Simmani: Runtime Power Modeling for Arbitrary RTL with Automatic Signal Selection

Donggyu Kim, Jerry Zhao,

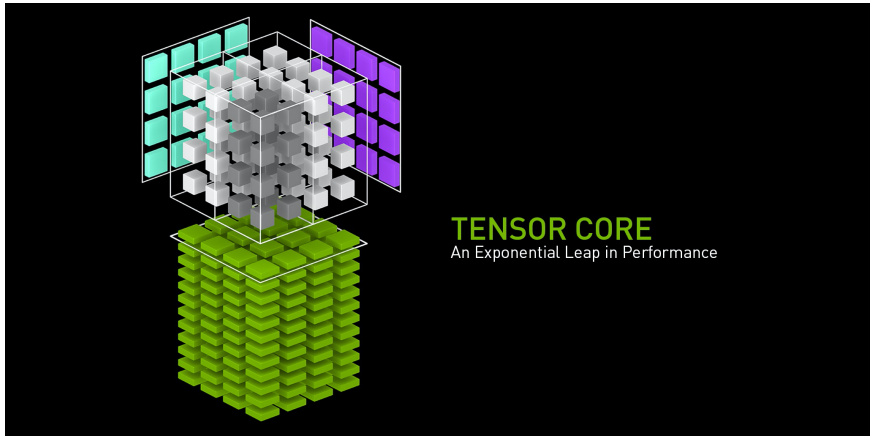
Jonathan Bachrach, Krste Asanovic

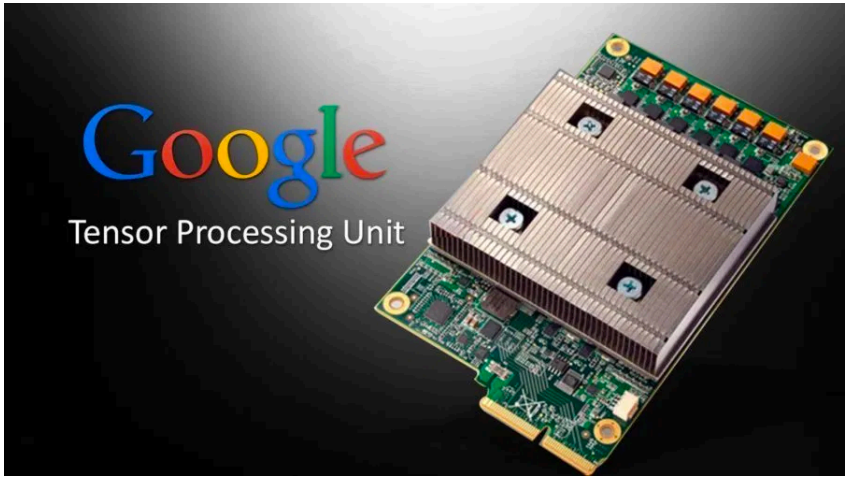
MICRO 2019

10/16/2019

Era of Accelerators

- What is the ***most efficient machine*** for emerging applications?





How efficient is my system?

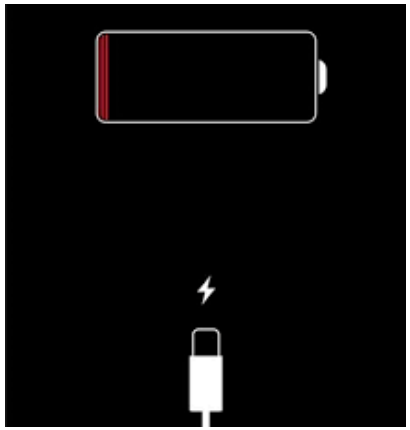
Energy Efficiency = Performance / Power

(task / energy) (task / time) (energy / time)

- Performance
  - How fast is my system?
- Power
  - A constraint for my system
- Energy efficiency
  - How much performance with a given constraint?

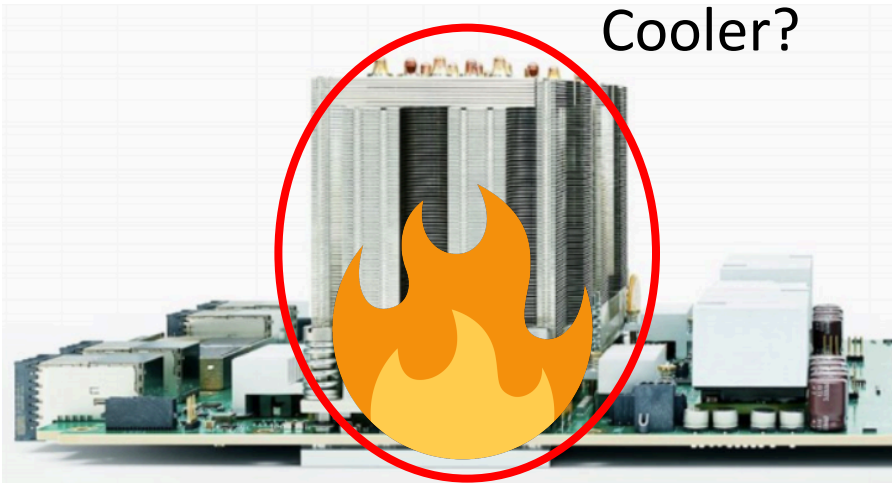
Why do we care about power & energy efficiency?

- All Day Battery Life

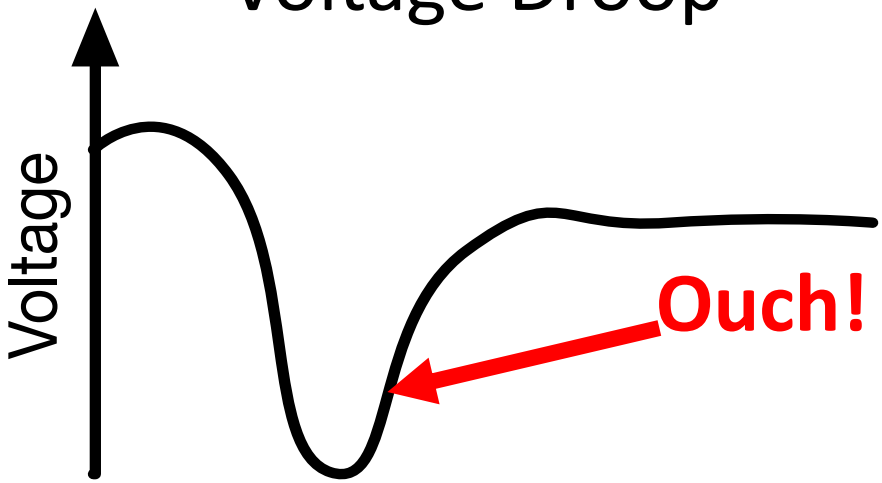


- Electricity Bill

- Heat



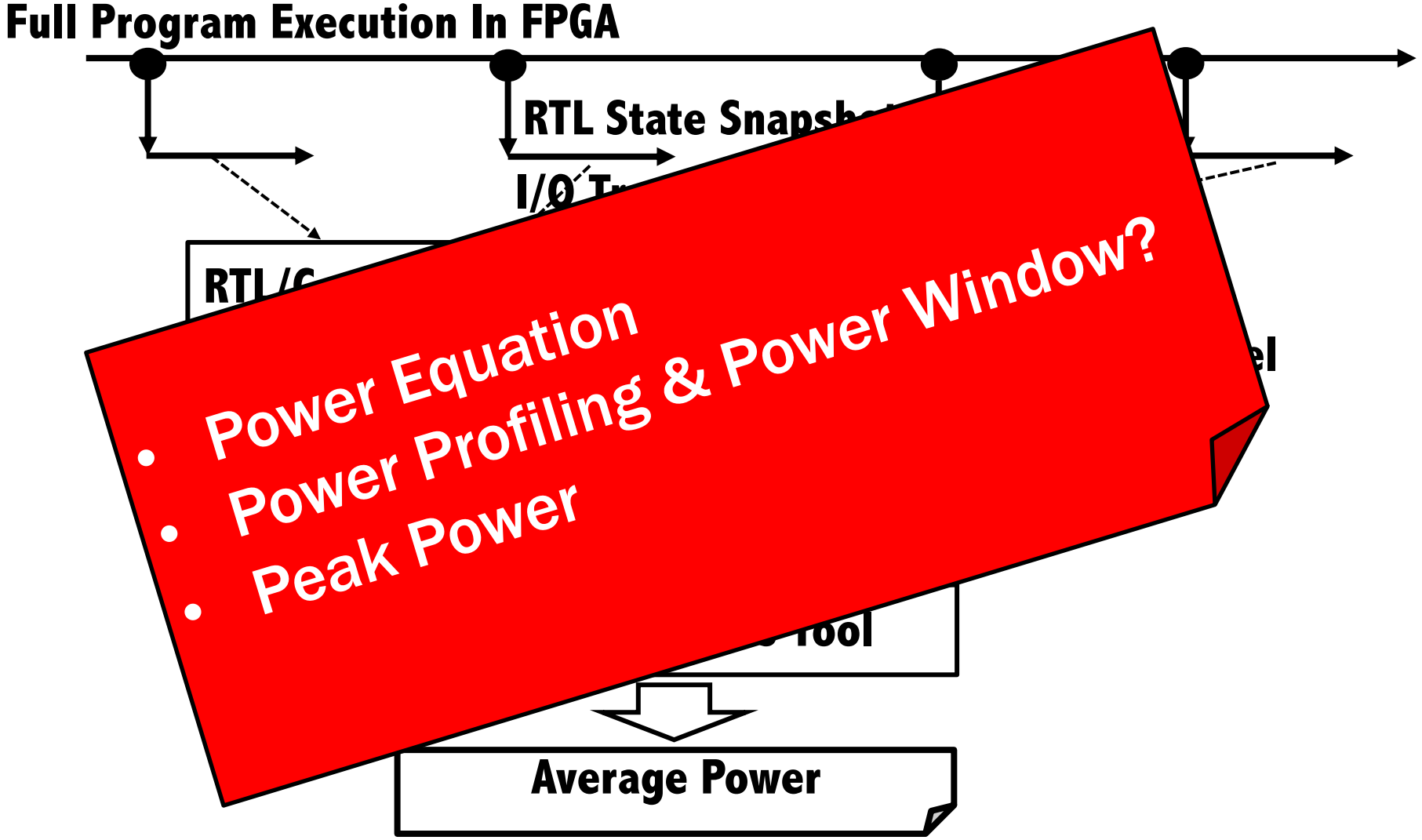
- Voltage Droop



How to do power & energy modeling?

- Power and energy efficiency
  - Power dissipation = *f(*signal activities*)*
  - Signal activities = *g(*applications*)*
- Commercial CAD tools with gate-level simulation (e.g. PrimeTime PX)
  - Most accurate but extremely slow
- Analytic modeling (e.g. McPAT)
  - Popular in computer architecture literature
  - Validation is hard, simulation is slow

Strober Power/Energy Modeling [ISCA `16]



[ISCA `16] Kim et al. "Strober: Fast and Accurate Sample-Based Energy Simulation for Arbitrary RTL"

Power Modeling 101

$$P_{total} = P_{dyn} + P_{leak} = \alpha\, C_L V_{DD}^2 f + I_{leak} V_{DD}$$

In Fact

$$P_{dyn} = \frac{1}{2} V_{DD}^2 \left(\sum_{i \in all\ signals} C_i\, D_i \right)$$ [1]

C_i: capacitance signal i drives

D_i: toggles per cycle of signal i

We hope

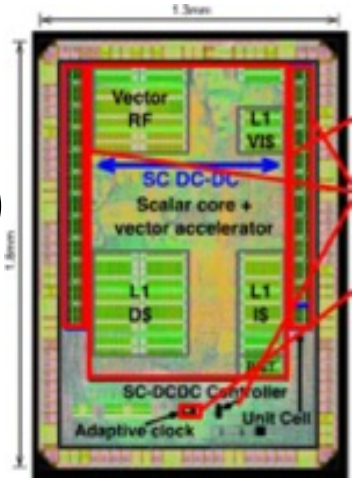
$$\approx \frac{1}{2} V_{DD}^2 \left(\sum_{k \in some\ signals} C_k\, D_k \right)$$

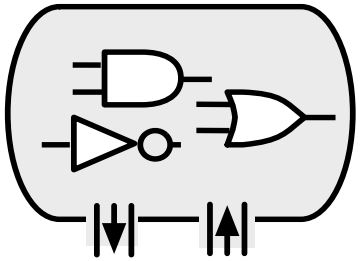
[1] Najm. "A Survey of Power Estimation Techniques in VLSI Circuits, IEEE Trans on VLSI, 1994

What Are *Key Signals*?

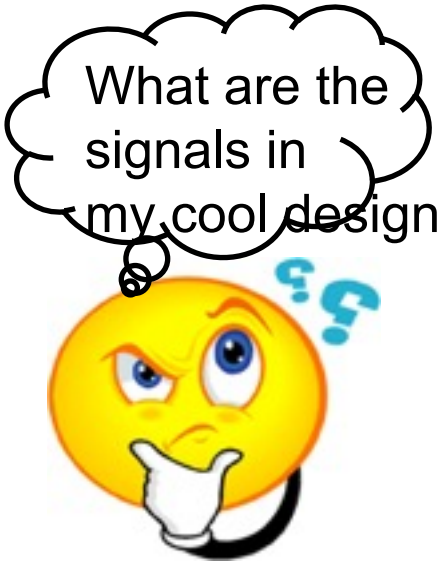
- Microprocessors
  - Well studied on µarch events (e.g. cache miss, branch misprediction)
  - Collect statistics from *existing performance counters*

$$\frac{1}{2} V_{DD}^2 \left(\sum_{\substack{k \in perf \\ counter}} C_k \, D_k \right)$$





AI Accelerator

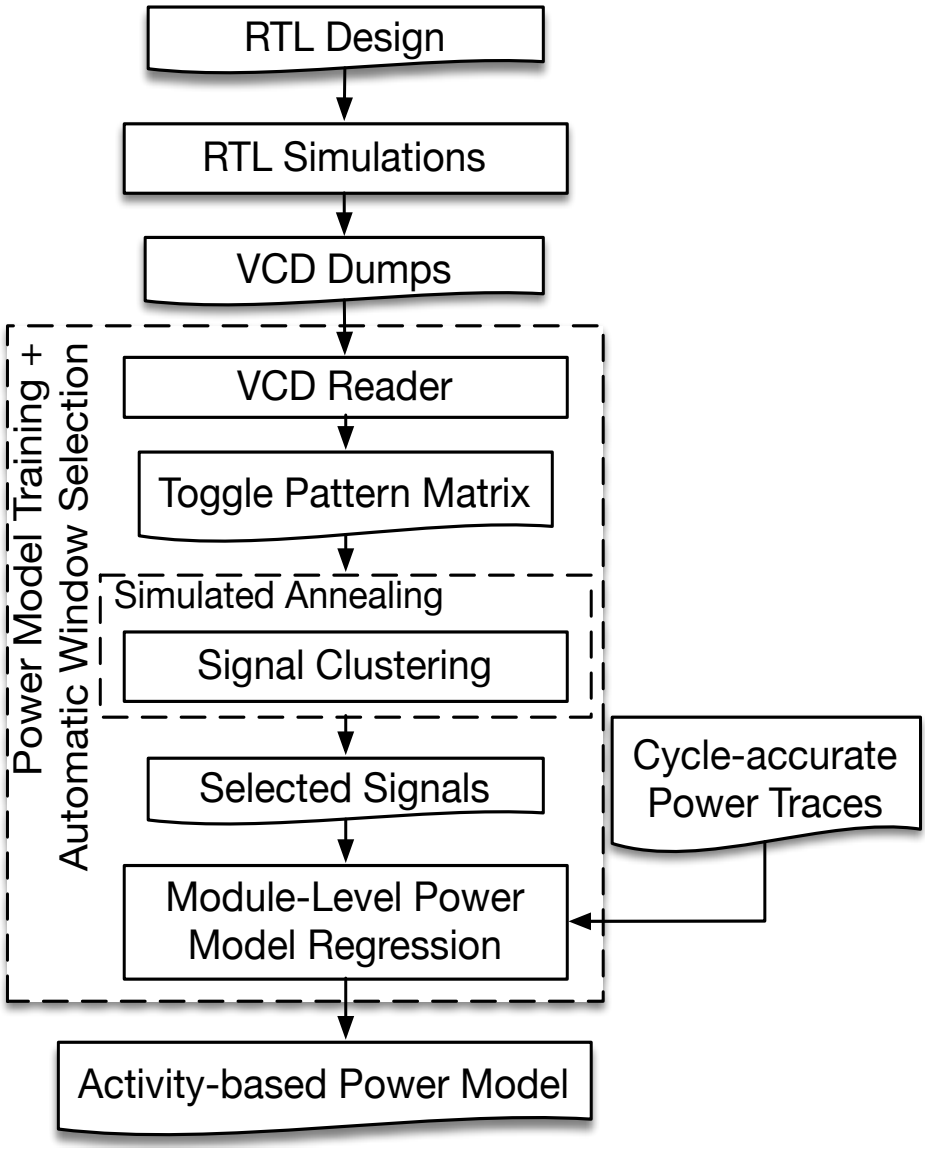


- What about AI accelerators?
  - Unlikely you have intuition
  - Very unlikely there are performance counters
  - ***Signals should be automatically selected***

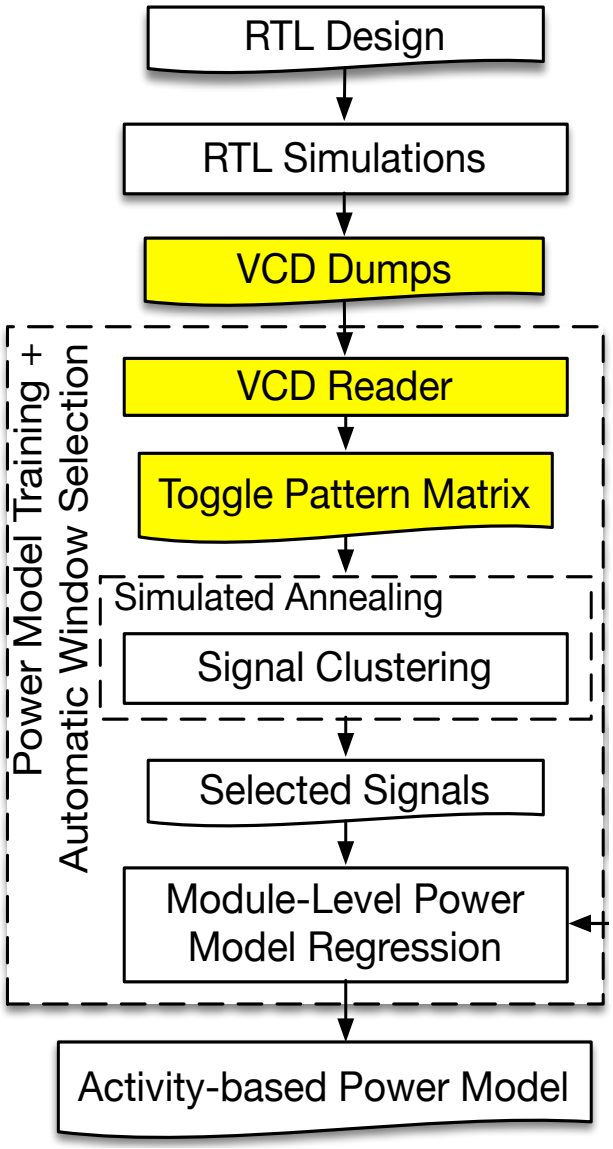
Simmani: Runtime Power Modeling with Automatic Signal Selection

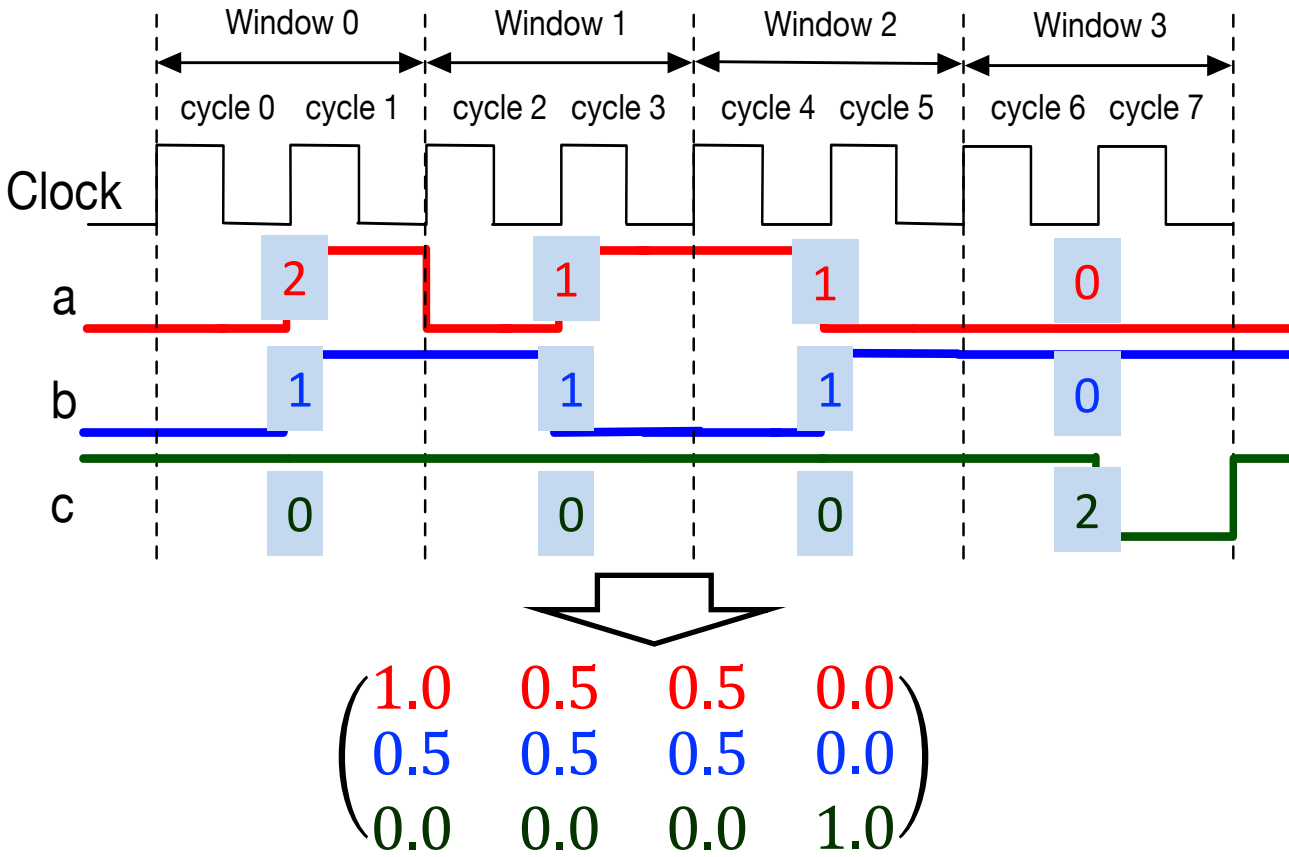
- Goal
  - Find *key signals for power dissipations from any RTL*
- Observation
  - Signals showing similar toggle patterns
    ➔ Similar effect on dynamic power dissipation
- Our Approach
  - Construct ***toggle pattern matrix*** from VCD dumps
  - Select key signals with ***signal clustering***
  - ***Module-level power model*** regression against power traces from CAD tools
  - ***Automatic counter instrumentation*** for runtime power estimation with FPGAs

Power Model Training Flow



Constructing Toggle Pattern Matrix

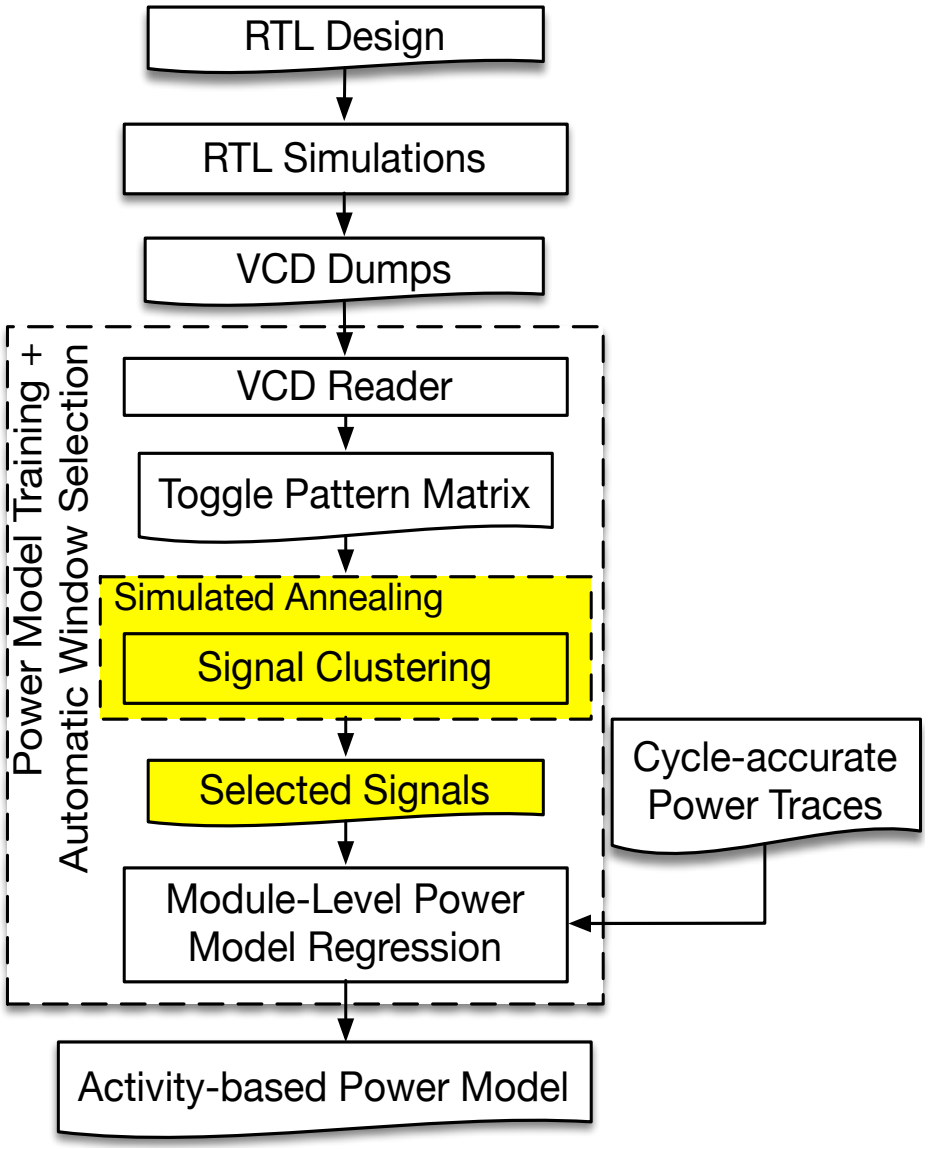




$$\begin{pmatrix} 1.0 & 0.5 & 0.5 & 0.0 \\ 0.5 & 0.5 & 0.5 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{pmatrix}$$

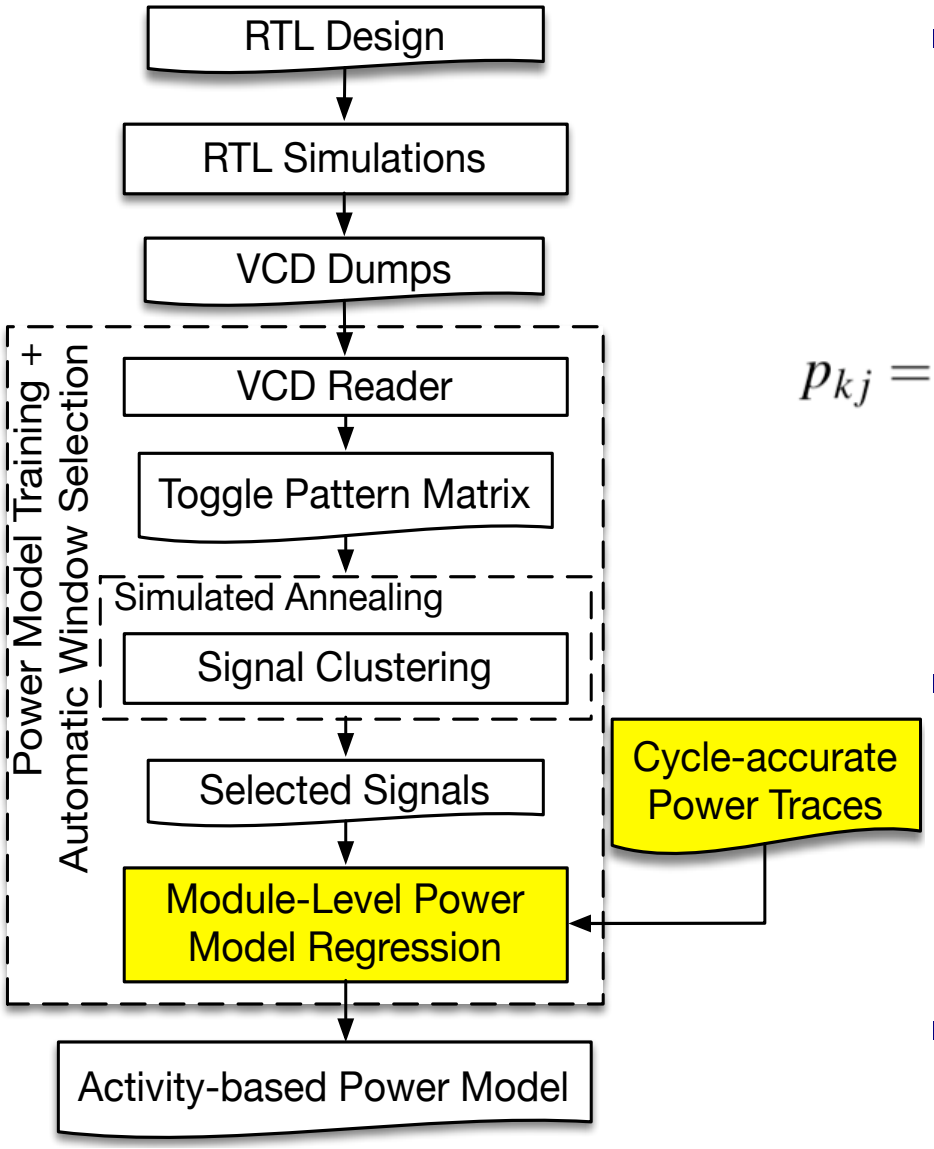
- Similarity: measured by Euclidean distance
- Matrix is big, but signal activities are sparse
  ➔ Compressed sparse row (CSR) format

Selecting Key Signals



- Signal clustering
  - Project high-dimensional data into low-dimensional space
  - Clustering (k-means) on the projected points
  - Select signals closed to the cluster centers
- Optimal number of signals
  - Model selection (# clusters) with Bayesian Information Criterion (BIC)
  - Simulated annealing for global optimum

Power Model Regression



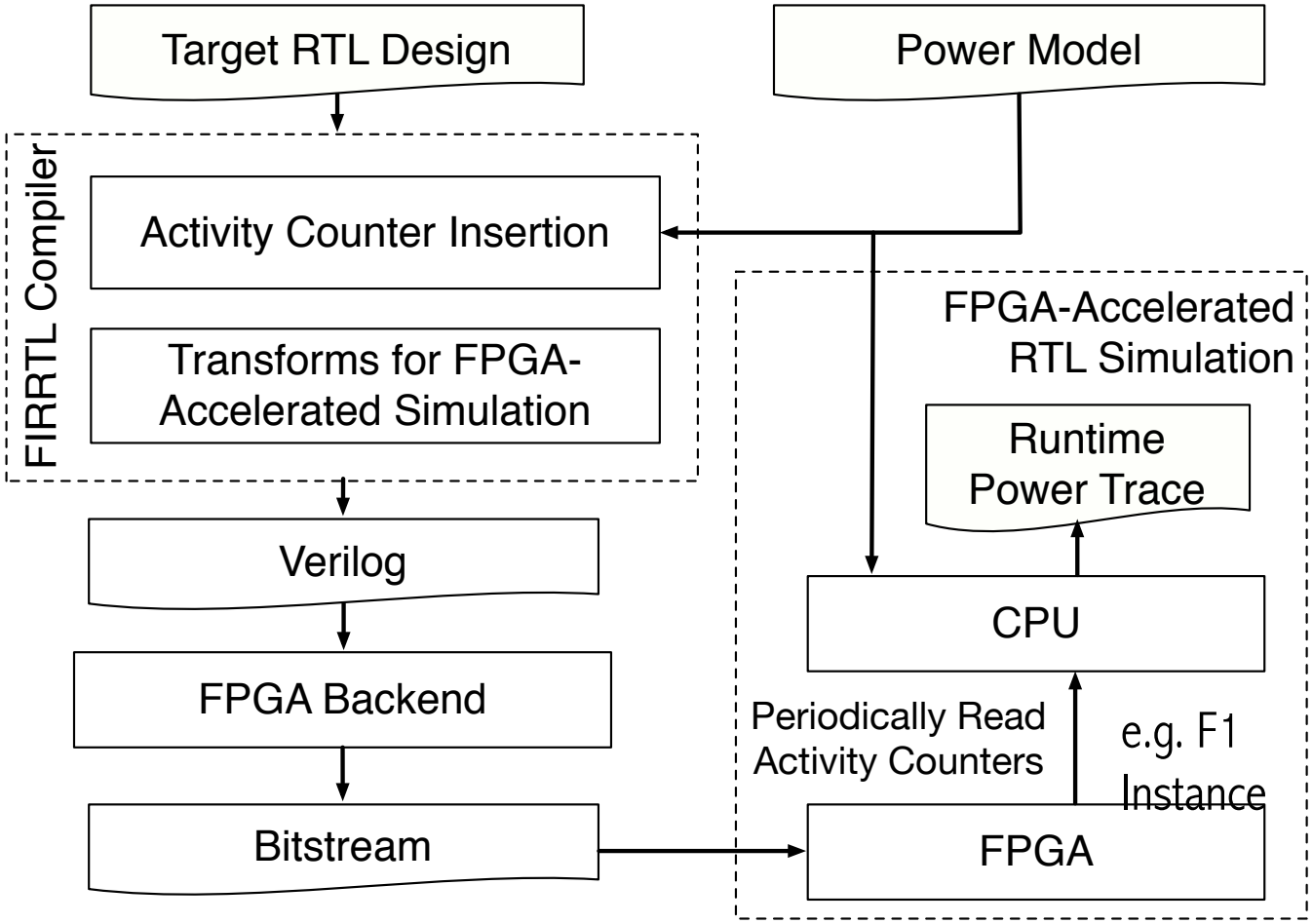
- Linear model with polynomial features → simple, intuitive, low training & inference overhead

$$p_{kj} = \alpha + \beta_1 x_{1j} + \beta_2 x_{2j} + \cdots \beta_n x_{nj} + \beta_{11} x_{1j}^2 + \beta_{22} x_{2j}^2 + \cdots + \beta_{12} x_{1j} x_{2j} + \cdots + \beta_{123} x_{1j} x_{2j} x_{3j} + \cdots$$

- Regularization & variable selection with the elastic net → prediction accuracy + interpretability
- Constrain coefficients >= 0 (except uncore)

Power Model Instrumentation

- Automatically insert activity counters for selected signals
- Read counters periodically from FPGA-accelerated simulation



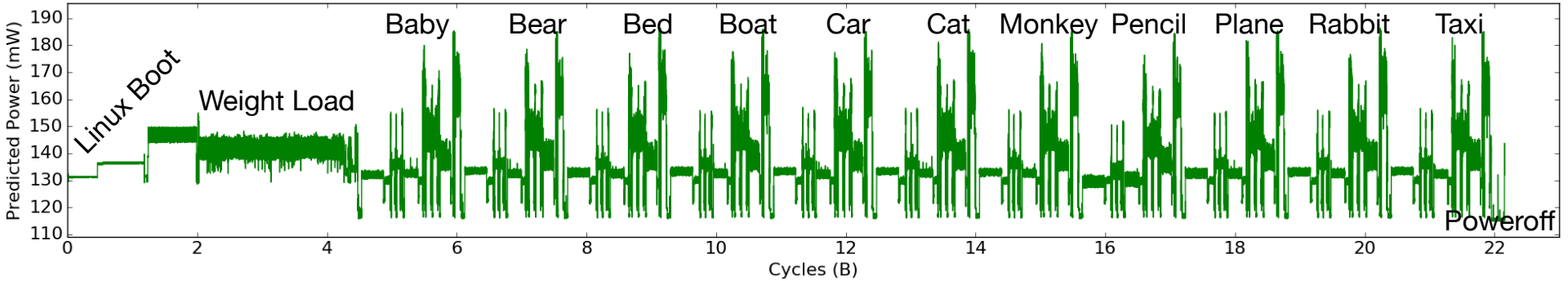
Experimental Results

- Target: Rocket(in-order processor) + Hwacha(vector accelerator)
- Total # of RTL signals = 115K
- Training set: ISA tests, µbmarks, random samples from long workloads.
- Technology: TSMC 45nm
- Training time
  - CAD tool (DC + ICC + PrimeTimePX) : ~ a day
  - Power model training: ~ a half day
  - FPGA compile time: ~ 6hrs
- Errors (AVGE, RMSE) for µbenchmarks (test set) <= 9 %

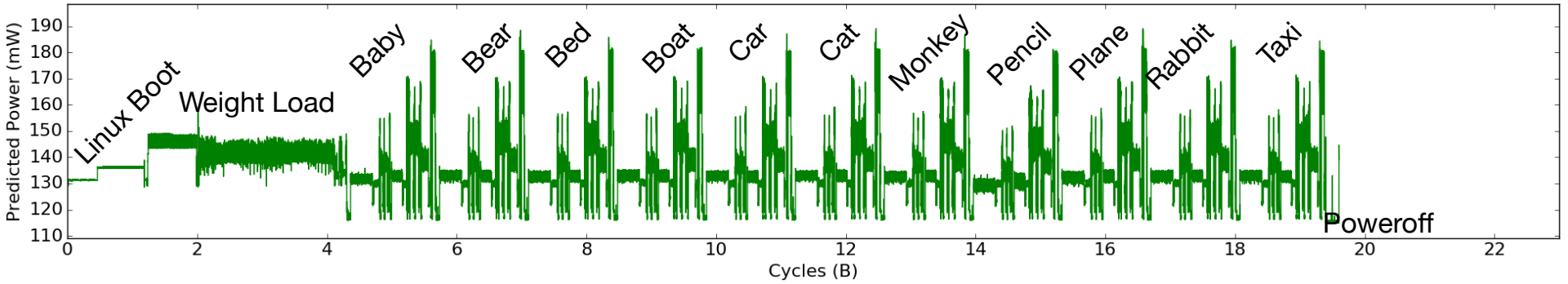
Power Trace of SqueezeNet on Hwacha

- Inference on 11 images (~22B cycles)
- Counters are samples every 100K cycles from the FPGA
- Errors against Strober <= 10%

SqueezeNet Baseline



SqueezeNet Quantization+Compression



What's Next with Simmani?

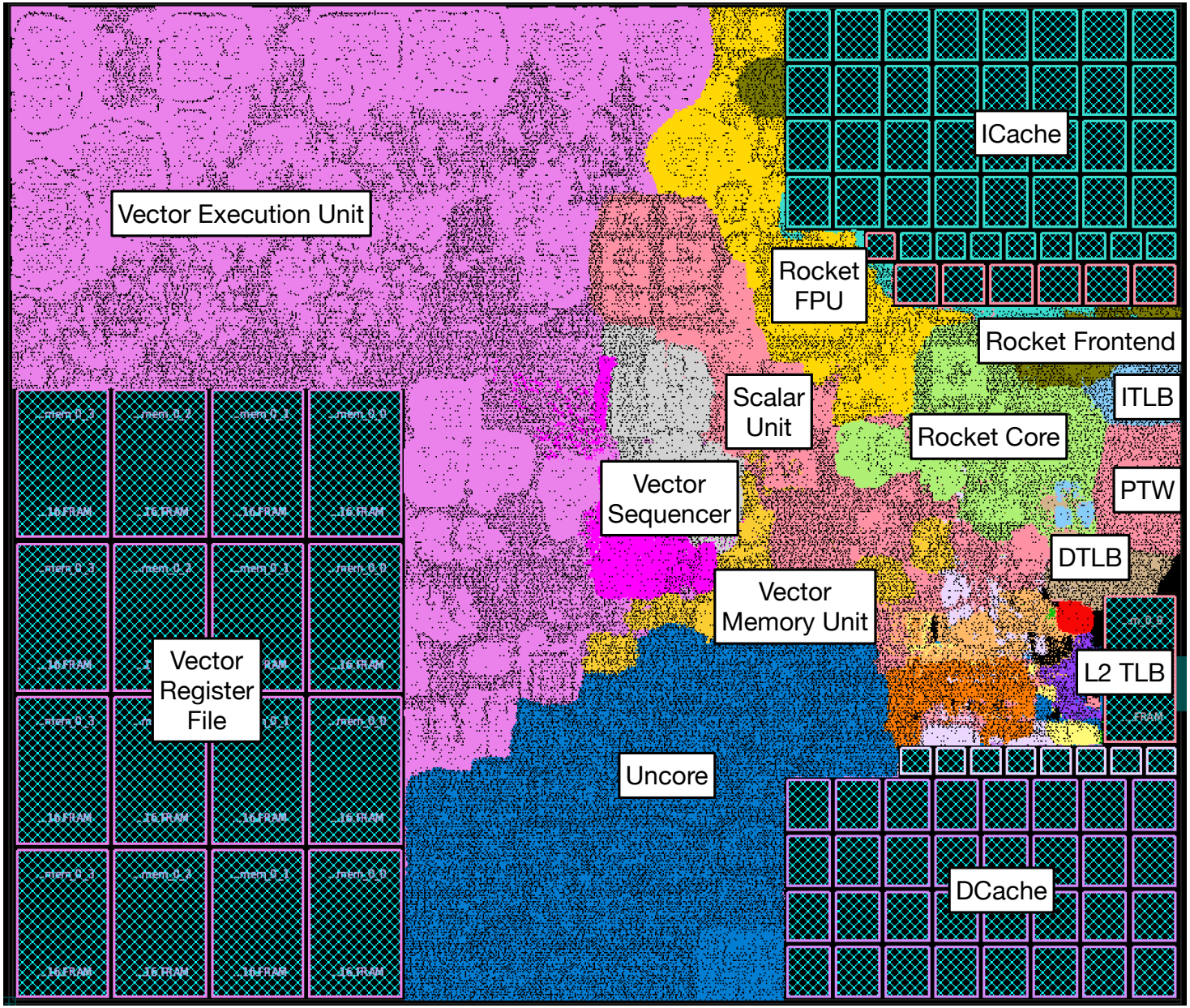
- Power window identification from power profiling
- Power virus problem
- Thermal modeling
- Average power management (e.g. DVFS)
- Peak power management for heat/voltage droop mitigation

- Open-source: simmani.github.io

Summary

- Power, Energy Efficiency, Heat Dissipation
  - Major concerns for computer systems
  - Runtime power modeling for design-time evaluations as well as dynamic power optimizations (e.g. DVFS)
- Simmani Power Modeling
  - Construct ***toggle pattern matrix*** from VCD dumps
  - Select key signals with ***signal clustering***
  - ***Module-level power model*** regression against power traces from CAD tools
  - ***Automatic counter instrumentation*** for runtime power estimation with FPGAs
  - More studies on dynamic power/thermal management for various classes of systems!
- Open-source: simmani.github.io

Floorplan


Vector Execution Unit
ICache
Rocket FPU
Rocket Frontend
Scalar Unit
ITLB
Rocket Core
Vector Sequencer
PTW
DTLB
Vector Memory Unit
Vector Register File
L2 TLB
Uncore
DCache