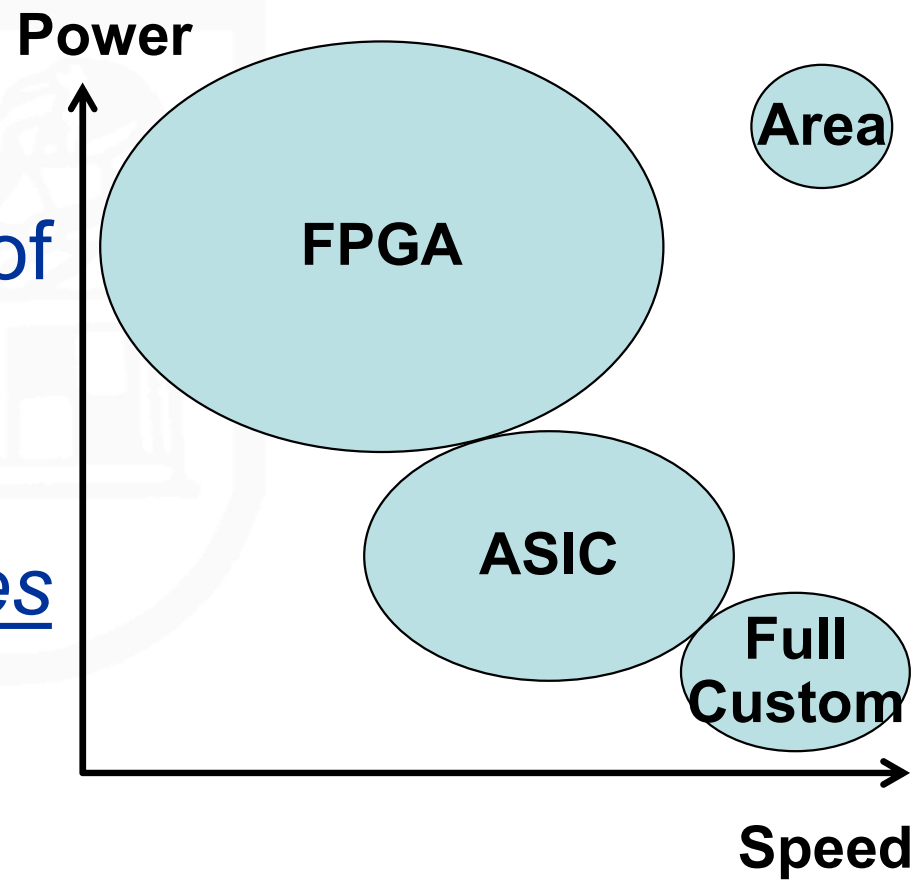


IC Design Styles

	Full Custom	**Semi Custom (ASIC)**	**FPGA**
Features	Optimize every transistor	Use of pre-existing libraries or IPs	Programming-like flow (HDL)
NRE cost	Very high	Medium	Very low
Cost per chip	Low	Medium	High
Performance	Best	Medium	Low
Power	Lowest	Medium	High

IC Design Styles

- These are non-competing styles
- Each has its domain of application
- Need to learn *all styles* and know how to chose which one to use



IC Design Styles