Jaguar: A Next-Generation Low-Power x86-64 Core

Teja Singh, Joshua Bell, Shane Southard, Deepesh John

AMD, Austin TX

Outline

- Motivation
- Architecture
- Technology
- Implementation
- Circuits
- Clocking
- Timing
- Power
- Reliability
- Conclusion

Motivation

Long Battery Life, Quad-Core Performance and Rich Entertainment Features

Rich Entertainment, Smart Price	Responsive Performance	Amazing Battery Life

- First AMD 28nm quad-core x86-64
- Build unit to deploy into a wide variety of SoCs for different applications
- Span wide array of applications from sub 5W to 25W
- Worthy successor to "Bobcat" x86-64 core

Target Markets

- ## Build SoC to fit range of markets
  - Tablet, hybrids
  - Value notebook
  - Ultrathin notebook
  - Value desktop

Core Comparison

	"Bobcat" (BT)	"Jaguar" (JG)
Process	40nm bulk	28nm bulk
# Cores	2	4
L2 Cache Size	1MB (512KB dedicated 16-way)	2MB (shared, 4x 512KB 16-way)
Core Size	4.9mm^2	3.1mm^2
Core Flop Count	159900	194490
Machine Width	2-wide	2-wide
Physical Address	36-bit	40-bit
L1 Instruction Cache	32kB, 2-way 64B line	32kB, 2-way 64B line
L1 Data Cache	32KB, 8-way 64B line	32KB, 8-way 64B line
Load/Store Bandwidth	8B/cycle, Write Combine	16B/cycle, Write Combine
FPU Datapath	64-bit	128-bit
EX Scheduler	16 entries	20 entries
AGU Scheduler	8 entries	12 entries

Architecture

- ISA enhancements added
  - SSE4.1, SSE4.2
  - Advanced Vector Extensions
  - AES, CLMUL
  - MOVBE
  - XSAVE/XSAVEOPT
  - F16C, BMI1
- 4x32B Instruction Cache loop buffer for power
- Improved Instruction Cache prefetcher for IPC
- Added hardware integer divider
- L2 prefetcher
- Improved C6 and CC6 entry/exit latencies
- Estimated typical IPC improvement over "Bobcat": >15%*
- Clock gate >92% flops in typical applications

* Estimates based on internal AMD modeling using benchmark simulations. This information is preliminary and subject to change without notice.

Technology

- TSMC 28nm bulk HKMG
- 3 Vt solution:
  HVT/RVT/LVT
- Longer lengths for each Vt
- BT had 10 metal stack
- JG uses 11 metal stack
  - stdcells block most of M2
  - additional 2x layer
    added to offset loss of tracks

Layer	BT Type	BT Pitch	JG Type	JG Pitch
M1	1x	126nm	1x	90nm
M2-M8	1x	126nm	1x	90nm
M9	14x	900nm	2x	180nm
M10	14x	900nm	10x	900nm
M11	n/a	n/a	10x	900nm

* Reference: Wuu, Shien-Yan, et al.. 2009 Symposium on VLSI Technology Digest. pp 210-211
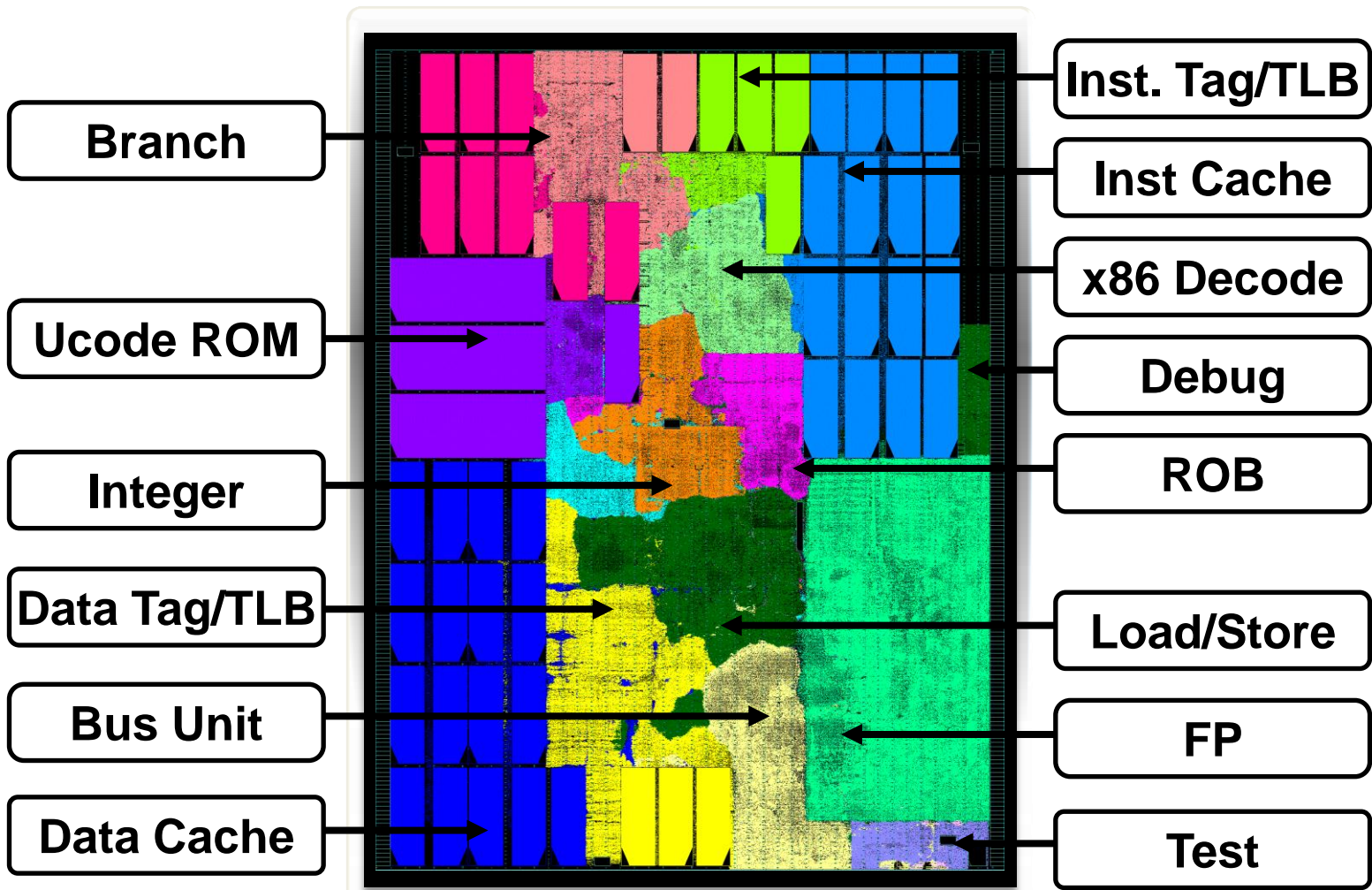
Implementation Overview

- Focus on density
    - Use high density 9 track library
    - Use 1x metals to increase routing resources
    - Implemented using large units to reduce boundary cases
        - Core is 1.25 million placed instances
        - L2I is 0.6 million placed instances
- Standard auto place and route design style
- JG Core has 2 unique custom arrays
- Achieved silicon frequency >1.85Ghz*
- Integrated Power Gating
- Power supply via towers oriented based on route congestion

* Estimates based on internal AMD modeling using benchmark simulations.This information is preliminary and subject to change without notice.
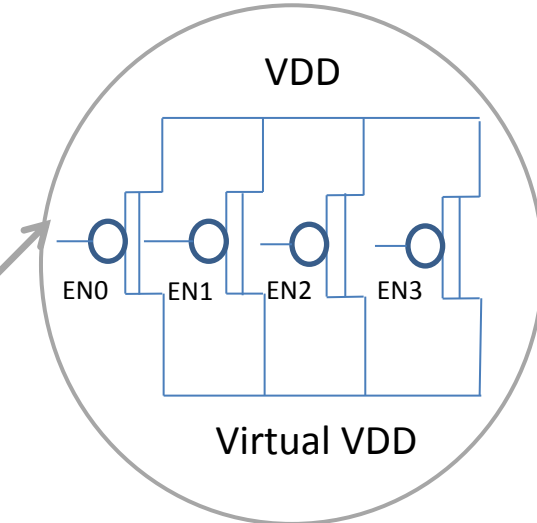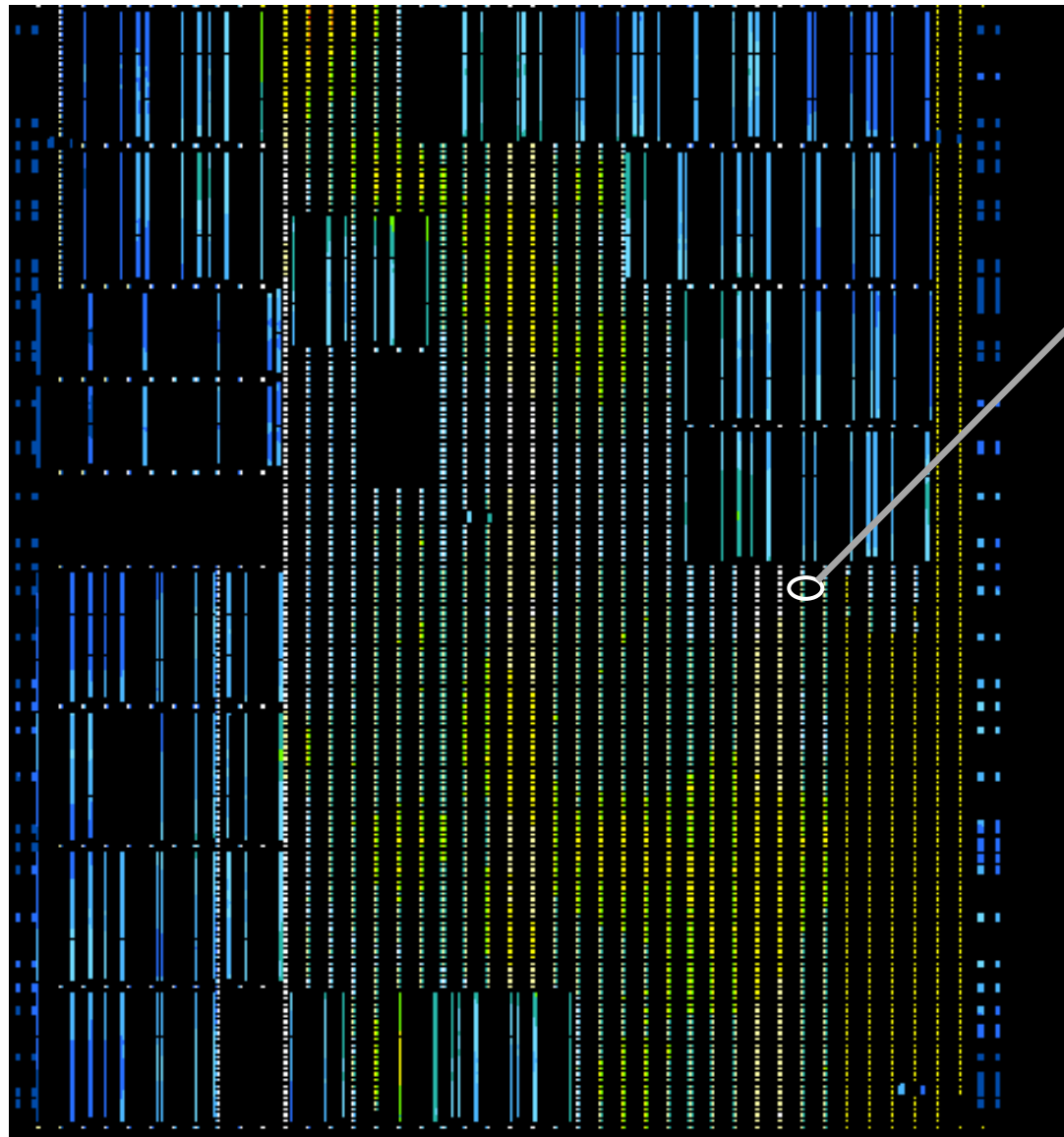
# Compute Unit Floorplan

Core Floorplan

Branch

Ucode ROM

Integer

Data Tag/TLB

Bus Unit

Data Cache

Inst. Tag/TLB

Inst Cache

x86 Decode

Debug

ROB

Load/Store

FP

Test

Core Power Gating

VDD

EN0 EN1 EN2 EN3

Virtual VDD
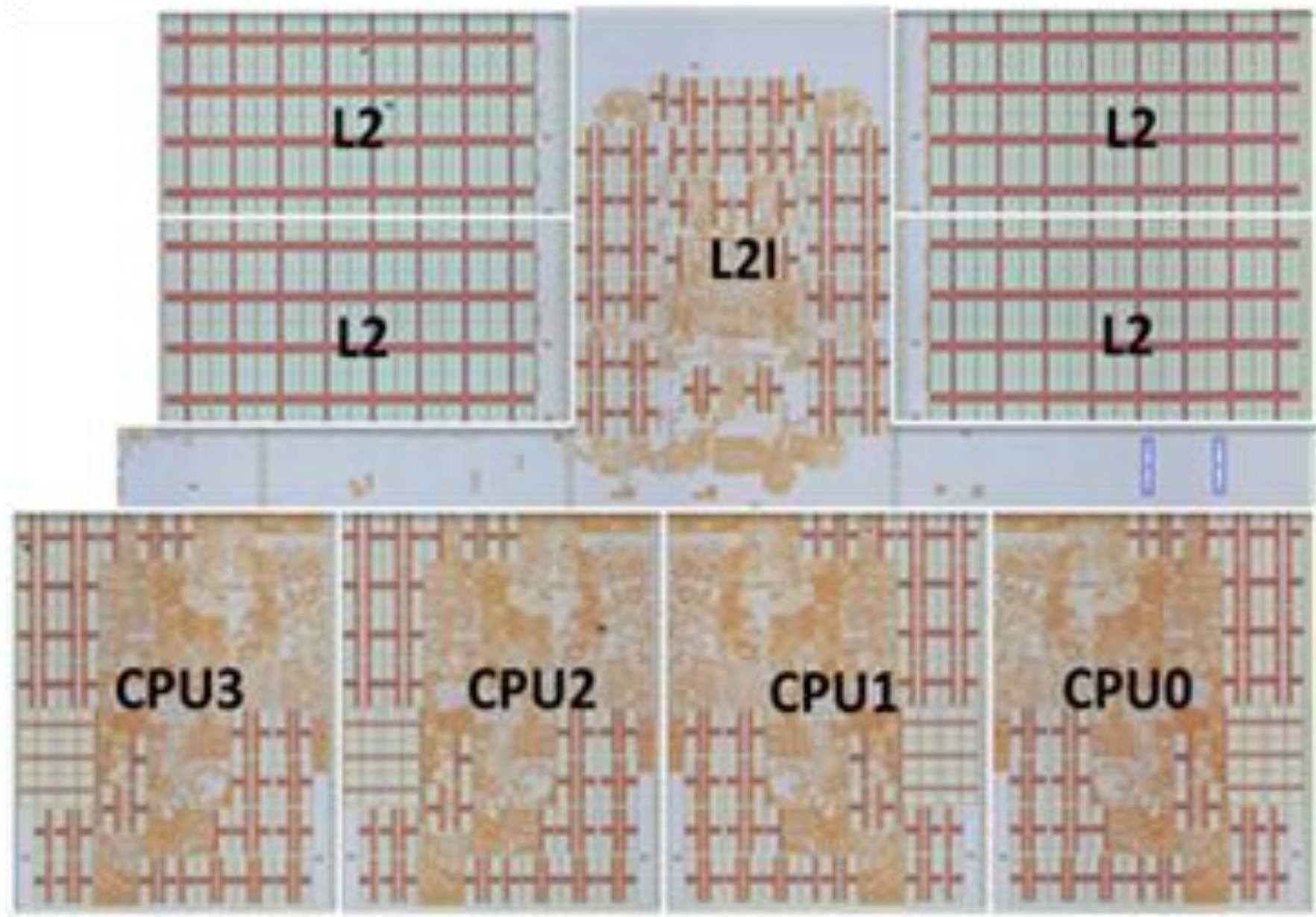
- Headers have 4 independent enables to control strength: {1/15, 2/15, 4/15, 8/15} of total width
- Diagram showing highlighted headers within the JG core
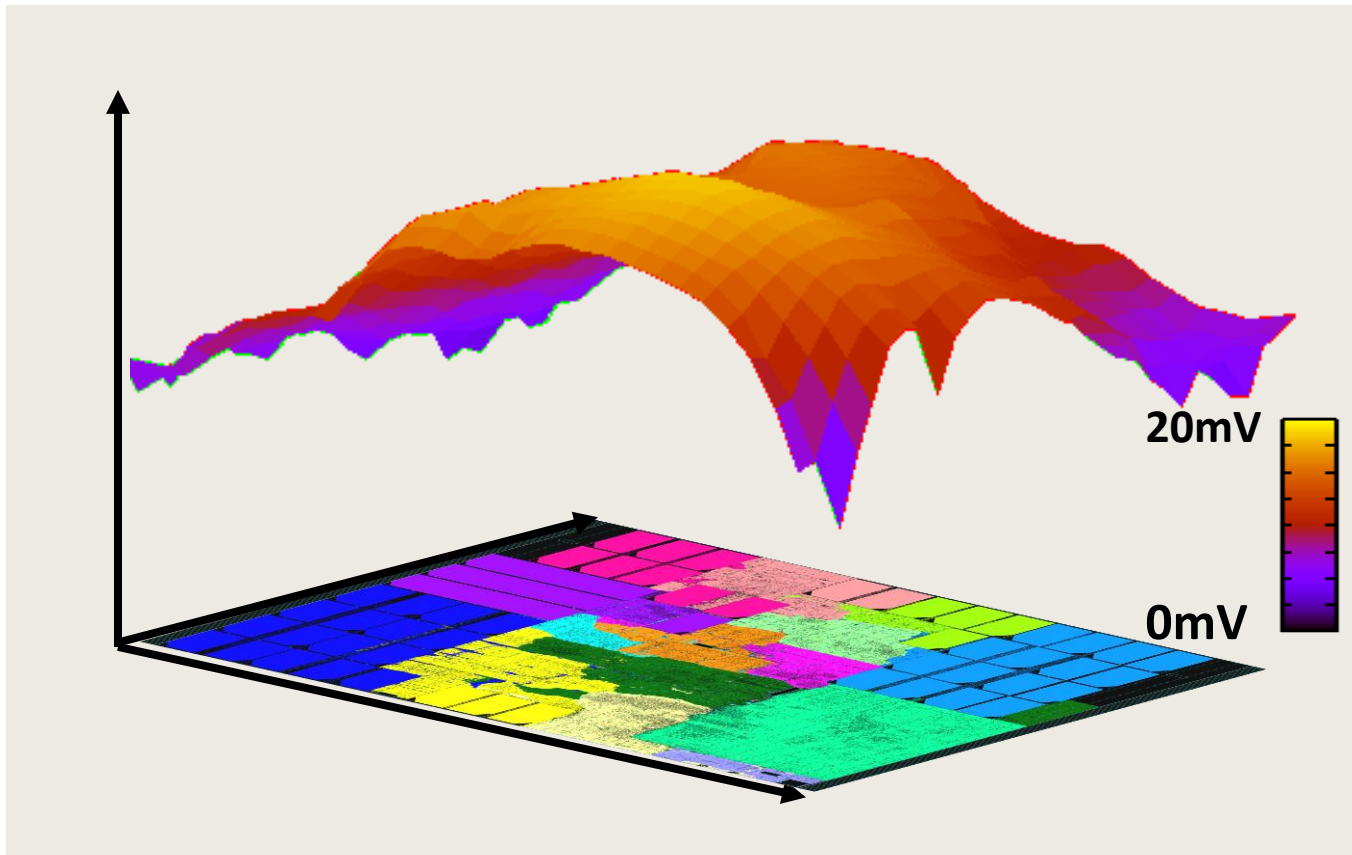- Area overhead is ~3%

Custom Array Power Gating
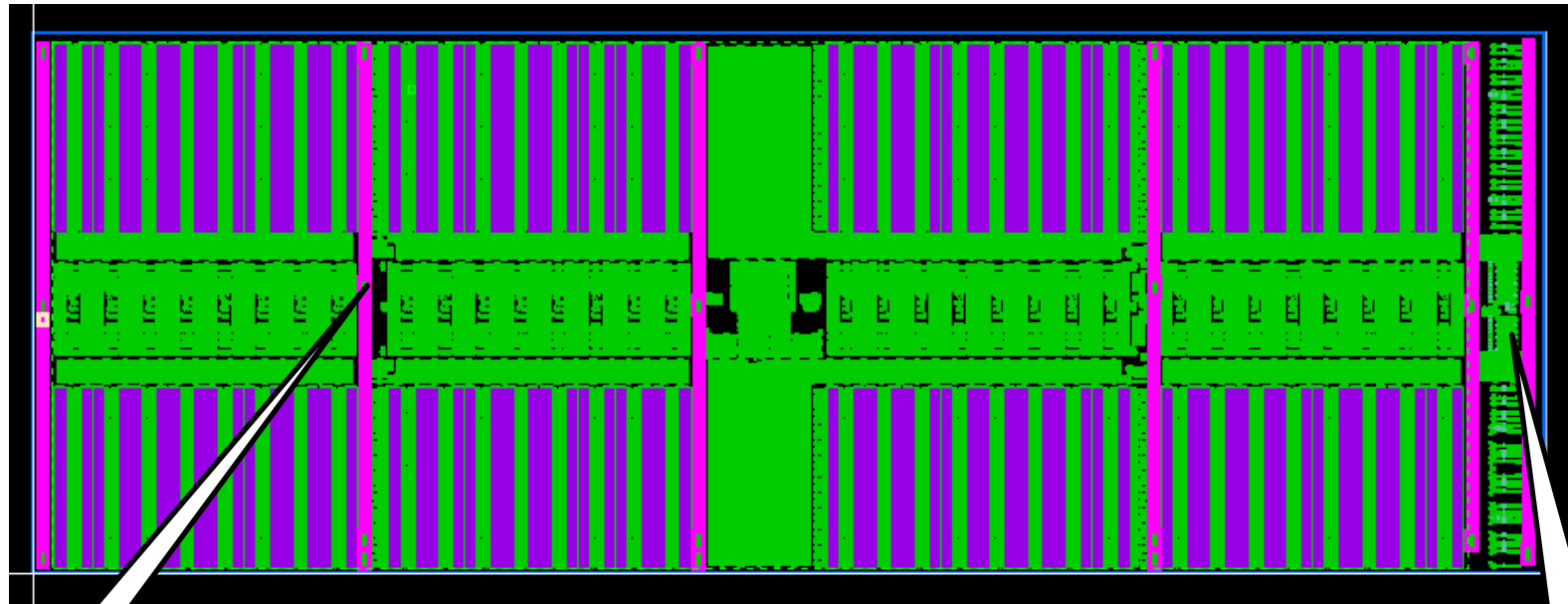
Power gater column

Control Section

- Arrays have integrated power header columns
- Control section with more columns to reduce IR drop
- ROM array shown above ~390k bits

Core Power Gating

- Contour IR map of power headers on the JG core
- Showing worst case pattern during a dynamic IR analysis
- Header IR drop is <20mV*; total IR drop within design limits

* Estimates based on internal AMD modeling using benchmark simulations. This information is preliminary and subject to change without notice.
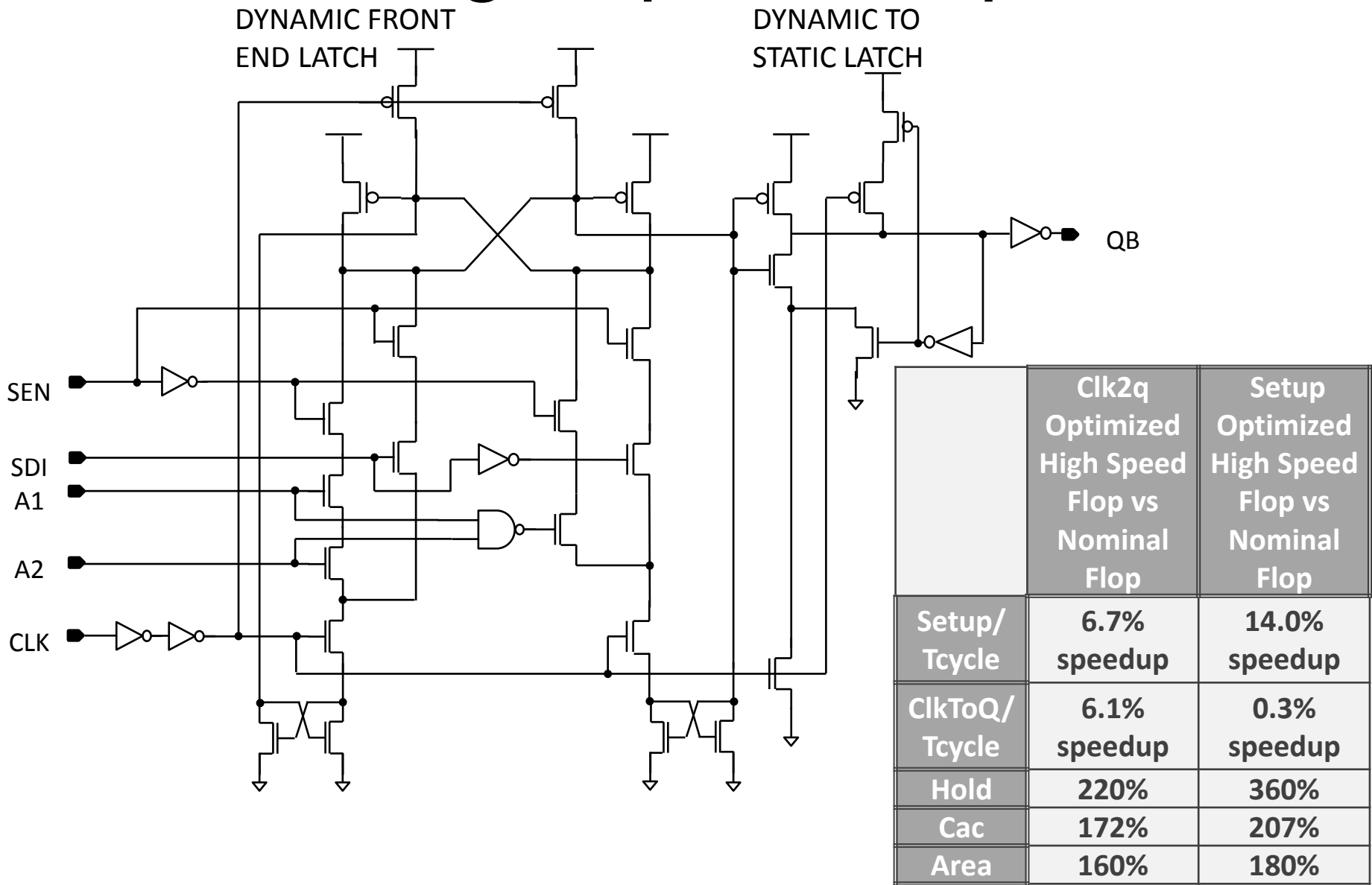
Compute Unit IR Map
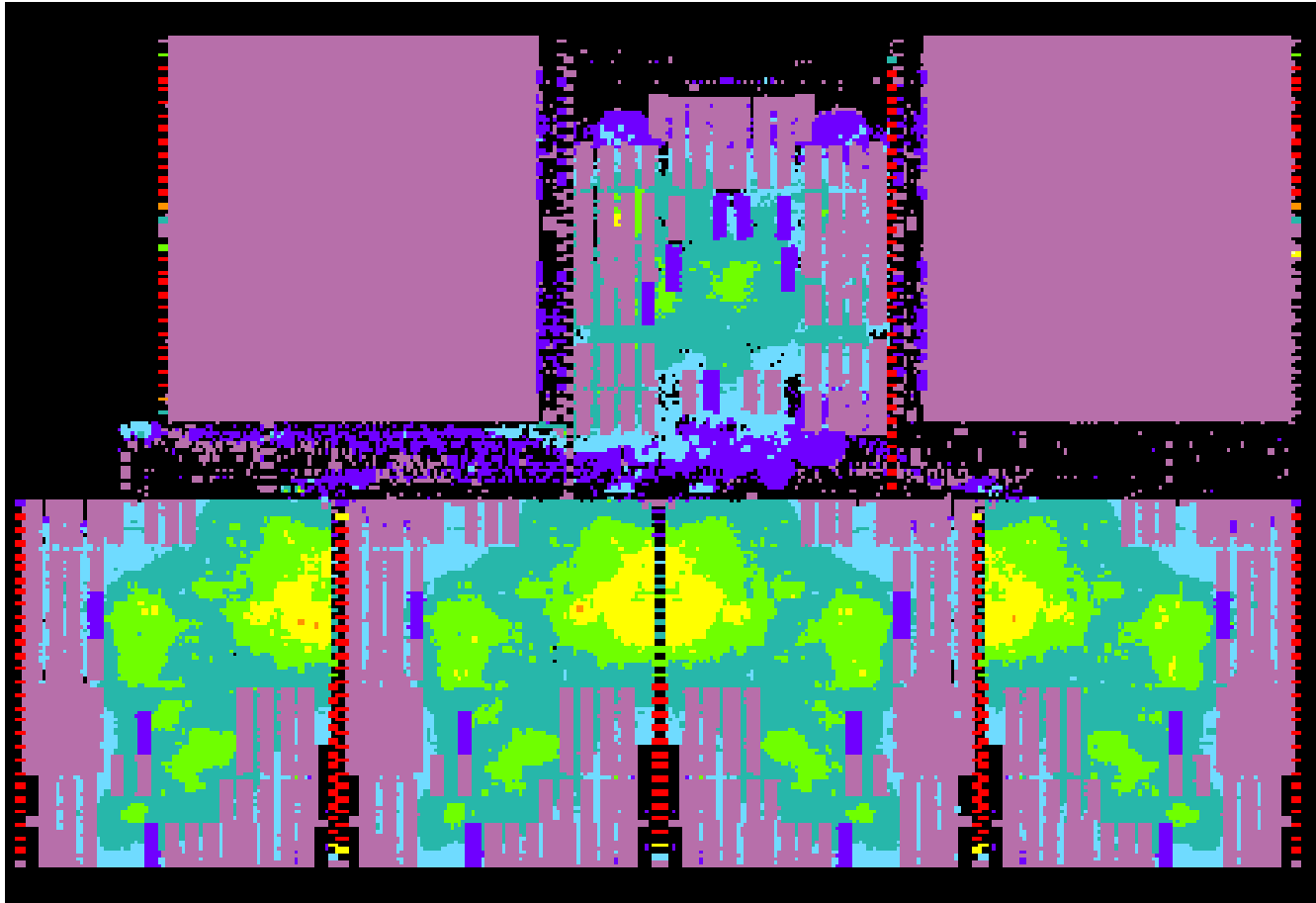
- IR map using a worst case pattern highlighting areas with larger drops
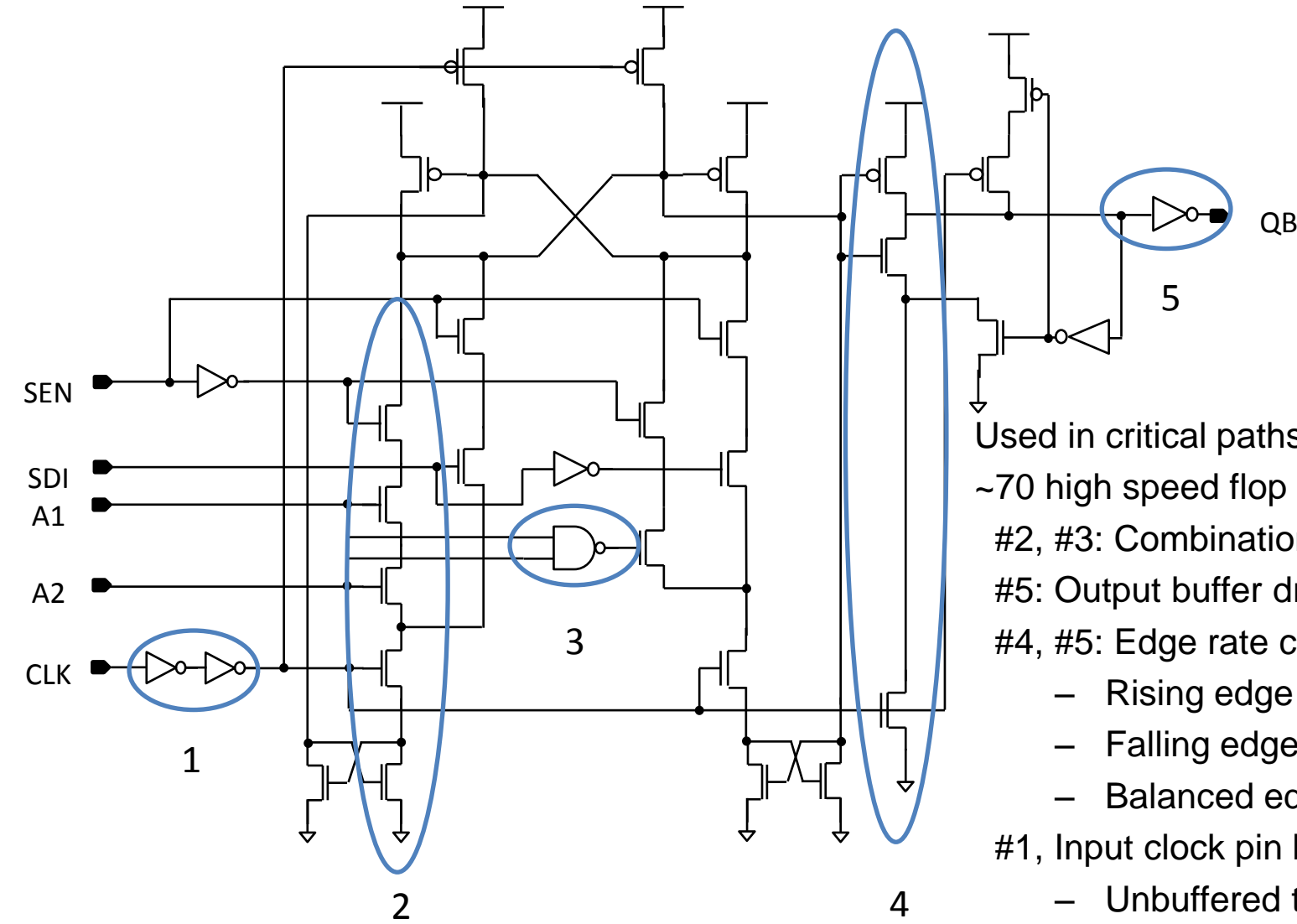- Showing worst case pattern during a dynamic IR analysis

Circuit Overview

- Reduce custom array count from BT
  - RAM array module
  - ROM array module
- Focus on process portability
- Used high speed flops in top critical timing paths
- Arrays utilize fuse programmability for flexibility and reuse

High Speed Flop

DYNAMIC FRONT
END LATCH

DYNAMIC TO
STATIC LATCH

|	Clk2q Optimized High Speed Flop vs Nominal Flop	Setup Optimized High Speed Flop vs Nominal Flop
Setup/ Tcycle	6.7% speedup	14.0% speedup
ClkToQ/ Tcycle	6.1% speedup	0.3% speedup
Hold	220%	360%
Cac	172%	207%
Area	160%	180%

High Speed Flop

Used in critical paths

~70 high speed flop variants based on

#2, #3: Combinational function

#5: Output buffer drive strength

#4, #5: Edge rate control on outputs

– Rising edge skewed

– Falling edge skewed

– Balanced edges

#1, Input clock pin buffering

– Unbuffered to favor Clk2Q

– Buffered to favor setup

RAM Fuse Capabilities
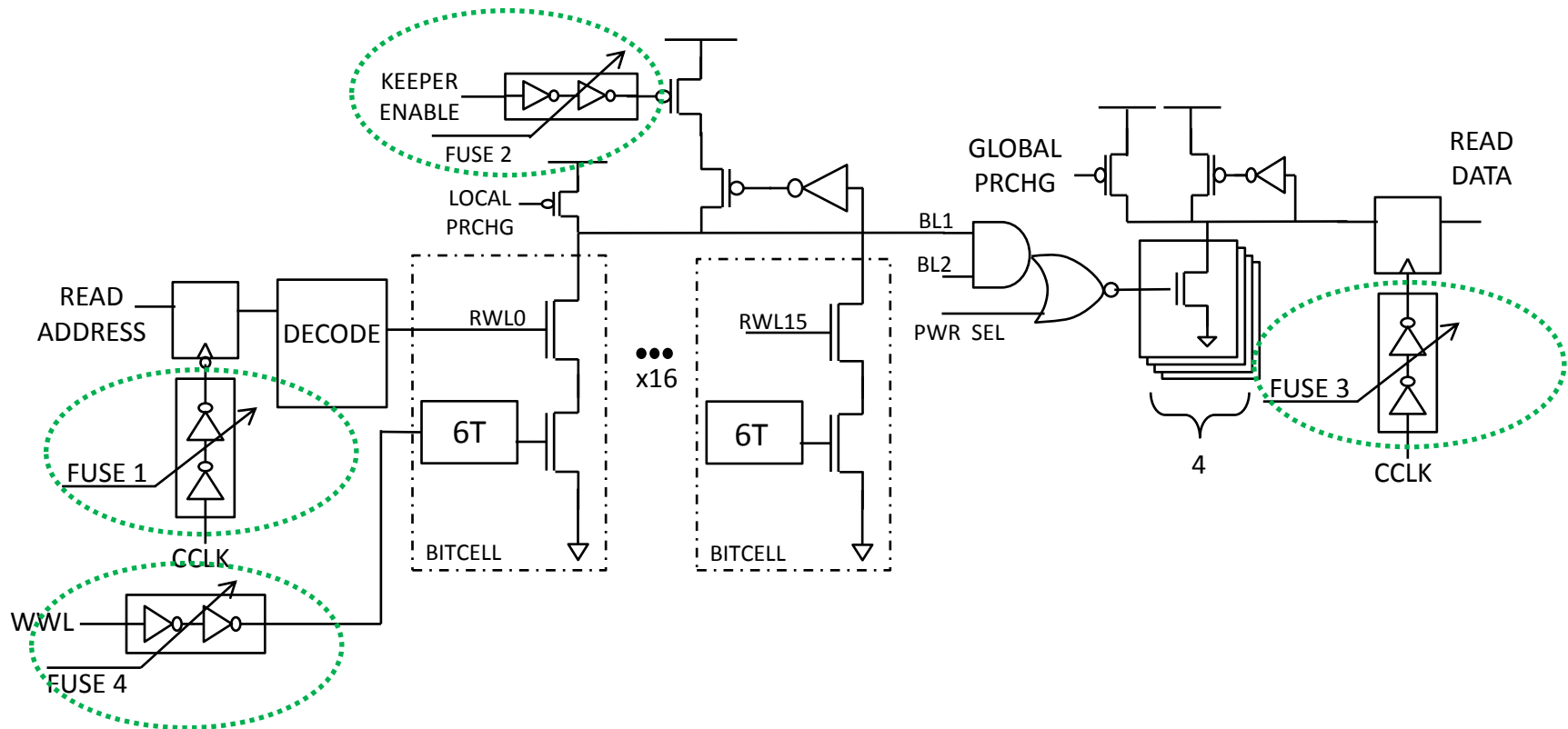
- RAM array reuse was a goal; 51 instantiations within the JG Core, 276 instantiations within the Compute Unit
- Utilize fuse capabilities to tune the design

Array Read Timing Fuses
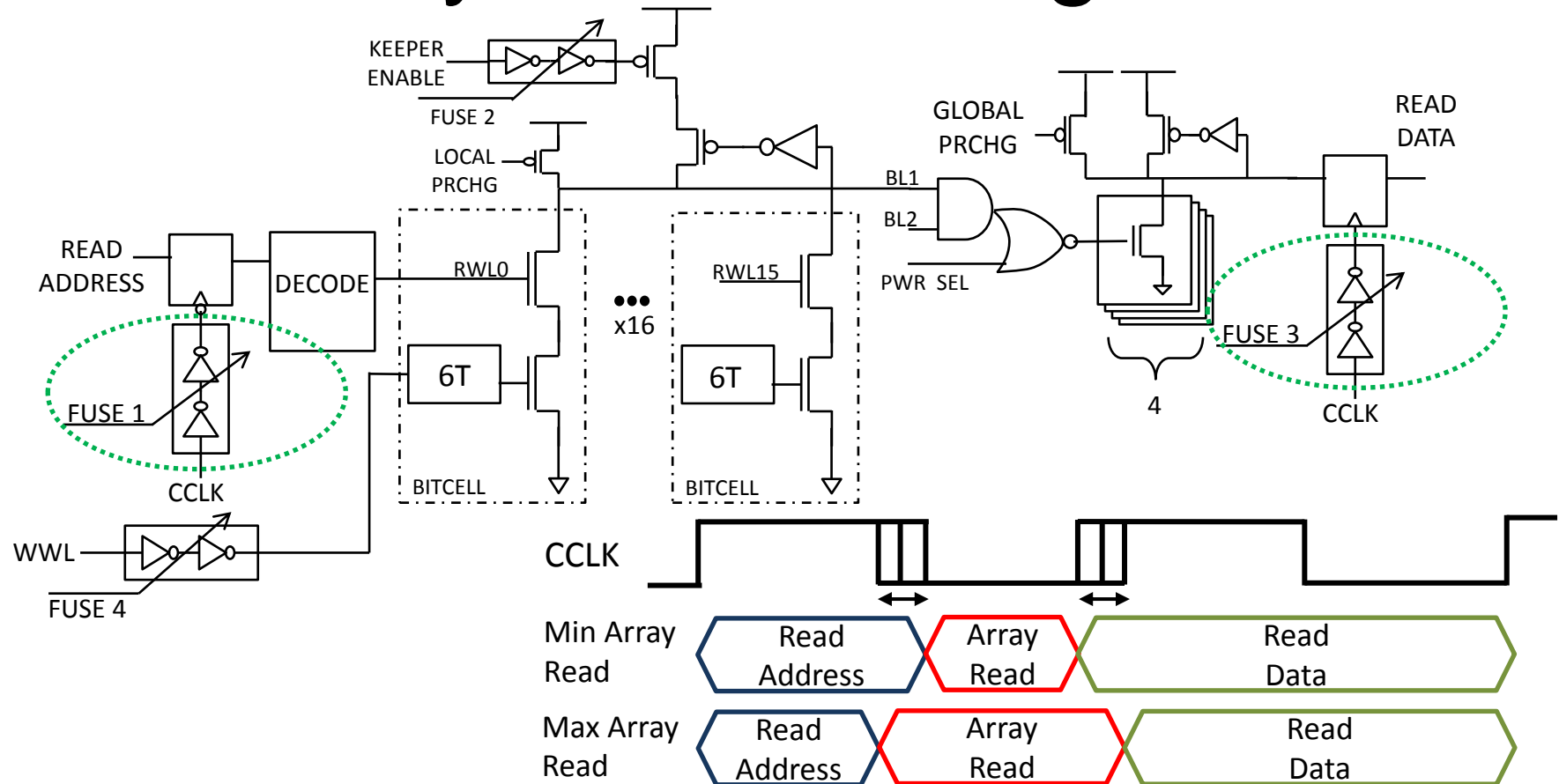
- FUSE1 (Read Address) and FUSE3 (Read Data) are used to modulate a half cycle access/write time
- These fuses control programmable delay cells and can be set per macro instantiation

Array Read Timing Fuses

Settings	Read Address delay (normalized to clock period)		Read Data delay (normalized to clock period)	
	High Voltage	Low Voltage	High Voltage	Low Voltage
00	14%	12%	11%	9%
01	5%	5%	7%	6%
10	10%	9%	15%	12%
11	18%	15%	18%	15%

- Four settings for both sets of fuses
- Delay ranges from 5-18% of clock period

Keeper Enable Fuse
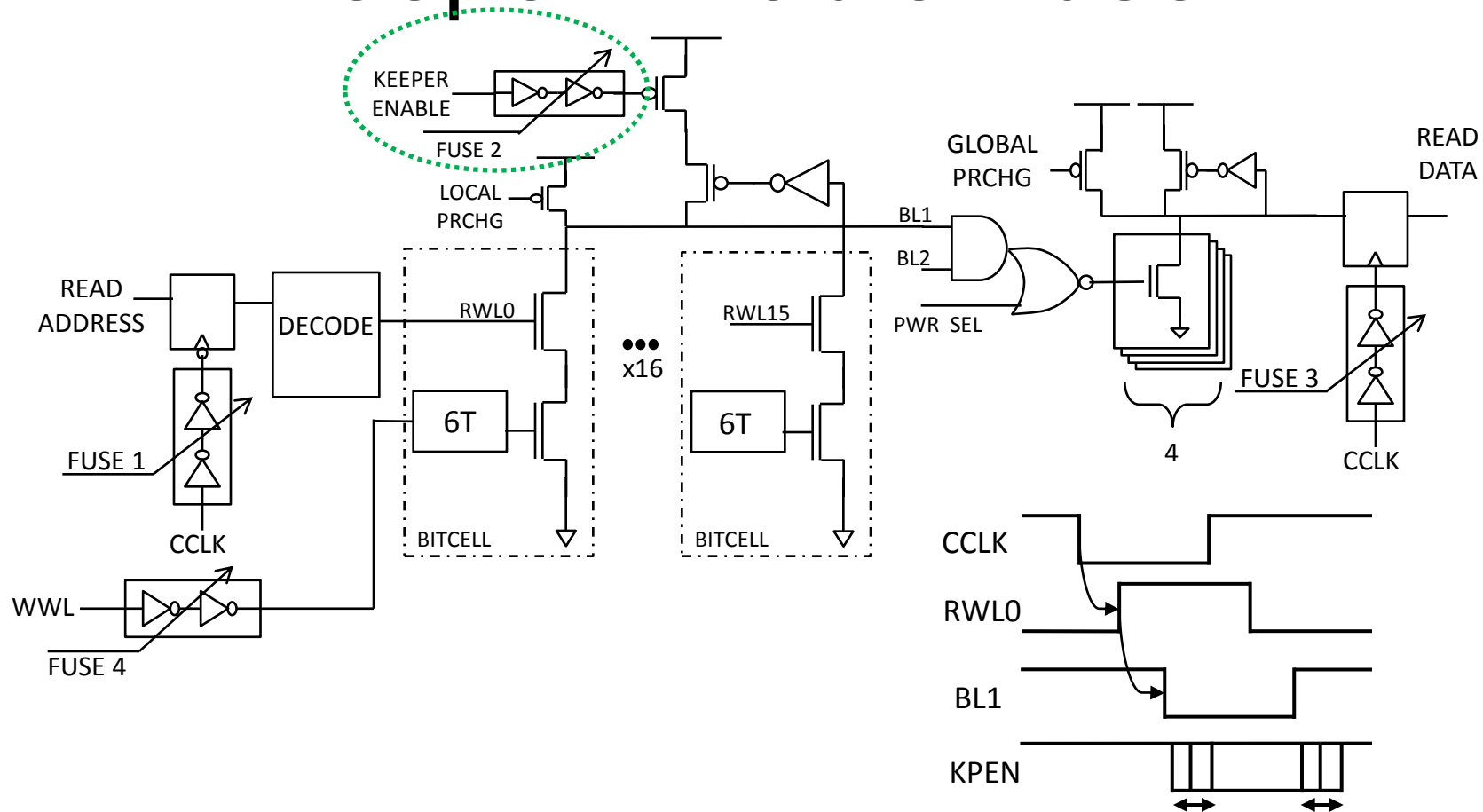
Keeper Enable signal can be delayed to improve performance or can be turned to an *Always ON* state for improved noise immunity

Keeper Enable Fuse

Settings	Bitline to Keeper Enable Delay	
	High Voltage (Normalized to clock period)	Low Voltage (Normalized to clock period)
00	1%	2%
01	6%	7%
10	11%	12%
11	Always ON	Always ON

- In the default case Keeper Enable turns on just after the bitline falls

- The keeper device is always on for *11* setting

Write Wordline Pulse Width Fuse

- WWL pulse width is chopped based on fuse settings
- Allows silicon measurement of write margin

Write Wordline Pulse Width Fuse

| Settings	Write Wordline Pulse Width	
	High Voltage (Normalized to clock period)	Low Voltage (Normalized to clock period)
00	56%	52%
01	34%	31%
10	28%	25%
11	18%	16%

- Pulse width is ~50% of clock period for the default setting

- Pulse width is controlled by combining write clock and its delayed inverted version

- Pulse width for non default settings are frequency independent

CU Level Clock Distribution
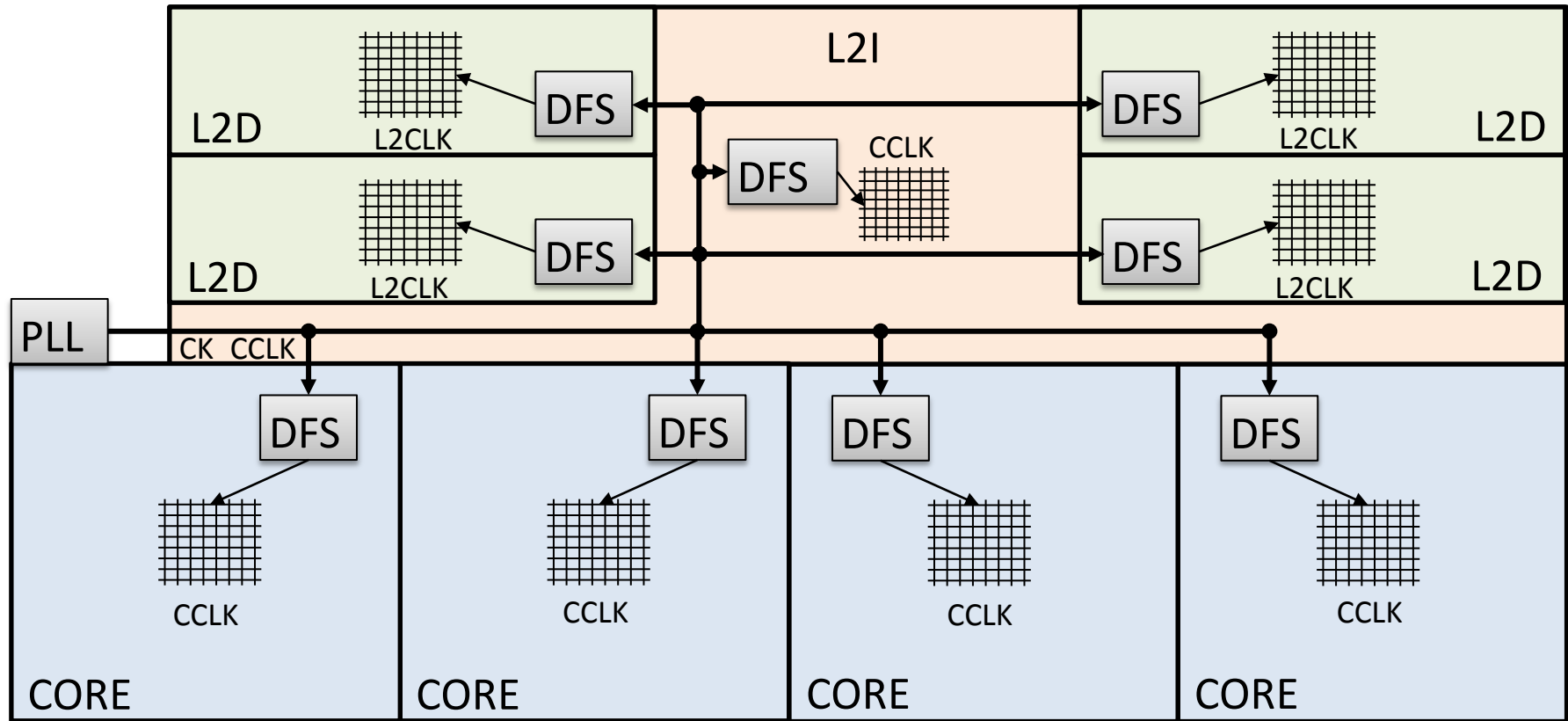
- Matched clock delay to all endpoints to minimize latency
- Each unit's clock independently gated to reduce dynamic power
- L2D half frequency operation supported without adding additional stages to clock path

DFS Design
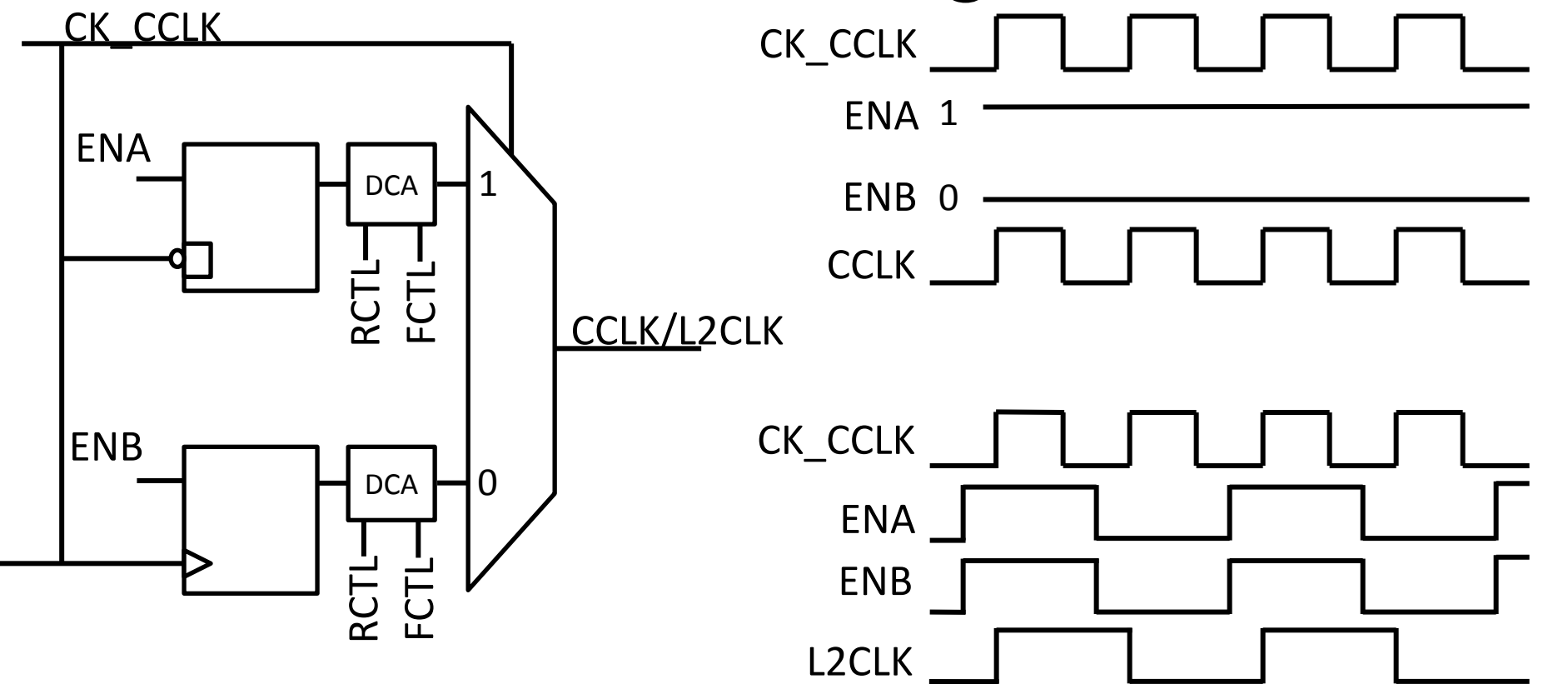
- Clock dividing for various operating modes
- Duty cycle adjuster for independent control of duty cycle within each block

Core clock distribution & CTS
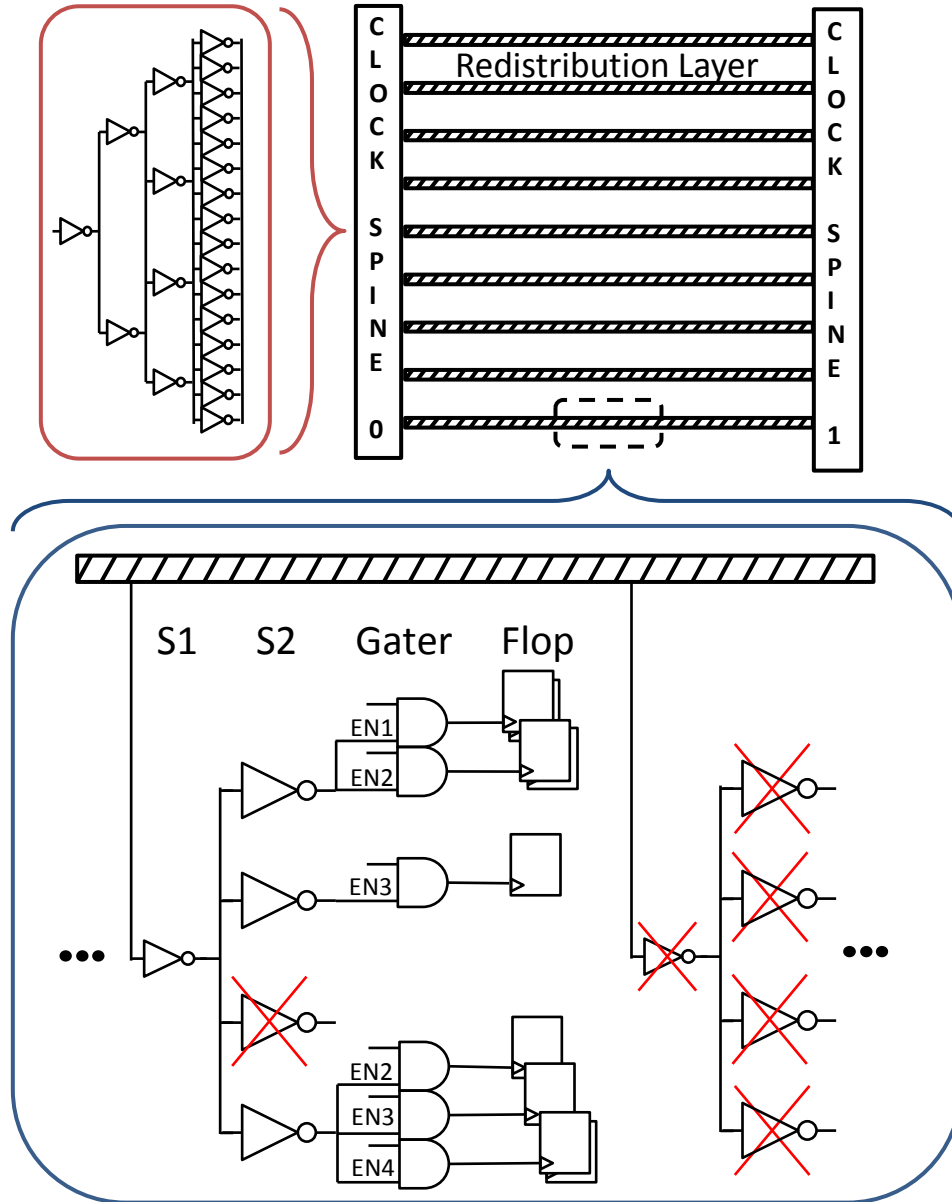
- Low skew recombinant mesh design

- Mesh driven by configurable custom cells to enable faster design closure and tunability

- Multipoint CTS start points created by preplacing 2 levels of inverters

- Delete unused S1/S2 levels

Timing Methodology

- Primary design optimization uses all Low Vt for speed and area

- Multi-Vt optimization done multiple times post-placement and in eco to reduce leakage

- Use Monte Carlo simulations to calculate Vt derates applied to High Vt and Regular Vt cells based on their variation relative to Low Vt
  - Ensure cells with large variation get sufficient margin
  - Ensure Si-critical paths are set by Low Vt

- Exclude cells with sigma/mean ratio worse than a set floor
  - Enable operation at lower voltages and expedite hold timing closure

Silicon Results
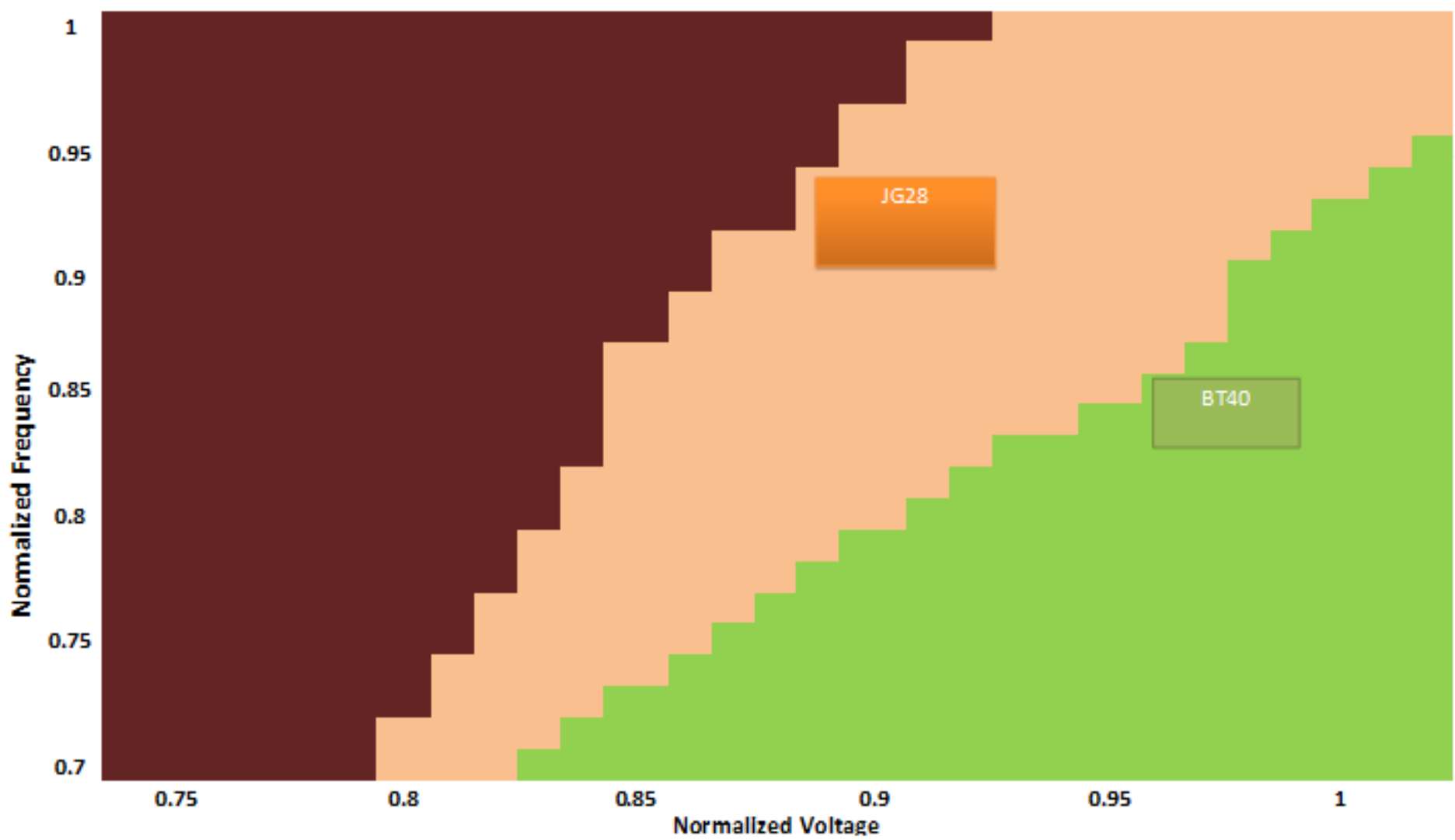
Power

- ## Dynamic
  - – Reduced number of clock spines versus BT
  - – Remove unused S1/S2 clock inverters
  - – Move clock spine to Low Vt versus BT
  - – Gate L2 clock when L2 not accessed
- ## Static
  - – Always ON buffer tree for power gate enables use longer length Hvt
  - – Vt usage tuned within custom arrays
 - – Measured silicon shows JG power gated leakage <10mW*

* Estimates based on internal AMD modeling using benchmark simulations. This information is preliminary and subject to change without notice.

Power Breakdown

JG Core Vt Cell Usage

JG Core Total Power Breakdown
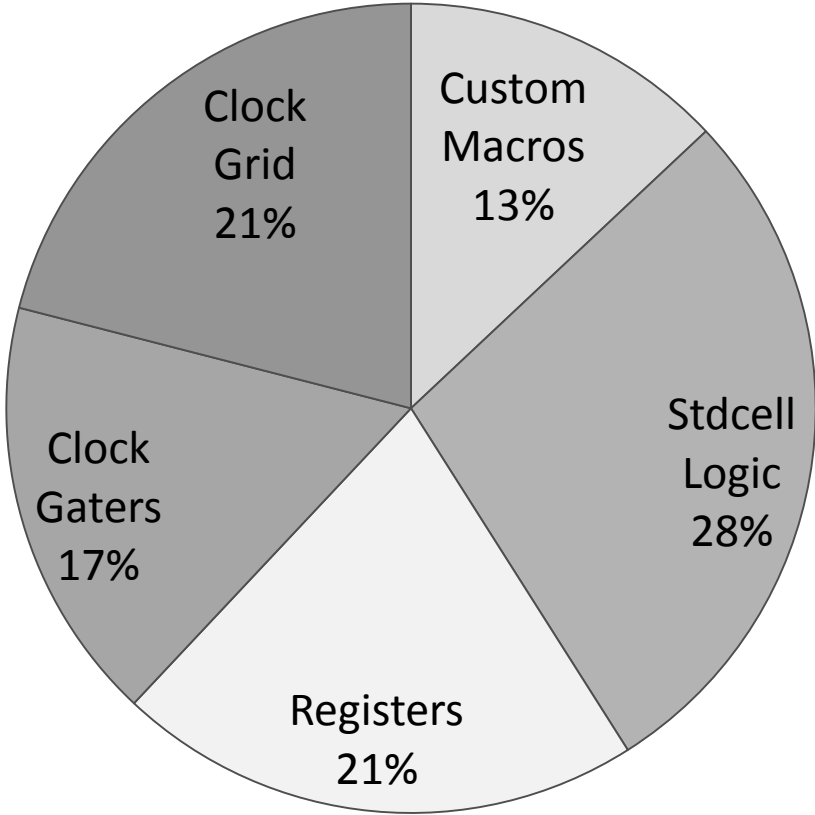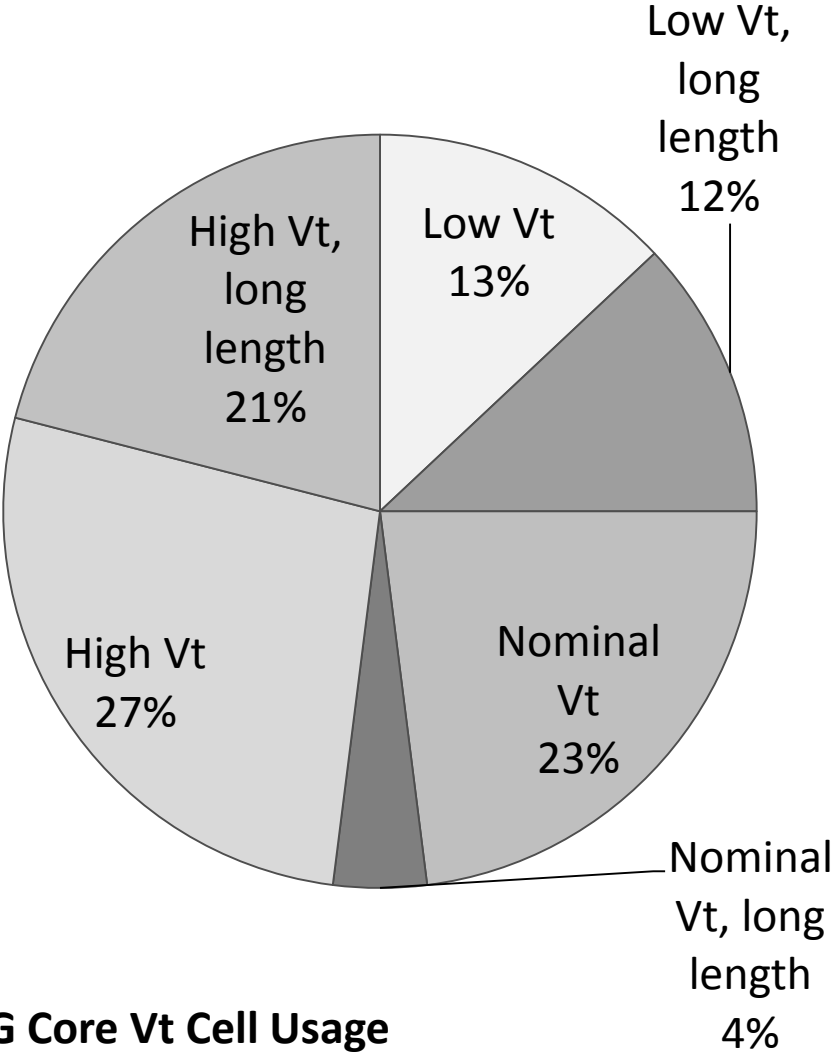
Reliability

- Design for superset of usage model conditions
- Numerous challenges for 28nm Vmin/Vmax support:
  - Time dependent and intra-metal dielectric breakdown
  - Bias Temperature Instability (BTI)
    - Use foundry calculator to determine Vt shift for given usage model
    - Use Vt shift in critical path simulations to gauge frequency degradation
    - Margin timing paths across units with different usage conditions via clock uncertainty
    - Compare pre-silicon to measured Si degradation
  - Electro-migration
    - Require statistical EM budgeting to close longest lifetime parts
    - Thermal solve used to reduce self heat pessimism for Irms calculations
    - Thermal map of RAM array shown
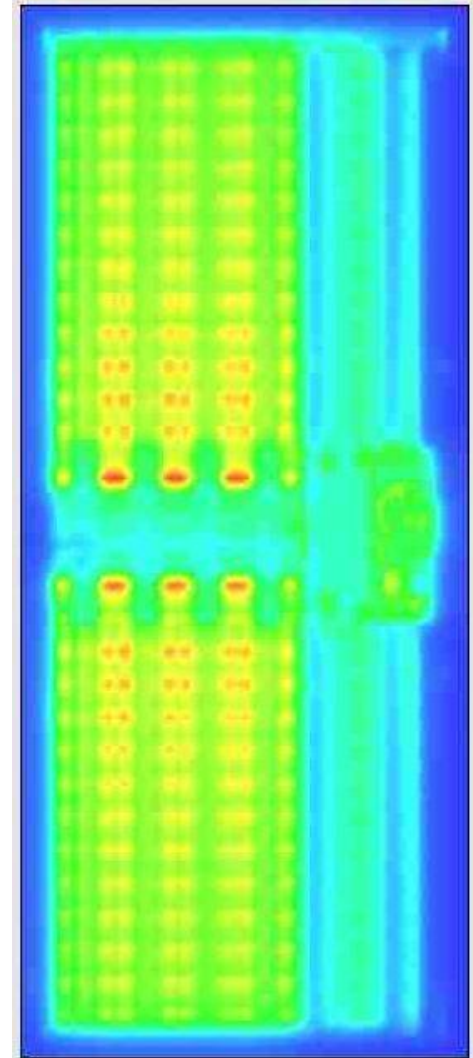
Conclusion

- "Jaguar" is first AMD 28nm bulk CPU

- Quad core with shared L2

- Substantially higher IPC and frequency than BT

- Unit built for reuse in multiple SoCs

- Design methods increase process portability

- Focus on high density and smaller chip area

- Low power and low skew configurable clock tree

- Highly utilize SAPR design flow but customize for high speed flops and programmable custom arrays

Trademark Attribution

AMD, the AMD Arrow logo and combinations thereof are trademarks of Advanced Micro Devices, Inc. in the United States and/or other jurisdictions. Other names used in this presentation are for identification purposes only and may be trademarks of their respective owners.

60 YEARS OF SOLID-STATE INNOVATION

1954 – 2013

60

ISSCC®

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS

UNIVERSITY OF PENNSYLVANIA