An Overview of Standard Cell Based Digital VLSI Design

With examples taken from the implementation of the 36-core AsAP1 chip and the 1000-core KiloCore chip

Zhiyi Yu, Tinoosh Mohsenin, Aaron Stillmaker, Bevan Baas
VCL Laboratory, UC Davis

Outline

- ***Overview of standard cell-based design***
- Design of the AsAP1 and KiloCore chips including CAD Tool Flow

Standard cell vs. Full-custom IC design

- Standard-cell based IC design
  - Design using standard cells
  - Standard cells come from library provider
  - Many different choices for cell size, delay, leakage power
  - Many EDA tools to automate this flow
  - Shorter design time
- Custom IC design (e.g., magic)
  - Design all by yourself
  - Higher performance
  - Lower energy per workload (lower power)
  - Smaller chip area

Standard cell based VLSI design flow

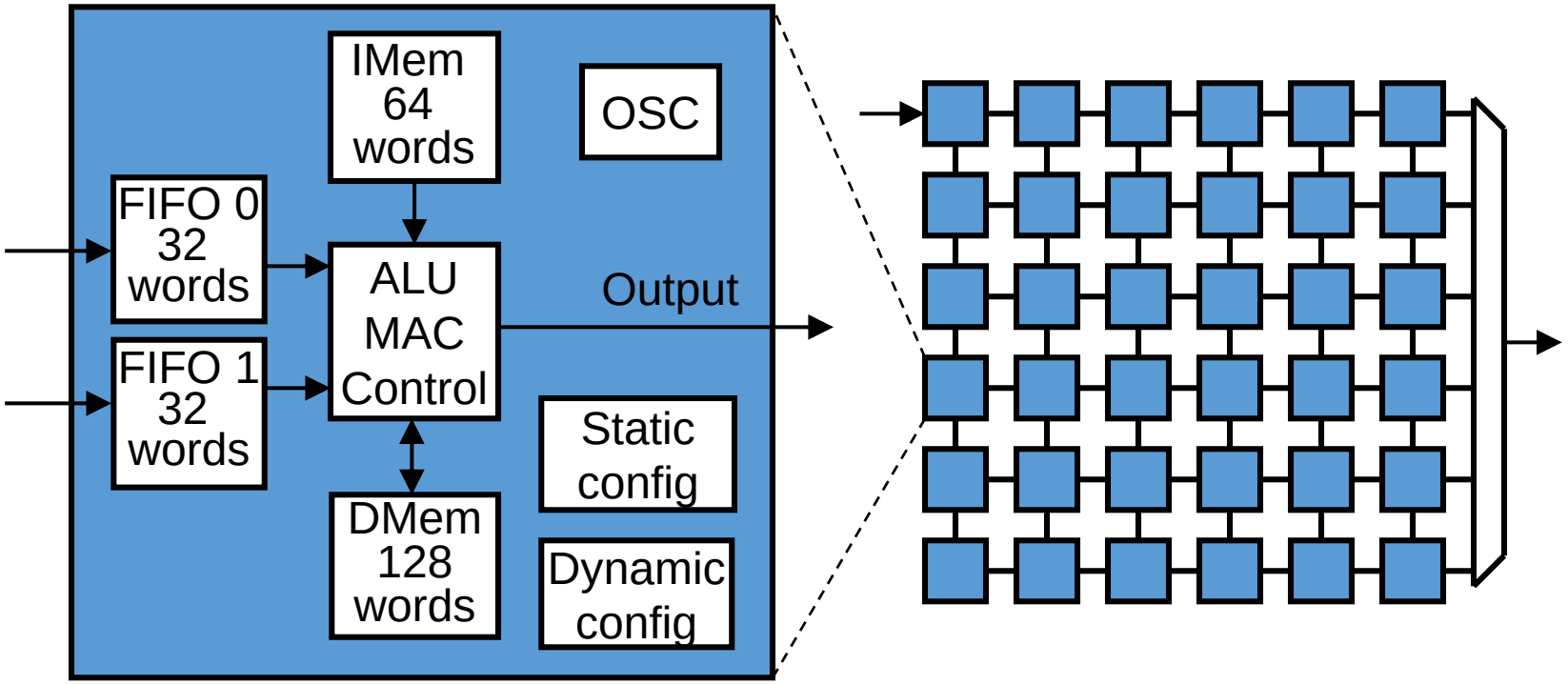
- "Front end"
  - System specification and architecture
  - HDL (verilog or VHDL) coding
    - Behavioral simulations using RTL (HDL)
  - Synthesis
  - Gate-level simulations
- "Back end"
  - Floorplanning, Power grid design
  - Standard-cell Placement
  - Interconnect routing
  - DRC (Design Rule Check)
  - LVS (Layout vs Schematic)
  - Dynamic simulation and static timing analysis

Outline

- Overview of standard cell-based design
- ***Design of the AsAP1 and KiloCore chips including CAD Tool Flow***

AsAP1 Block Diagram

- GALS array of identical processors
  - Each processor is a reduced complexity programmable DSP with small memories
  - Each processor can receive data from any two neighbors and send data to any of its four neighbors

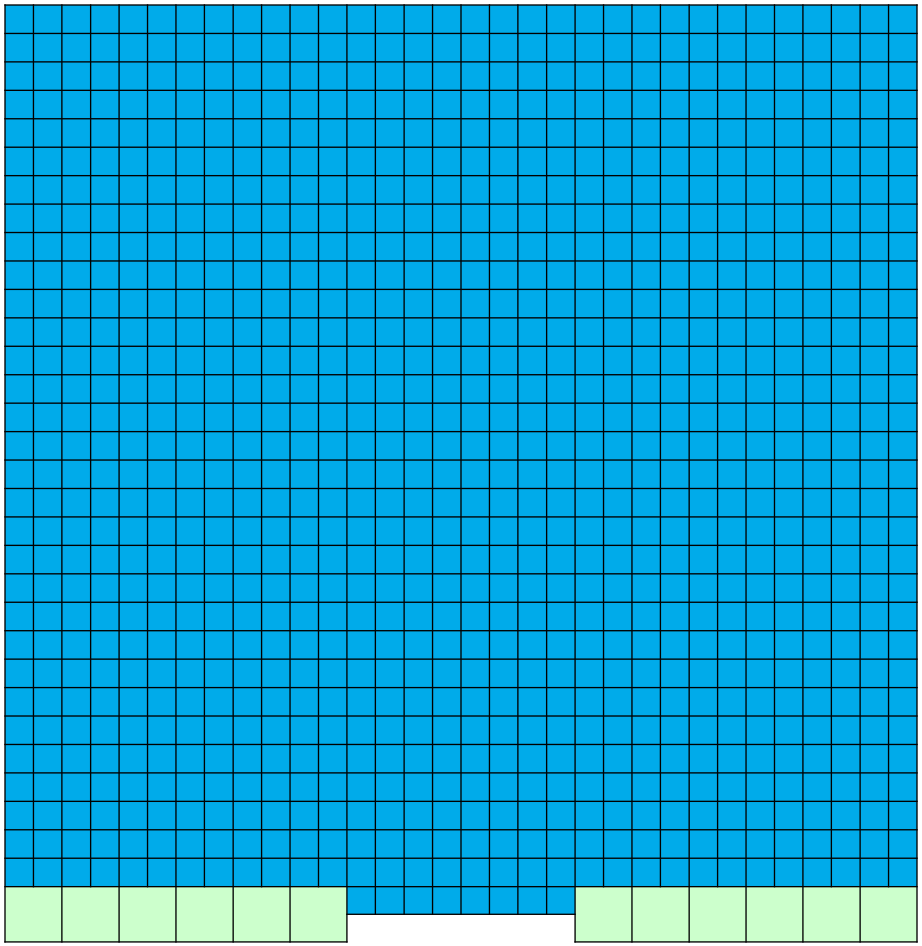


KiloCore

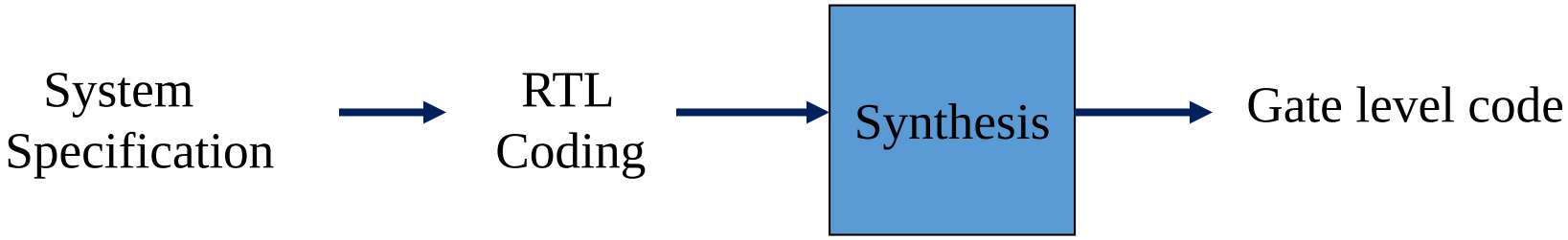
- Developed by the VLSI Computation Laboratory at UC Davis, with a similar architecture to AsAP (Asynchronous Array of Processors)
- A processing chip containing multiple uniform simple processor elements
- Globally Asynchronous Locally Synchronous (GALS)
  - Each processor has its local clock generator
- Each processor can communicate with its neighbor processors using dual-clock FIFOs

KiloCore Design

- Contains 1,000 processors on one chip
- Fastest clock rate processor designed at a university
- 12 memories containing 64 KB each for 768 KB of shared memory

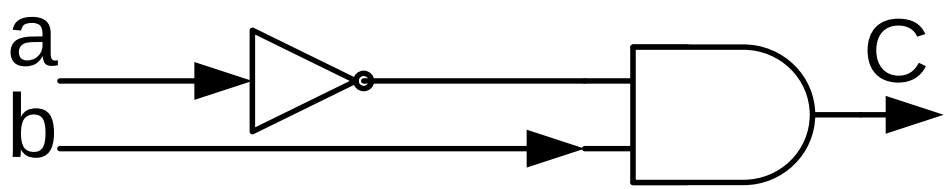


Simple Diagram of the Front-End Design Flow

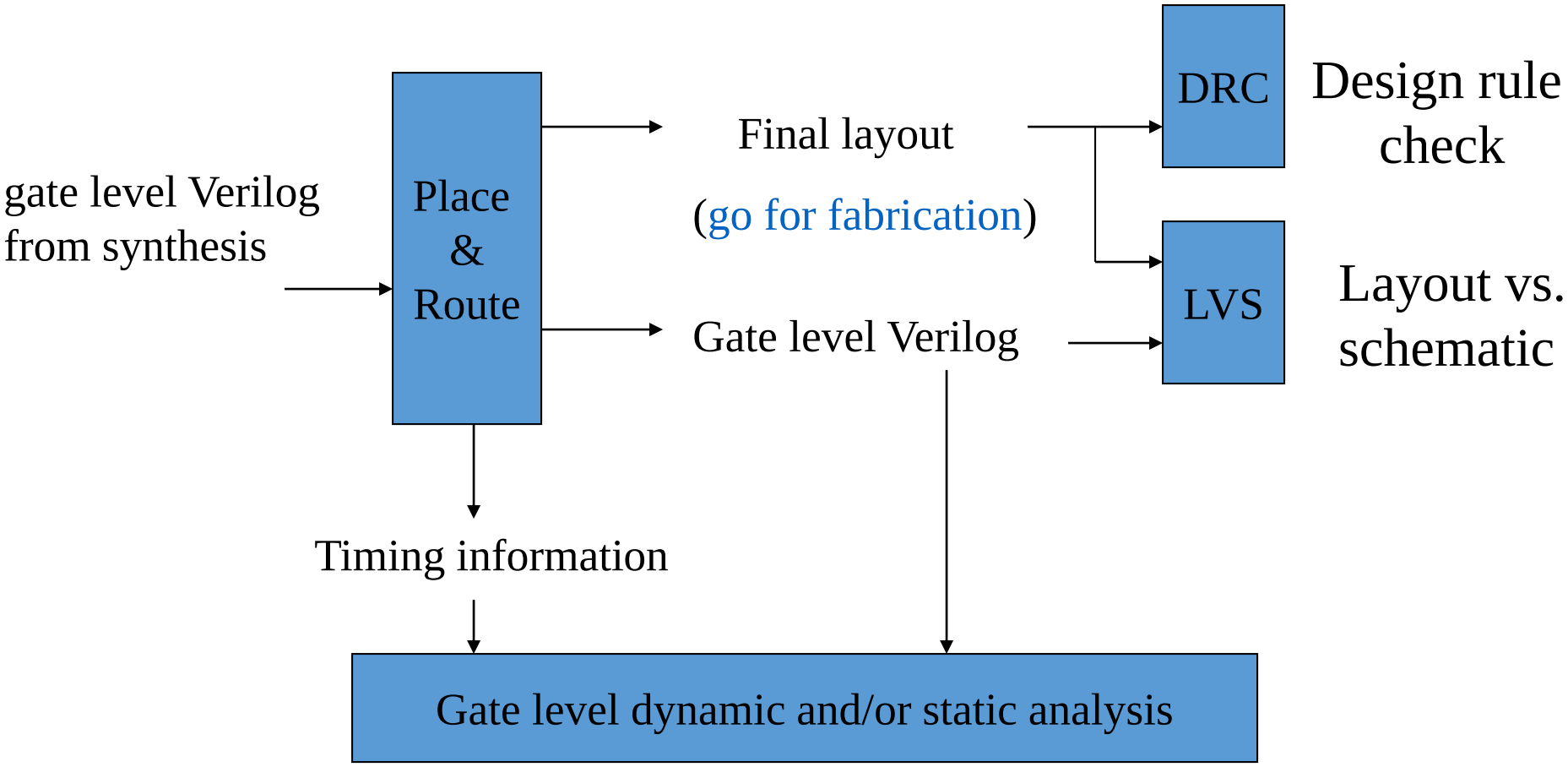


Example: `c = !a & b`

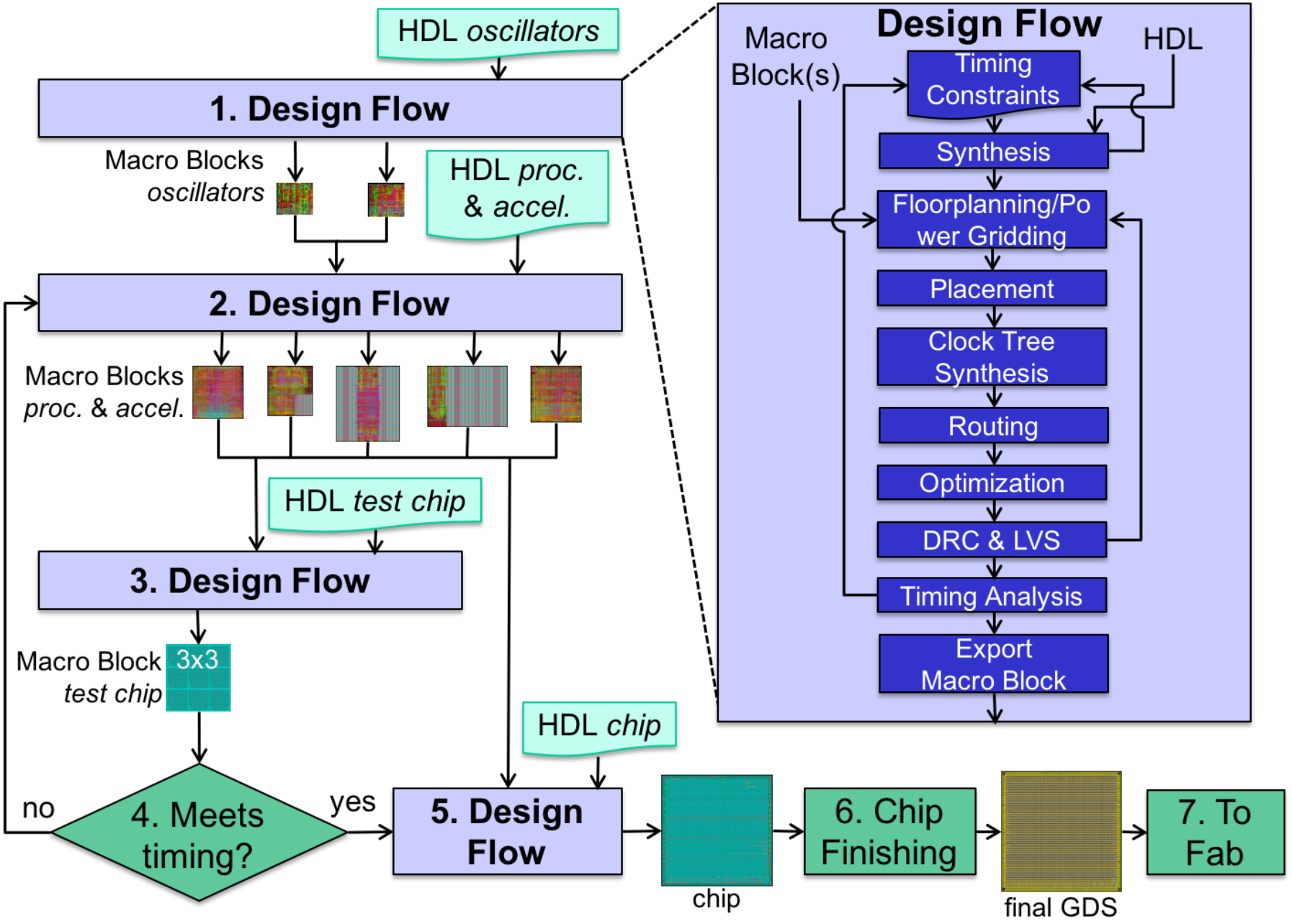
```
INV (.in(a), .out(a_inv));
AND (.in1 (a_inv), .in2(b),
     .out(c));
```



Simple diagram of the Back-end design flow



Back-End: Physical Design Flow



Back-end design of AsAP1

- Technology: TSMC 0.18 μm CMOS
- Standard cell library: Artisan
- Tools
  - Synthesis: Synopsis Design compiler
  - Placement & Route: Cadence Encounter
  - DRC & LVS: Mentor Calibre
  - Static timing analysis: Primetime

Back-End Design of KiloCore

- Technology: IBM 32 nm PD-SOI CMOS
- Standard cell library: ARM-Artisan
- Tools
  - Synthesis: Synopsis Design compiler
  - Placement & Route: Cadence Encounter
  - DRC & LVS: Mentor Calibre
  - Static timing analysis: Primetime and UltraSim (Spice)

Flow of Placement and Routing

- Import needed files
- Floorplan
- Placement & in-place optimization
- Clock tree generation
- Routing

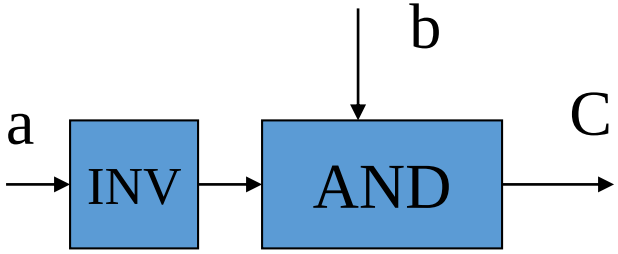
Import Needed Files

- Gate-level verilog (.v)
- Geometry information (.lef)
- Timing information (.lib)

```
INV (.in (a),      .out (a_inv));
AND (.in1 (a_inv),   .in2 (b),  .out (c));
```

INV: 1um width AND: 2 um width

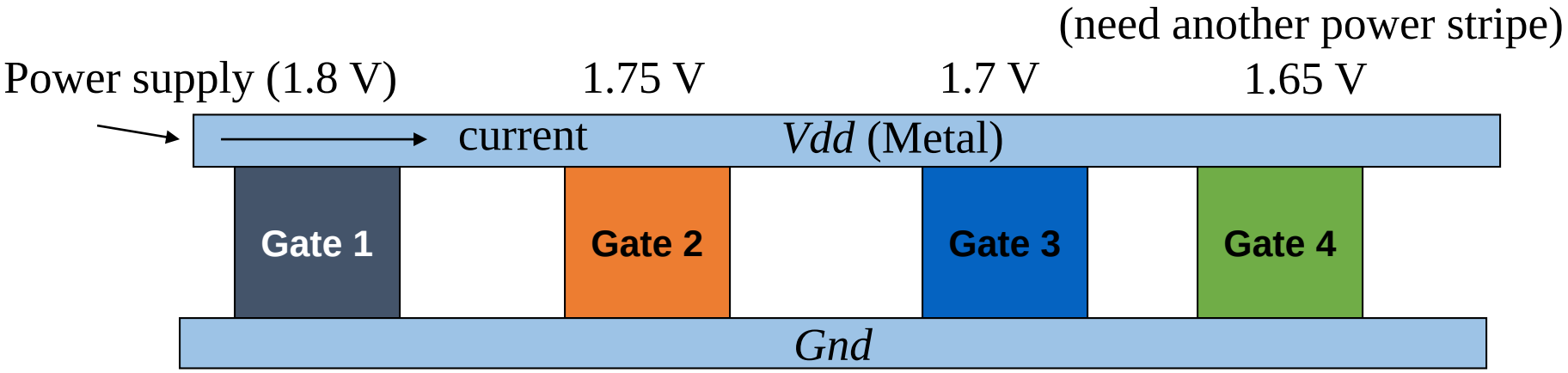
INV: 1ns delay; AND: 2 ns delay



Delay (a→c): 1ns + 2ns = 3ns

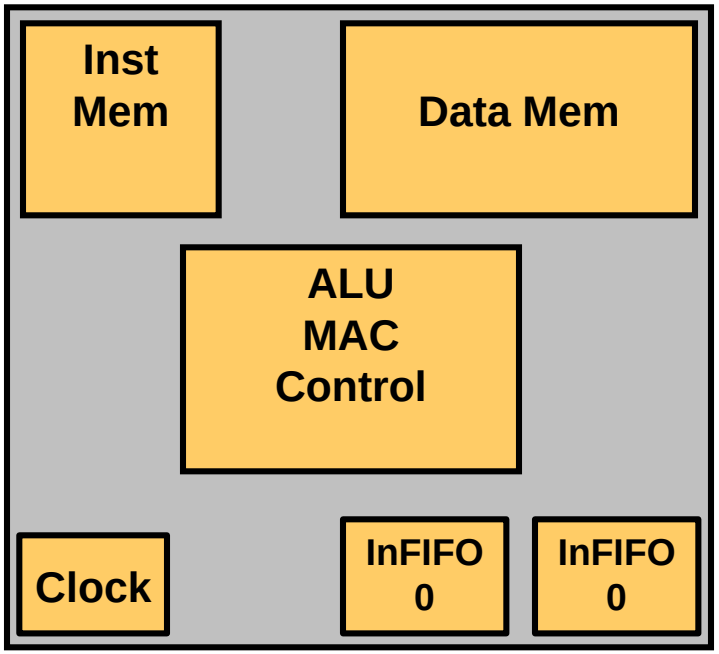
Floorplan

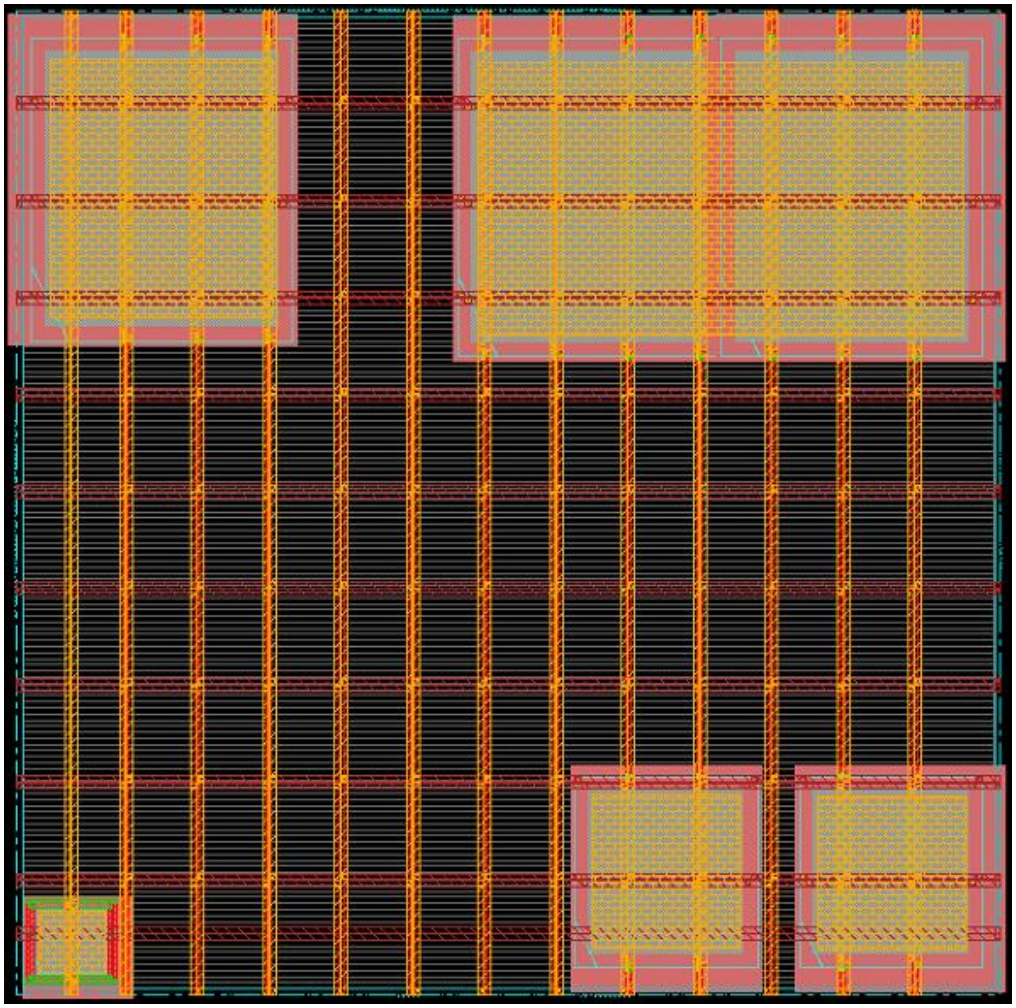
- Size of chip
- Location of Pins
- Location of main blocks
- Power supply: give enough power for each gate



Voltage drop equation: V2 = V1 – I * R

Floorplan of a single processor



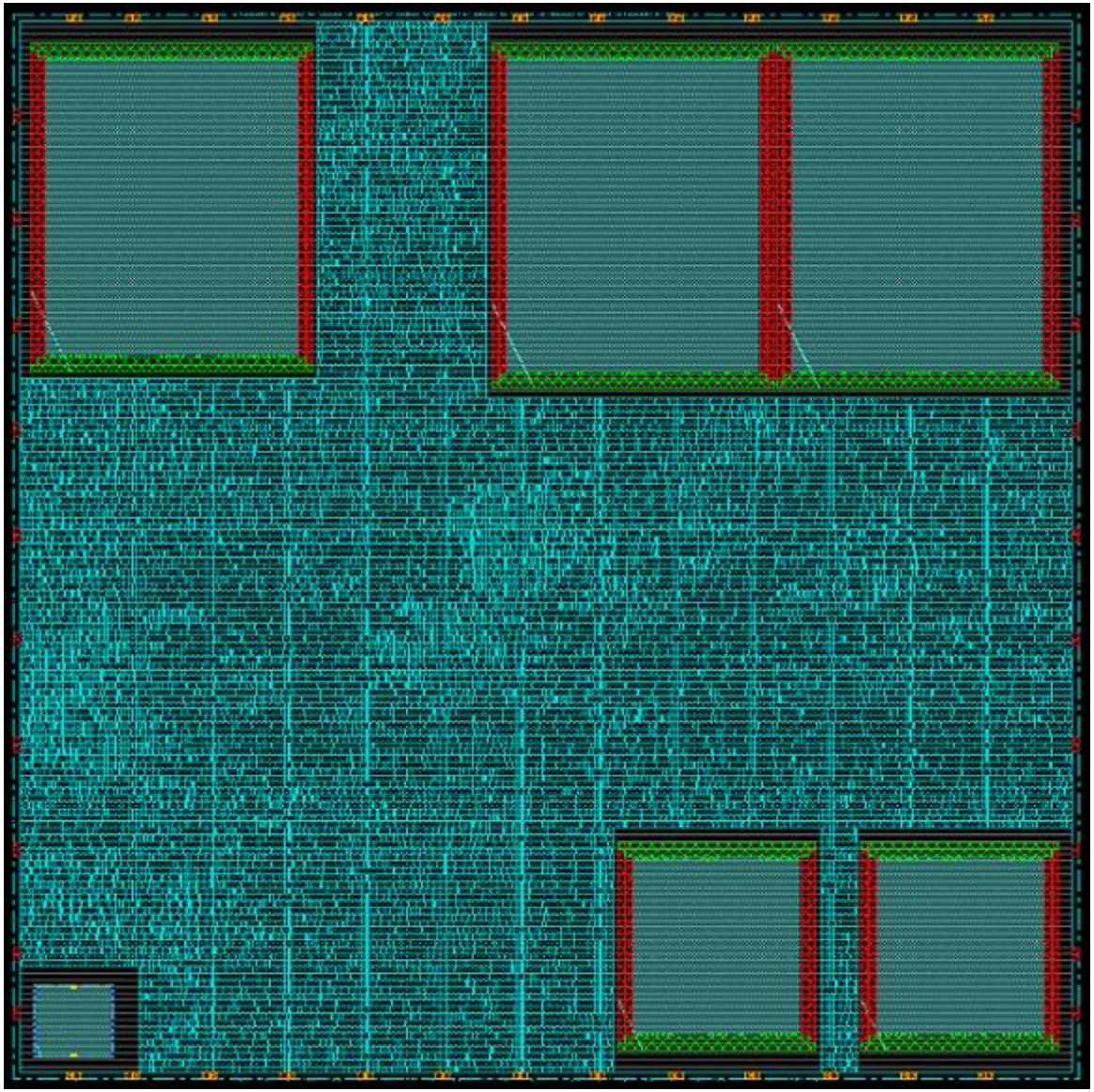


Floorplan of Single KiloCore Processor

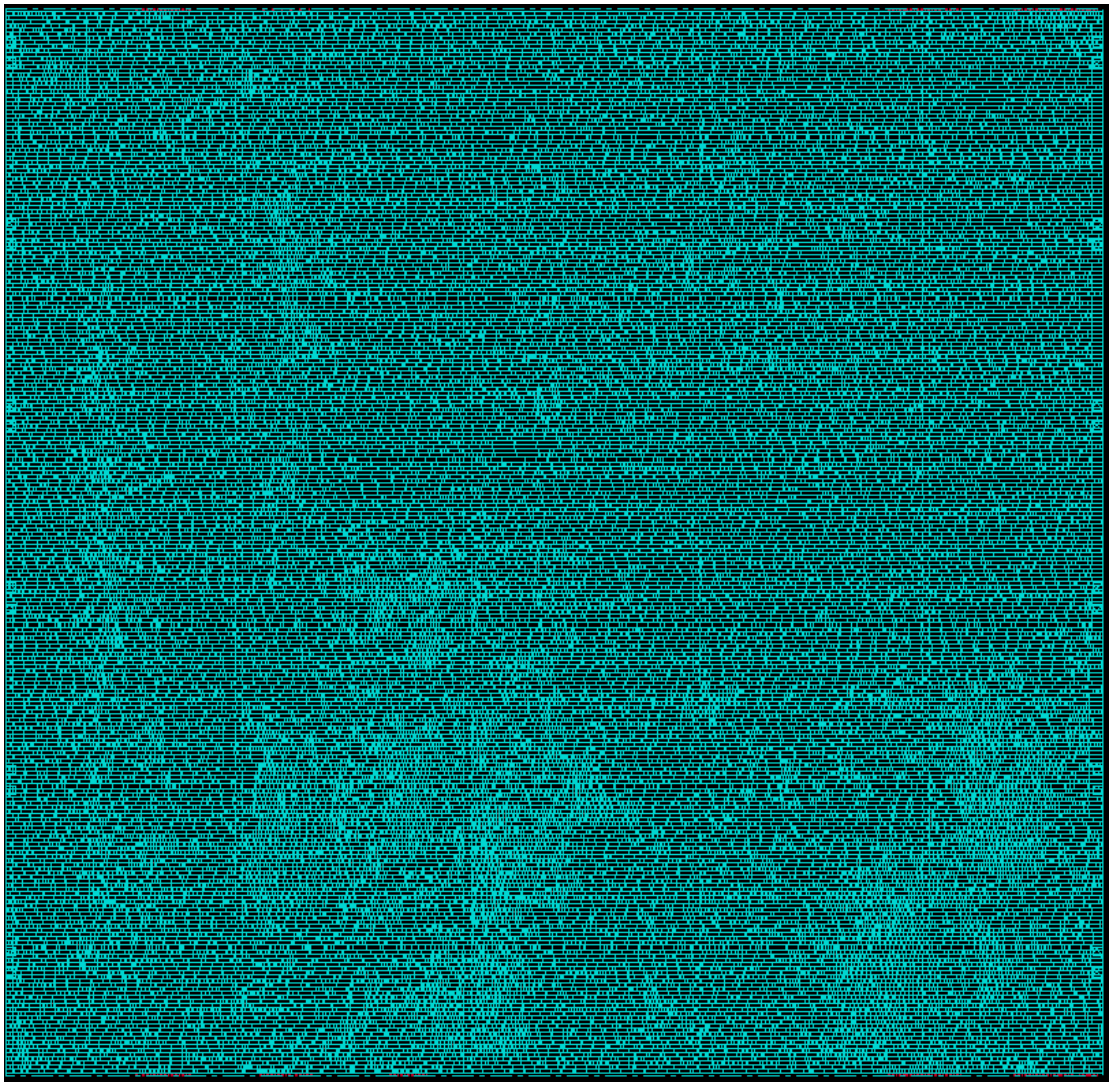
Placement & In-Place Optimization

- Placement: place the gates (standard cells)
  - Utilizes a long series of very complex optimizations to meet all design goals as best as possible. Design goals include: maximum delay of all paths, minimize length of all wires (to increase probability of a successful route), etc.
- In-place optimization
  - Why: there will always be a timing difference between synthesis and layout (e.g., actual wire delay is different than predicted)
  - How: change gate size, insert buffers
  - May not change the circuit function!!

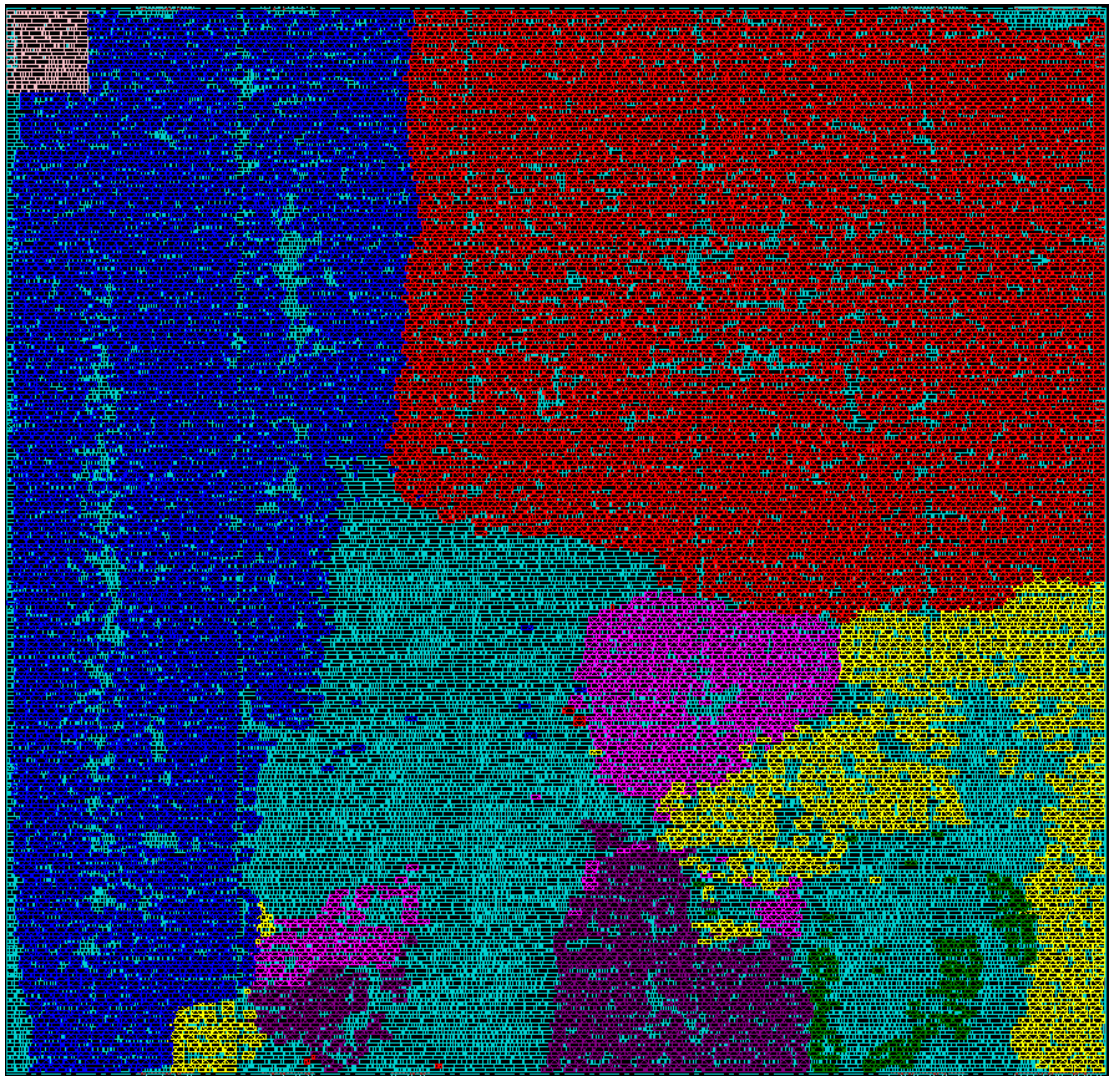
Placement of a single AsAP1 processor

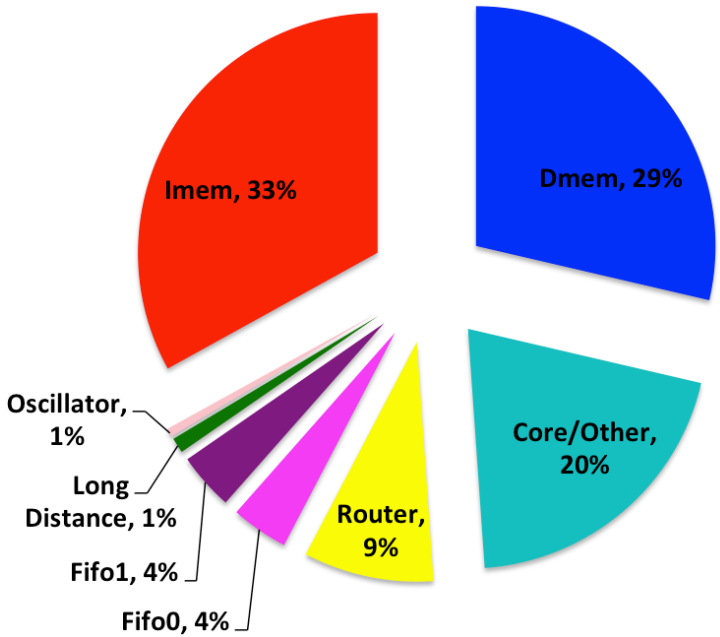


Placement of Single KiloCore Processor



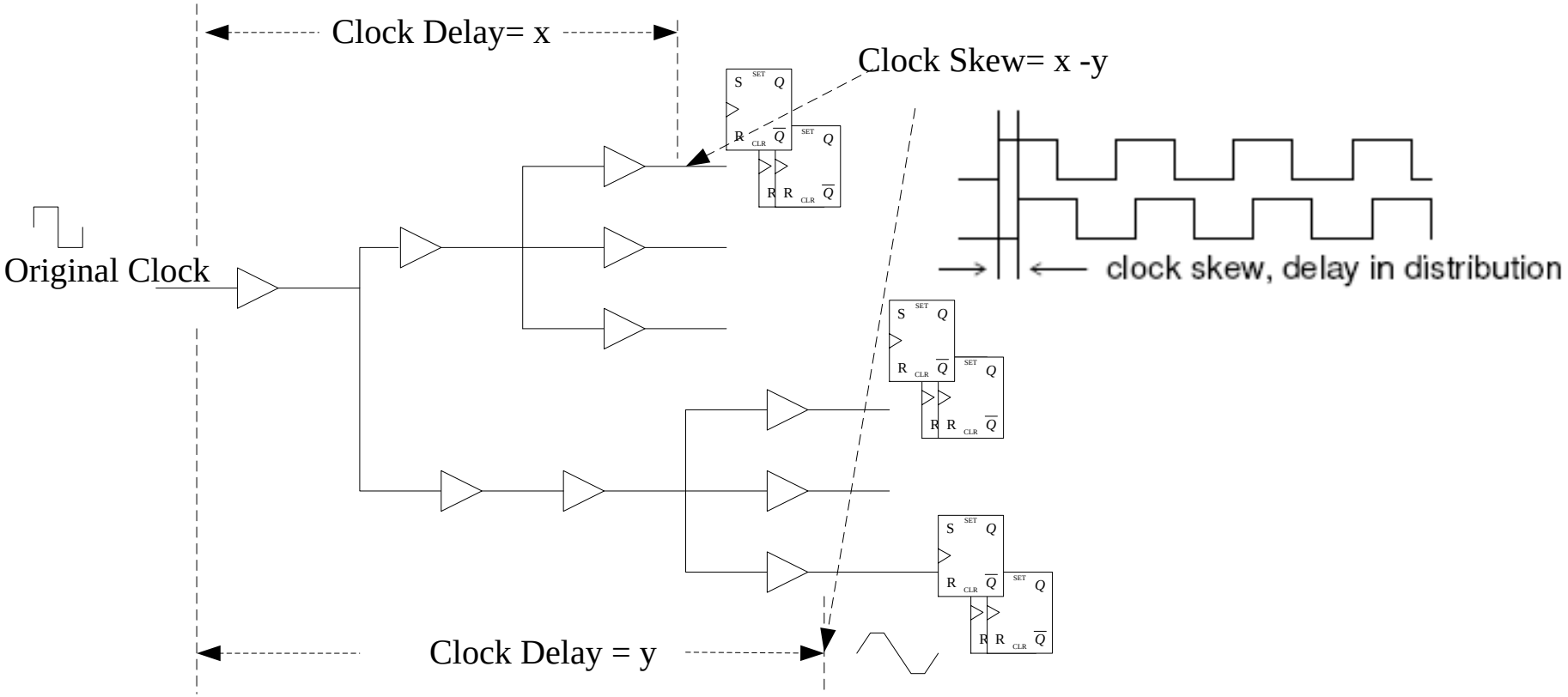
Placement of Single KiloCore Processor



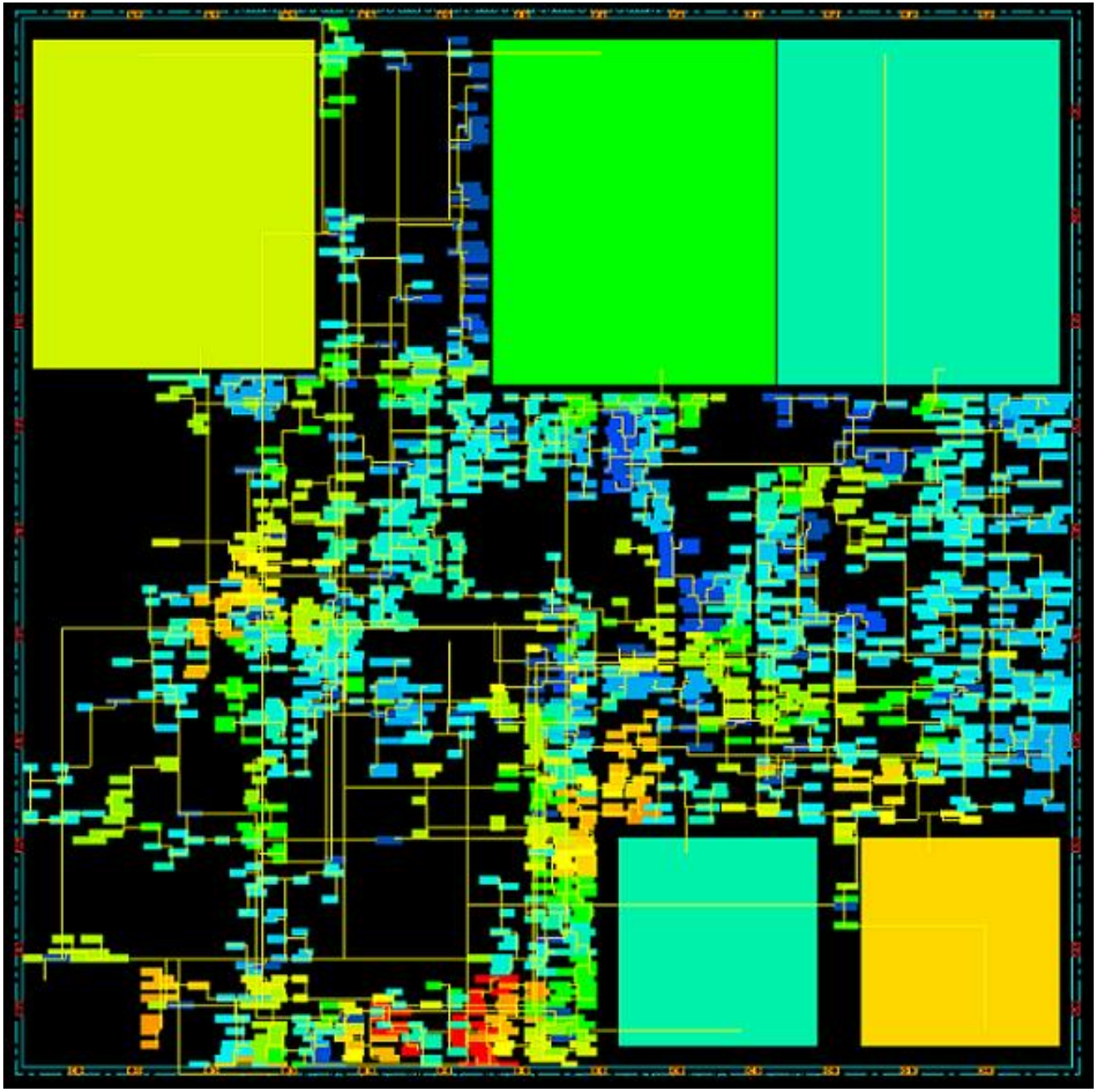


Clock Tree Design

- Main parameters: skew, delay, transition time

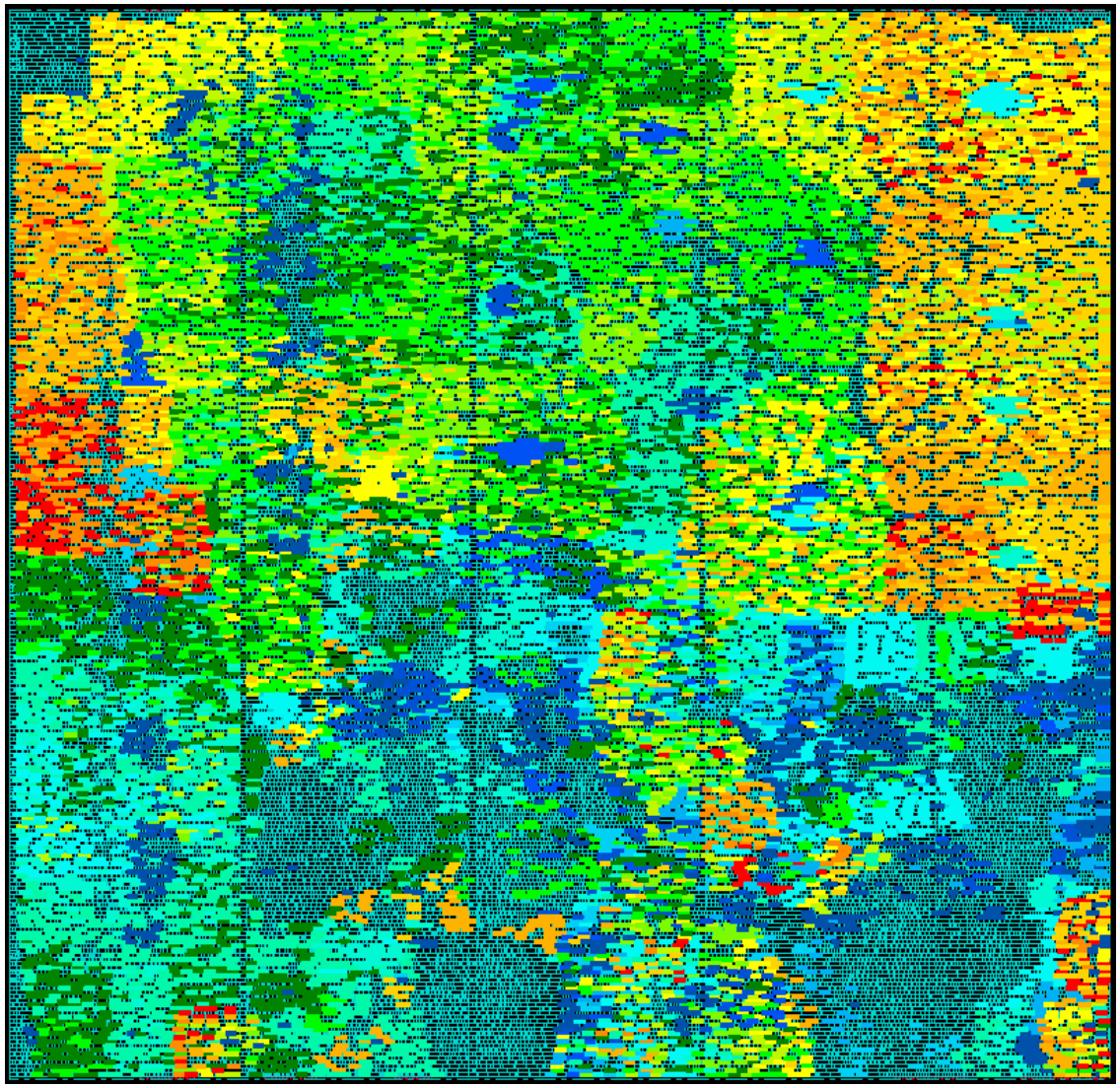


Clock tree of a single AsAP1 processor



Clock Tree of a Single KiloCore Processor

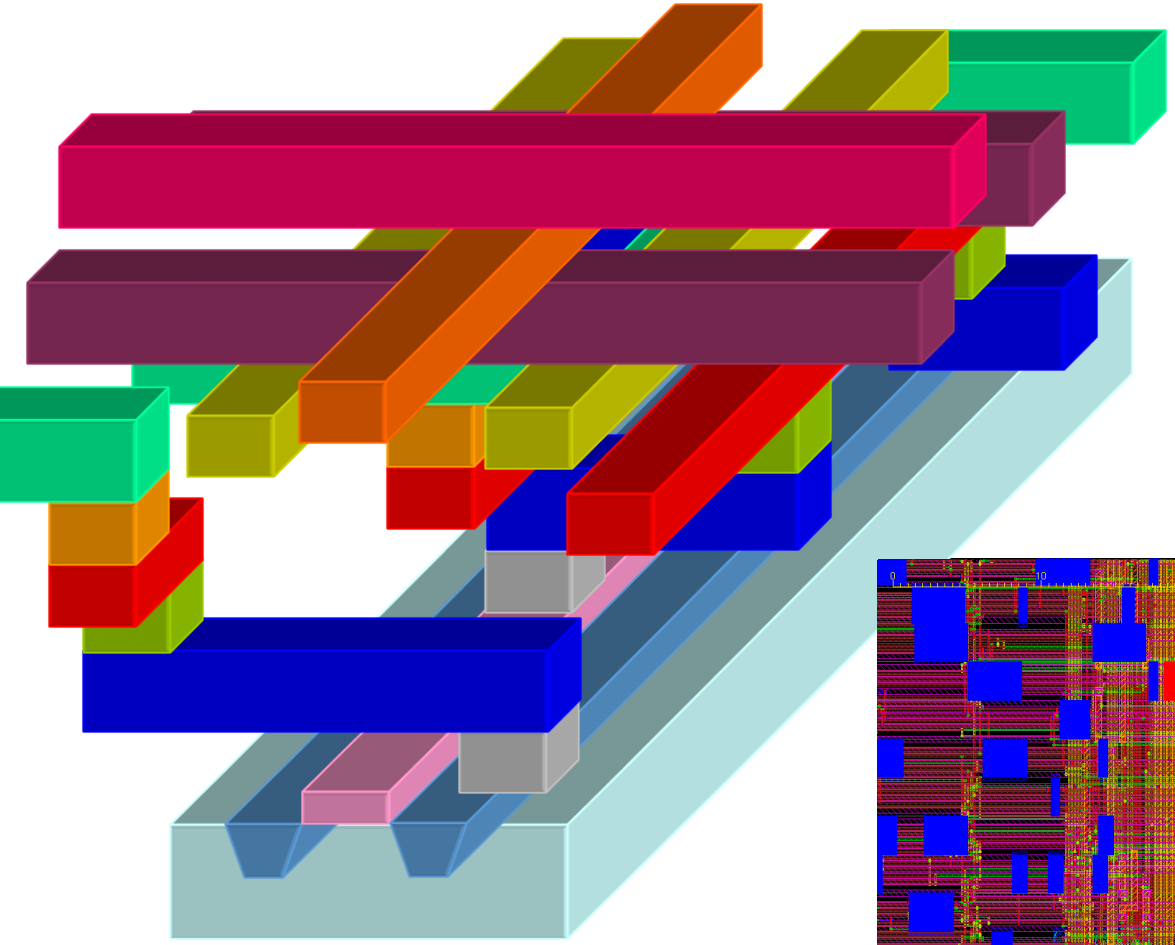
- Colors indicate clock phase which is the same as clock skew



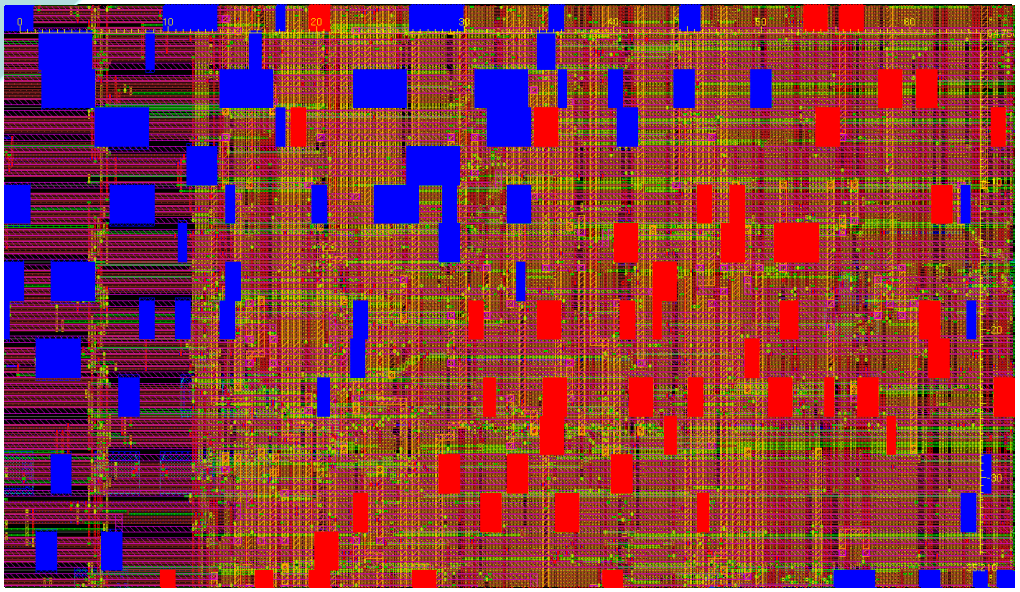
Routing

- Routing is the second step in the "Standard Cell Place & Route" process and consists of the CAD P&R tool routing all necessary signal wires
  - Local power and ground connections to standard cells are made during an earlier power striping step and are made by a much simpler process of simply laying down horizontal power stripes
- Routing consists of two main steps
  - Connection of global signals (power)
  - Connect other signals

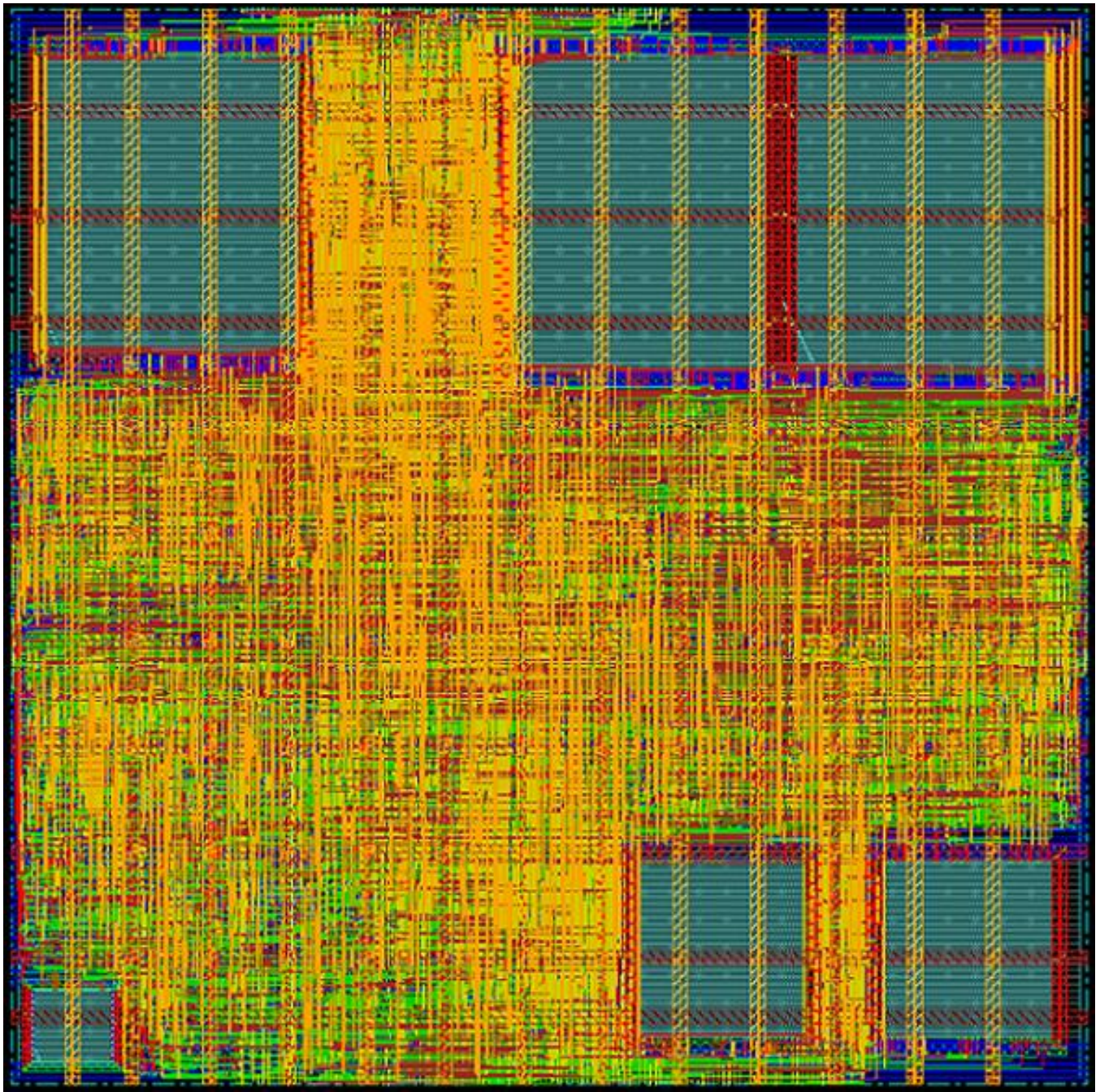
Routing



Metal Layer Topology

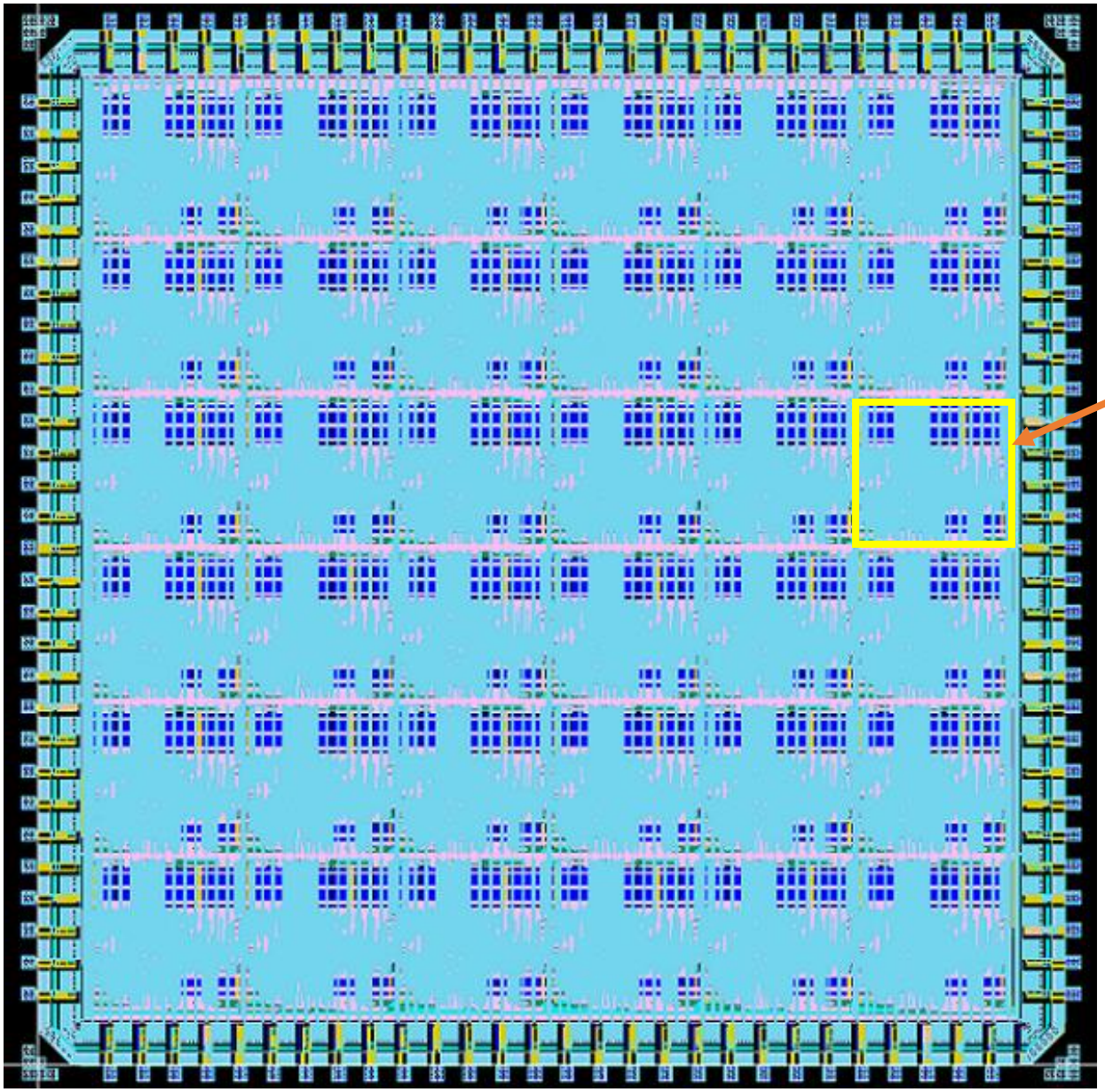


Layout of a Single AsAP1 Processor After Routing



Area:
0.8mm x 0.8mm

Layout of the first generation 6x6 AsAP1

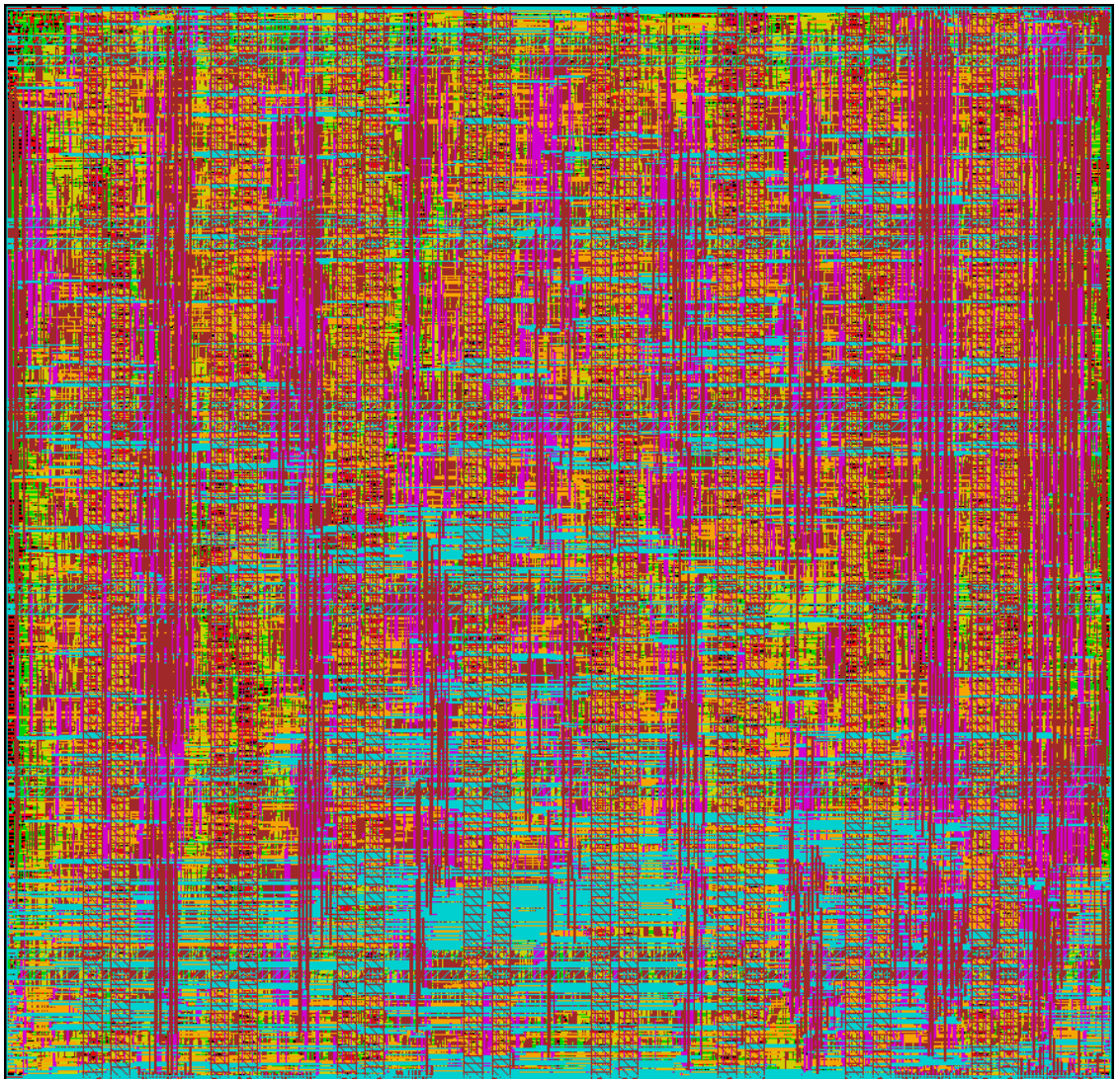


Area: 30 mm^2
in 180 nm CMOS
- 36 processors
- 114 PADs

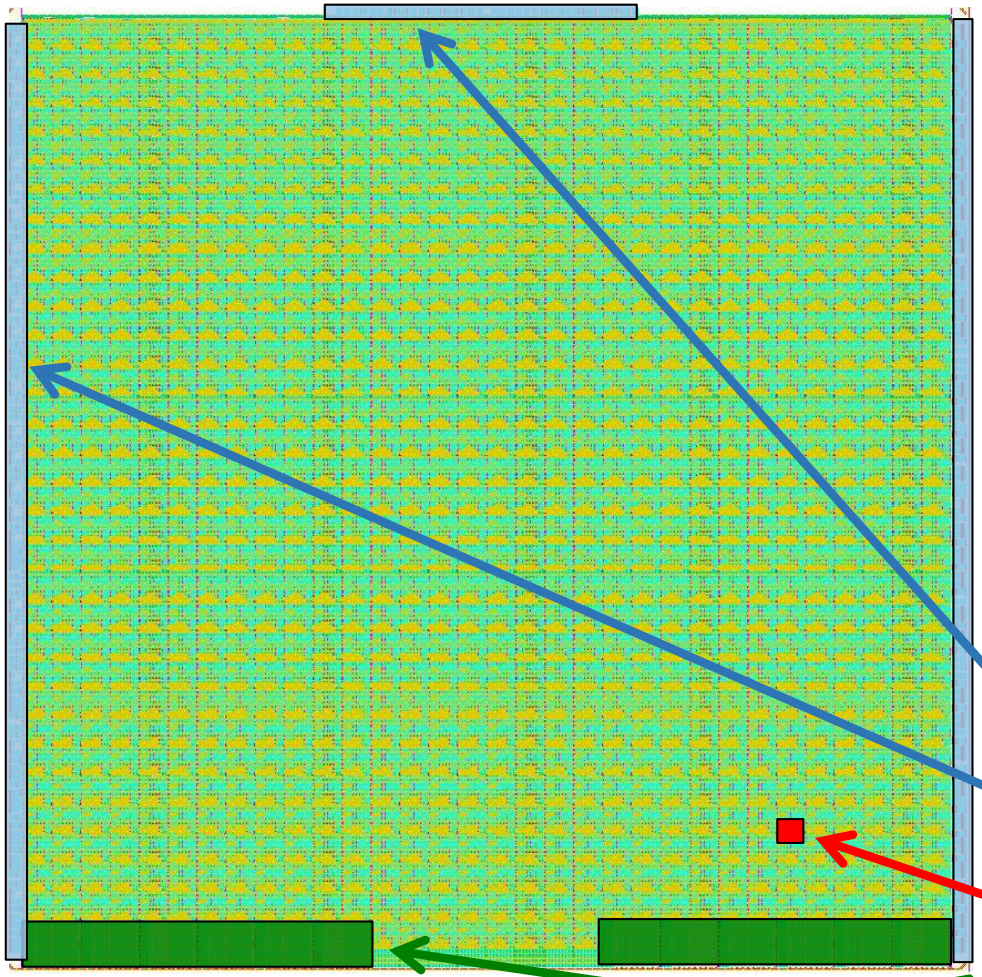
One processor

Routing on a Single KiloCore Processor

- 239 µm x 231.3 µm
- 574,733 transistors



KiloCore Chip Layout



- KiloCore Chip
  - Entire chip takes up 8 mm x 8 mm or 64 mm^2
  - LVDS pairs used for high speed data I/O
  - Drivers connected through C4 bump array

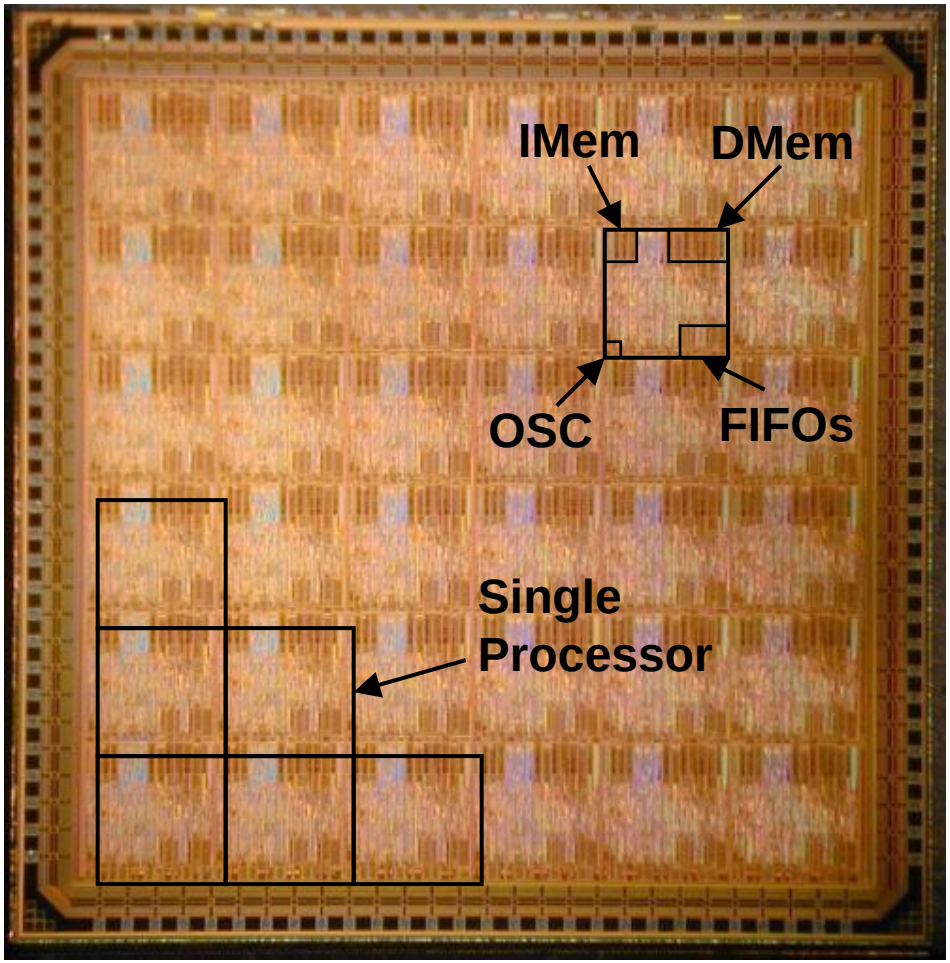
Verification after layout

- DRC (design rule check)
- LVS (layout vs. schematic)
  - GDS-II vs. (verilog + spice module)
- Gate level verilog dynamic simulation
  - Mainly check the function
  - Different with synthesis result

Useful tools

- Dynamic Simulation:
  - Modelsim (Mentor), NC-verilog (Cadence), Active-HDL
- Synthesis:
  - Design-compiler, design-analyzer (Synopsys)
- Placement & Routing
  - Encounter & Virtuoso (Cadence)
  - Astro (Synopsys)
- DRC & LVS
  - Calibre (Mentor)
  - Dracula (Cadence)
- Static Analysis
  - Primetime (Synsopsys)

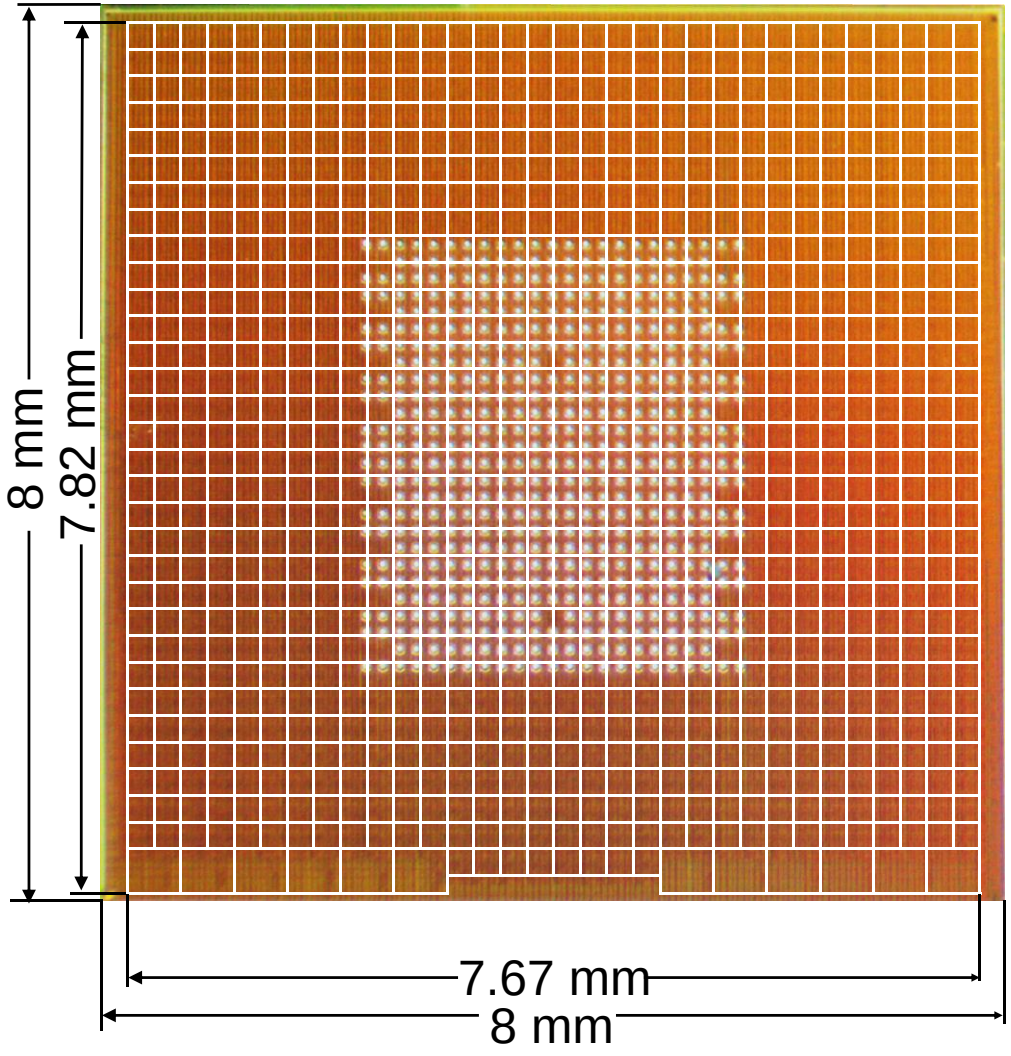
Chip Micrograph of the 36-Core AsAP1



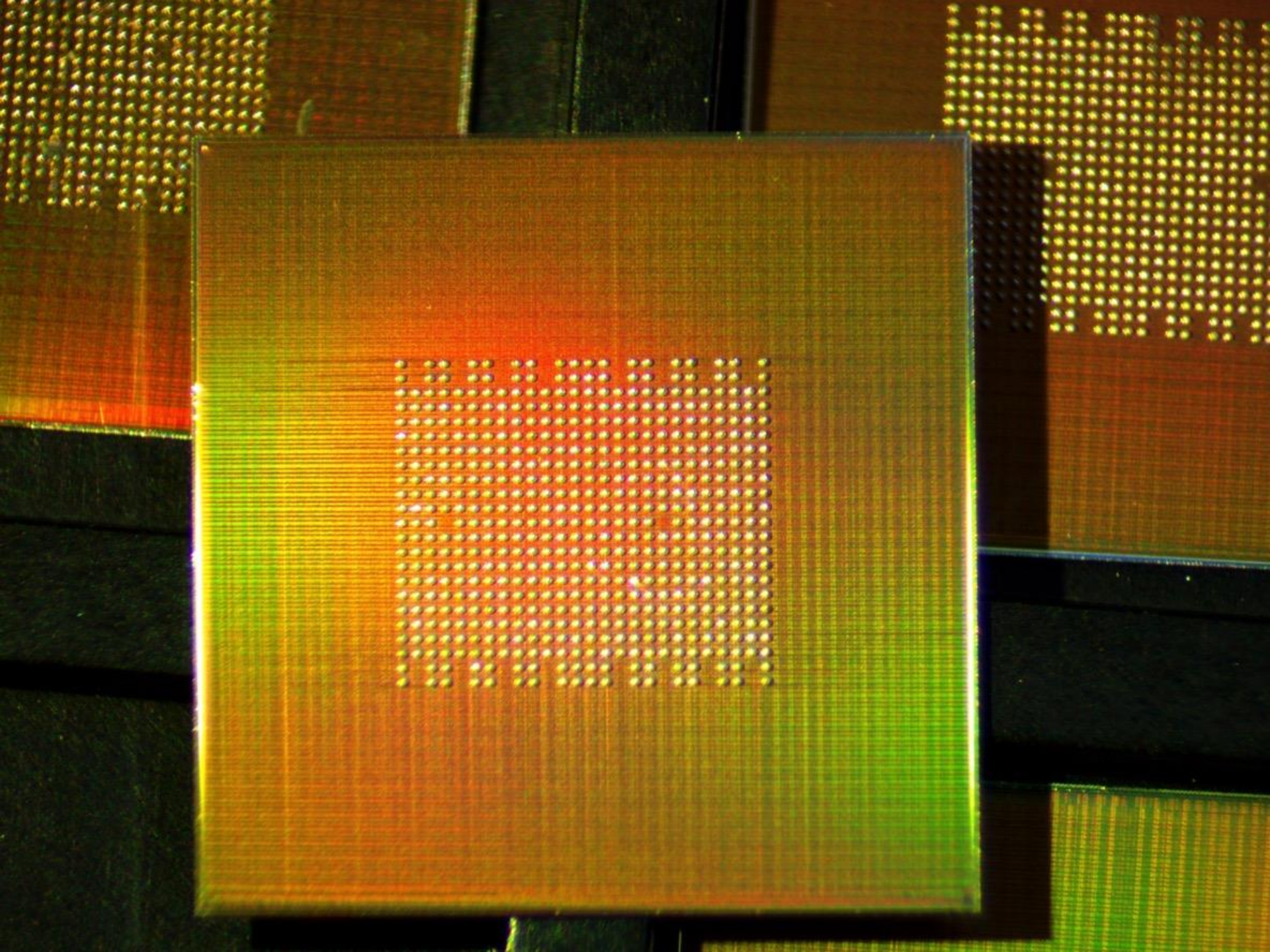
Flow:	Standard-cell based
Technology:	TSMC 0.18 µm
Transistors:	
1 Proc	230,000
Chip	8.5 million
Max speed:	**610 MHz** @ 2.0 V
Area:	
1 Proc	**0.66 mm²**
Chip	32.1 mm²
Power (1 Proc @ 1.8V, 475 MHz):	
Typical application	**32 mW**
Typical 100% active	84 mW
Power (1 Proc @ 0.9V, 116 MHz):	
Typical application	**2.4 mW**

- *ISSCC* 2006
- *HotChips* 2006
- *IEEE Micro* 2007
- *JSSC* 2008

KiloCore Chip



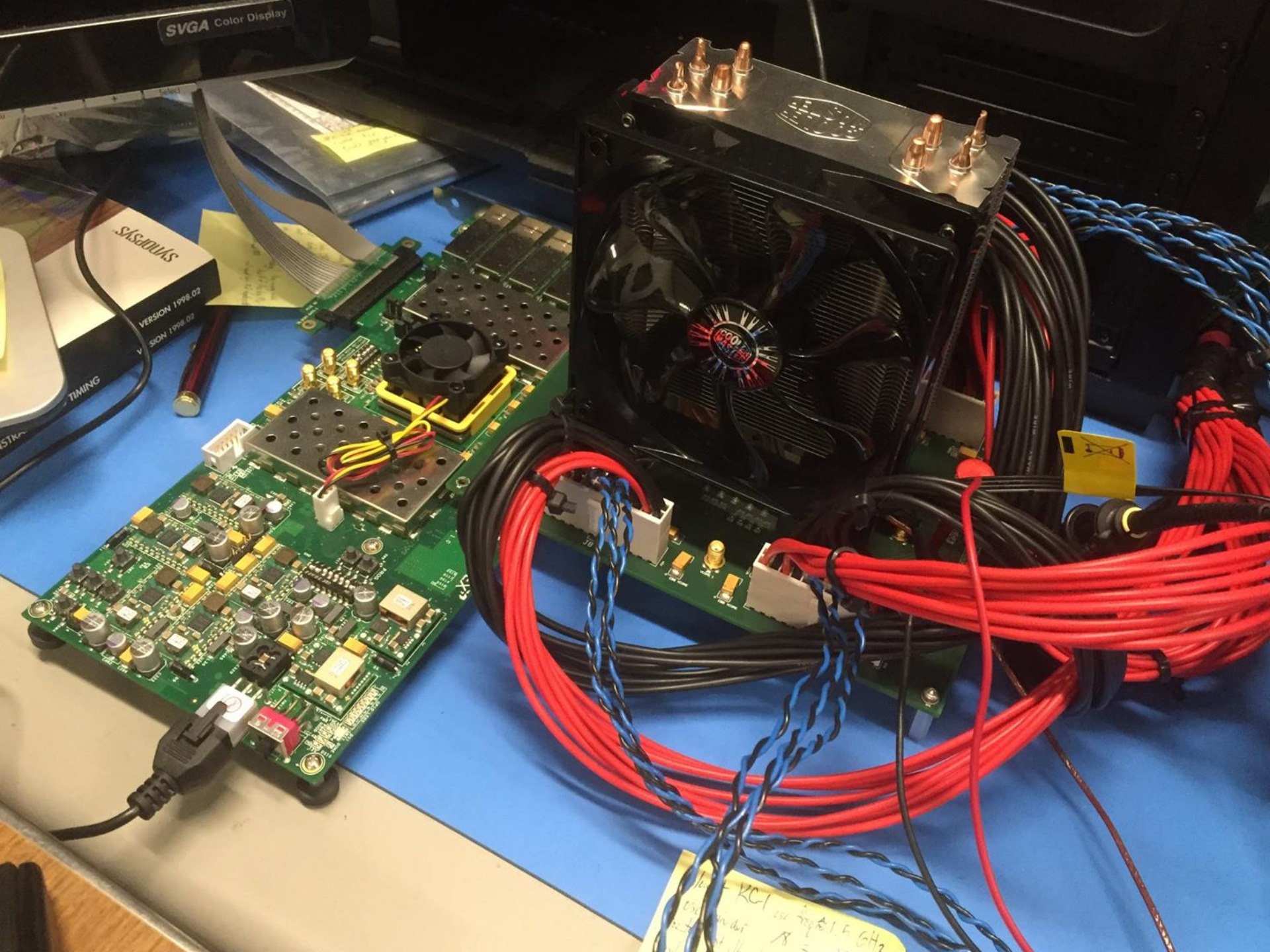
Technology	32nm IBM PDSOI CMOS
Processors	1000 per chip 1.78 GHz @ 1.1 V 1.24 GHz, 18 mW @ 0.90 V 115 MHz, 0.7 mW @ 0.56 V
Indep. Mems	12 per chip
Num. Oscs.	2012
Die Area	64 mm^2
Array Area	60 mm^2
Transistors	621 Million
C4 Bumps	564 (162 I/O)
Package	676 Pad Flip-Chip BGA



KiloCore Measurements

- A maximum of 1.78 trillion instructions per second
  - Assuming a custom package design
- Processors achieve their optimal energy times time of 11.1 (pJ x ns / instruction) at 0.9 V
- At minimum voltage, KiloCore can perform 115 billion instructions per second using 0.7 Watts—low enough to be powered by a single AA battery!

Supply Voltage	Max Clock Freq (MHz)	Total Chip Instructions / sec	Total Chip Power
1.10 V	1782	**1.78 Trillion**	39 Watts
0.90 V	1237	1.24 Trillion	17 Watts
0.84 V	1000	**1.00 Trillion**	**13 Watts**
0.75 V	638	638 Billion	6.3 Watts
0.56 V	115	**115 Billion**	**0.7 Watts**


SVGA Color Display
SYNOPSYS
VERSION 1998.02
TIMING