

## Optimized Register File Implementation of SRAM Bit Cell

**J. Madhavi**

M.Tech Student,  
Department of ECE,

Krishna Chaitanya Institute of Technology & Sciences.

**M. Raju**

Asst Prof,

Department of ECE,

Krishna Chaitanya Institute of Technology & Sciences.

### Abstract:

This project presents a predictive SRAM power model that reduces the changes required to adapt existing models to handle new circuit topologies, process corners, and design space exploration. Analytical equations model the impact of varying common characteristics such as bit-width, entries, segmentation, gating, and sizing while topology specific characteristics are captured empirically from a reference design. In validating the model, we will generate the production quality schematic and layouts of different RF topologies using an Electric tool. We then extract the layout parasitics and run an Electric analysis tool to obtain the actual dynamic and leakage power for all configurations. Electric Tool is used to design the schematic and layout level diagrams of our project. The LT-SPICE Tool will be used for simulation of the Spice code which tests the functionality of our generated layout and schematic blocks.

### KeyWords:

Register File, SRAM, Power Model, Leakage Power, Dynamic Power, Reference Design.

### 1. INTRODUCTION:

In order to achieve high performance/watt in future deeply-scaled CMOS technologies, accurate prediction of power is critical for early-stage architectural design explorations of performance and power tradeoffs. Register files (RF) consume a significant portion of embedded and high-performance processors [1, 2] power. A large number of studies that explore energy efficiency tradeoffs involve changes to RFs. Hence, accurate power modeling of SRAMs is important for early architectural explorations.

Consider a typical micro-architectural study to explore a range of RF bit/entry sizes for best power/performance tradeoff. A modern processor and SoC could easily have >30 unique and custom RFs [1, 2]. Current parametric approaches for estimating RF power, analytical or empirical, are based on specific topologies and circuit implementations. To model a different RF topology, today's architectural power models and performance simulators [3, 4] either use the existing power model essentially unchanged (inaccurate for the new topology) or modify existing models for the different topology (time consuming). Analytical use device process parameters to calculate the power using analytical equations that model the key capacitances (dynamic) or transistor sizes (leakage) in the RF. To adapt these models to different topologies/technologies require changes to the analytical formulas and parameters. Our proposed approach does not require any changes to the analytical formulas.

Empirical models rely on power simulation on the implemented circuit for the entire SRAM. A major drawback of regression based models [6, 7] is they require the implementation of several RF configurations to curve fit the empirical data for each topology and technology. Applying statistical techniques, such as design of experiments, typically requires at least 5 data points to accurately fit the data. Empirical models are therefore only valid for the specific circuit topology and technology used to generate the model coefficients. Thus, they usually present a method rather than reusable model equations. Liang [8] presented a hybrid model that empirically captures the power of three array structures (1-bit x 1-entry, 2-bit x 1-entry, and 1-bit x 2-entry) and composes them analytically to obtain the power of an n-bit, m-entry structure. To reuse the model without modification however requires empirical data from the 3 specific array configurations on which the model is based.

Moreover, these 1bx1e, 2bx1e, 1bx2e configurations do not exist in real design. As shown in Figure 4 and 5, using very small array configurations as reference to predict power for larger configurations is less accurate due to circuit and layout anomalies that could be magnified in very small arrays. We present a hybrid model that addresses the aforementioned limitations of adaptability, reusability, and for the first time expands the architect's exploration options to include circuit-level design choices of segmentation, gating, and sizing. Our hybrid model does not calculate the base leakage and dynamic power values which are process technology and circuit topology dependent as in cacti [9]. Instead we rely on a single "reference design" to capture those dependencies and model relative changes from the reference.

The empirical "reference design" data, which is an input parameter, captures topology-specific characteristics such as dual-ended/single-ended writes/reads, static/dynamic read, and process technology dependencies. We then analytically model the impact of cross-topology features such as changes in bit-width, entry-count, and common designer choices such as segmentation, gating, and sizing; using the same analytical model for all topologies. To further improve model accuracy and adaptability, we derive an equation for each RF stage independently. This enables the capture of stage-specific characteristics, thereby reducing prediction error and making the model easily adaptable to different SRAM topologies.

The distinct advantages of the modeling approach presented in this paper as compared to previous efforts are:

- A single adaptable model that can accurately predict power for different topologies by only modifying the input parameters of the model.
- Requires only a single reference design empirical data.
- Allows the use of any n-bit, m-entry reference to empirically capture design, topology, and technology specifics.
- Enables design space exploration of circuit implementation choices of gating, segmentation, and device sizing.

We validated our model on fully extracted layout of 3 different topologies, each with ~25 RF configurations. Section 3 presents the model results with an average error range of 5% (leakage) and 7% (dynamic). Section 4 presents scenario application of the proposed model to design space exploration of power sensitivity of two distinct topologies. Section 5 discusses summary and the accuracy of our results, suggesting that the power model presented herein can be used to easily and accurately predict and explore the power of an RF for any topology.

## 2. POWER MODEL:

### 2.1 General Model Approach :

The model is a hybrid of analytical equations and empirical data. We use an analytical approach to model topology independent impacts and empirical data by way of a "reference design" for topology and technology specific characteristics. A "reference design" refers to a circuit implementation of one configuration (bits/entry) of the topology under study from which power and timing data is known or can be obtained. A reference design is required for each distinct circuit topology. We capture the empirical data for each stage of the reference design (Figure 1, 2) and model the relative change in power due to changes in bitwidth, number of entries, delay, and common designer choices such as segmentation, gating. We use this approach to make the model adaptable to different design topologies. We model each stage independently. This enables accurate modeling of the unique characteristics of each stage and easy adaptation of the model to different SRAM topologies. The model is of the form:

$$Power_{stage} = f \left\{ \begin{array}{l} RefStagePower, Bit, Entry, \\ Segmentation, Sizing, Gating, AF, SP \end{array} \right\} \quad (1)$$

$$Power_{Total} = \sum Power_{stage} \quad (2)$$

We use both delay and power models to capture the totality of the impact of design choices. In normal design, the increase in array size by additional bits and entries results in the need to increase drive strength for timing. Since leakage and capacitive loading correlates with device size, the impact of bits and entry growth on sizing is modeled by a delay penalty. The model uses a delay threshold number to capture the realistic design scenario where the driver is not upsized for any arbitrary increase in bits or entries but only after a specific threshold.

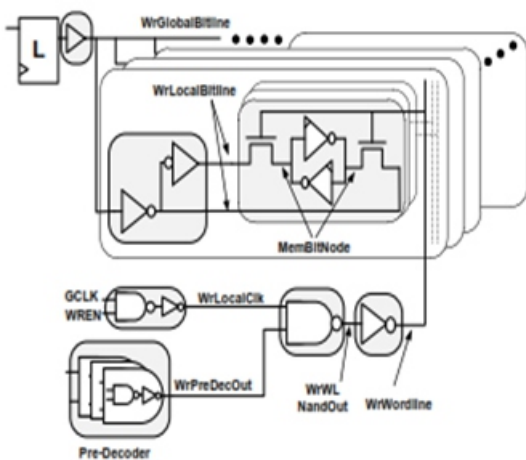
### 2.2 Unified Stage Model:

We use a single unified model for all stages. Thus, bits and entries are used interchangeably in the model equations depending on the loading seen by the stage driver. The subscript “x” denotes the entity (bits or entry) whose increase (decrease) results in increased(decreased) loading on the stage driver. Subscript “y” denotes the orthogonal entity that does not affect the load on the stage driver.

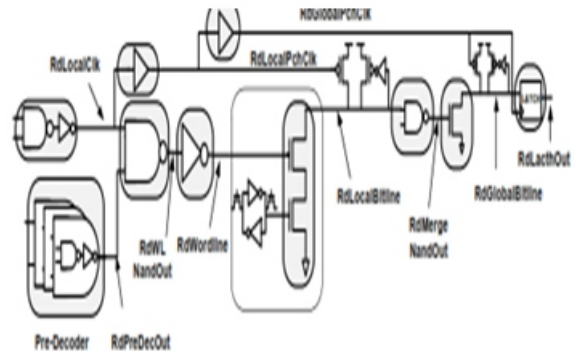
For example, the “x” and “y” entities of a wordline represents “bits” and “entries” respectively since a change in number of bits changes the load on the wordline driver. On the other hand, for the bitlines “x” and “y” entities represents “entries” and “bits” respectively since bitline driver load depends on the number of entries. The words “entity” (“entities”) therefore refers to bit (bits) or entry (entries) depending on stage.

### 2.3 Register File Topology:

Figures 1 and 2 show an illustration of a register file write and read stage definitions. While our model is not specific to this topology, a typical RF/SRAM topology can be broken down to these basic stages. To model a different topology, the reference design is decomposed into component stages. The impact of each stage’s distinct characteristic on power and delay is captured by the reference design per stage empirical data.



**Fig 1: RF Write path showing write stages. Each stage is modeled independently.**



**Fig 2: RF read path showing read stages. Each stage is modeled independently**

Typically, large array bitlines are segmented into local (primary) and global (secondary) bitlines. A segment is an instance of a physically connected stage node. Thus a stage can have multiple instances of a segment. A global bitline drives (e.g. “WrGlobalBitline” stage) or combines (e.g. “RdGlobalBitline”stage) multiple local bitline segments. Thus the characteristics of a global bitline (number of drivers, gate loading, etc.) depend on the local bitline segments. To capture this in a unified model, two segmentation parameters,  $N_{xpersegment}$  and  $N_{xpedrsegment}$  defined.  $N_{xpersegment}$  represents the number of entities per segment while  $N_{xpedrsegment}$  is the number of entities per dependent segment. The number of entities (bits or entries) for a stage is therefore scaled by its dependent segment as:

$$N_{xest} = \text{Ceil} \left[ \frac{N_{x\_estimate}}{N_{x\_persegment}} \right] \quad (3)$$

$$N_{xref} = \text{Ceil} \left[ \frac{N_{x\_reference}}{N_{x\_persegment}} \right] \quad (4)$$

$$N_{xperwireseg} = \text{Ceil} \left[ \frac{N_{x\_persegment}}{N_{x\_persegment}} \right] \quad (5)$$

$N_{x\_estimate}$  : Number of “x” entities to be estimated for the stage.  
 $N_{x\_reference}$  : Number “x” entities of the reference design stage.  
 $N_{x\_persegment}$  : Maximum number of “x” entities per segment.

$N_{xpedrsegment}$ : Number of “x” entities per dependent segment. If the stage has no dependency,

$N_{xest}$  : Number of scaled “x” entities to be estimated for the stage.

$N_{xref}$  : Number of scaled “x” entities of the reference design stage.

$N_{xperwireseg}$  : Number of scaled “x” entities per physically connected wire segment

## 2.4 Leakage Power Model :

### 2.4.1 Stage Leakage Power ( PstageLeakage ) :

The stage leakage is modeled by the driver instance count relative to the reference design.

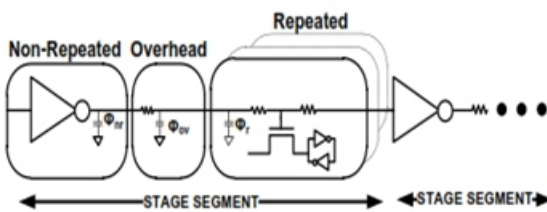


Fig 3: Stage cap definition for dynamic power model.

$$P_{\text{StageLeakage}} = P_{\text{stageleakage\_reference}} \times SP_f \times \left[ \lambda + (1 - \lambda) \times \partial_{\text{effectsize}} \times \left( \frac{N_{x\text{driver}}}{N_{x\text{driver\_ref}}} \right) \times \left( \frac{N_{y\text{driver}}}{N_{y\text{driver\_ref}}} \right) \right] \quad (11)$$

$$N_{x\text{driver}} = \text{Ceil} \left[ \frac{N_{x\_estimate}}{N_{x\_perdriver}} \right] \quad N_{x\text{driver\_ref}} = \text{Ceil} \left[ \frac{N_{x\_reference}}{N_{x\_perdriver}} \right]$$

$$N_{y\text{driver}} = \text{Ceil} \left[ \frac{N_{y\_estimate}}{N_{y\_perdriver}} \right] \quad N_{y\text{driver\_ref}} = \text{Ceil} \left[ \frac{N_{y\_reference}}{N_{y\_perdriver}} \right]$$

$P_{\text{stageleakage\_reference}}$  : Reference design empirical stage leakage  
 $N_{y\_estimate}$  : Number of “y” entities to be estimated for the stage  
 $N_{y\_reference}$  : Number of “y” entities of the reference design stage  
 $N_{x\_perdriver}$  : Maximum number of “x” entities per driver.  
 $N_{x\text{driver}}$  : Total number of “x” entity drivers  
 $N_{x\text{driver\_ref}}$  : Total number of reference design “x” entity drivers.  
 $N_{y\_perdriver}$  : Maximum number of “y” entities per driver  
 $N_{y\text{driver}}$  : Total number of “y” entity drivers  
 $N_{y\text{driver\_ref}}$  : Total number of reference design “y” entity drivers.  
 $\lambda$  : Fraction of reference leakage ( $P_{\text{stageleakage\_reference}}$ ) that is fixed (from auxiliary circuits)  
 $SP_f$  : Leakage signal probability factor of the stage node

## 2.5 Dynamic Power Model

### 2.5.1 Stage Dynamic Power ( Pstagedynamic )

The dynamic power of a stage is a function of the stage capacitance (C), voltage (V), activity (AF) and frequency (F).

$$P_{\text{dynamic}} = C \times AF \times V^2 \times f \quad (14)$$

The effects of voltage and frequency on power estimation are captured by the empirical stage power of the reference design. The activity factor and capacitance are the factors that will therefore determine the stage power relative to the reference power. We categorize the capacitance of a stage into three components as shown in Figure 3:

Repeated Capacitance ( $\Phi_{nr}$ ) – This is the fraction of the stage capacitance that is an instantiated multiple of the reference and changes with number of “x” entities. Non-repeated capacitance ( $\Phi_r$ ) – This is the fraction of the stage capacitance that is not directly dependent on the number of “x” entities but indirectly affected by device resizing as a result of change in the number of “x” entities.

Overhead Capacitance ( $\Phi_{ov}$ ) – This is the fraction of the stage capacitance that is not impacted by change in number of entities. This is usually a fixed cap from routing overhead and fixed logic associated with the stage. We capture these components individually and analytically model the impact of changes in number of bits, entries, and driver sizing on each of the components.

### 2.5.2 Segment Dynamic Power :

The dynamic power of a stage is modeled by the stage segment count relative to the reference design. The stage segment dynamic power is modeled as:

$$P_{\text{segmentdynamic}} = \frac{AF_{\text{estimate}}}{AF_{\text{reference}}} \times P_{\text{dynamicpersegment\_ref}} \times N_{x\text{stagesegment}} \times N_{y\text{stagesegment}} \quad (15)$$

$$N_{x\text{stagesegment}} = \Phi_r \times N_{rpt} \times \partial_{\text{effectsize\_rpt}} + \Phi_{nr} \times \partial_{\text{effectsize\_nr}} + \Phi_{ov} \quad (16)$$

$$N_{y\text{stagesegment}} = \left( \frac{N_{y\text{seg}}}{N_{y\text{seg\_ref}}} \right)^{(1-G_y)} \quad (17)$$

$$N_{y\text{seg}} = \text{Ceil} \left[ \frac{N_{y\_estimate}}{N_{y\_persegment}} \right] \quad N_{y\text{seg\_ref}} = \text{Ceil} \left[ \frac{N_{y\_reference}}{N_{y\_persegment}} \right]$$

$$P_{\text{dynamicpersegment\_ref}} = \frac{P_{\text{stagedynamic\_reference}}}{\left( \text{Ceil} \left[ N_{x\text{segment\_ref}} \right] \times (\Phi_{ov} + \Phi_{nr}) + N_{x\text{segment\_ref}} \times \Phi_r \right)^{(1-G_x)}} \quad (18)$$

$$\text{where } N_{x\text{segment\_ref}} = \frac{N_{xref}}{\min(N_{xref}, N_{xperwireseg})}$$

$\Phi_r, \Phi_{ov}, \Phi_{nr}$  are the reference design cap components.  $P_{\text{stagedynamic\_reference}}$  : Fraction of the segment to be estimated.  $N_{y\_persegment}$ : Reference design empirical stage dynamic power  $N_{y\_seg}$  : Maximum number of “y” entities per segment for the stage.  $N_{y\_seg\_ref}$ : Number of “y” entities per stage segment

$\partial_{effectsize\_rpt}$  : Total number of reference design “y” entity drivers.

$\partial_{effectsize\_nr}$  : Delay effect of repeated cap

$G_x, G_y$  : Indicates gating of the reference design “x” and “y” entity segments respectively. (Gated), (not Gated).

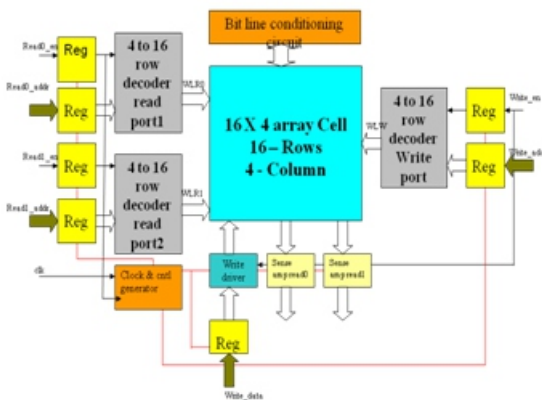
### III SRAM CELL BASED REGISTER FILE:

Design, schematic entry, Layout and Verification of 16 entry 4 bit register file with two read ports and write port



**Fig 4 Top level view of three port SRAM register file**

The block diagram of SRAM register file is shown below. The Major sub-components of the are described below



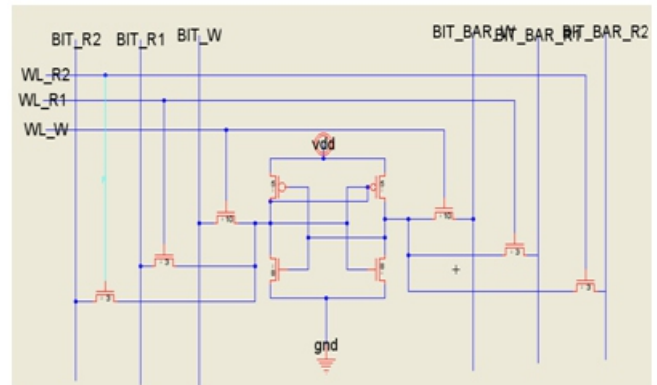
**Fig 5 Block Diagram of SRAM Register**

. Internal circuits to realize the subcomponents The following internal circuits are will be used to realize the components required for building the RW/M

- Array of Bit Cells
- Sense Amplifiers ( For prompt reading )
- Write Driver
- Bit Conditioning Circuit
- Decoder circuits ( two for read ports and one for write port )

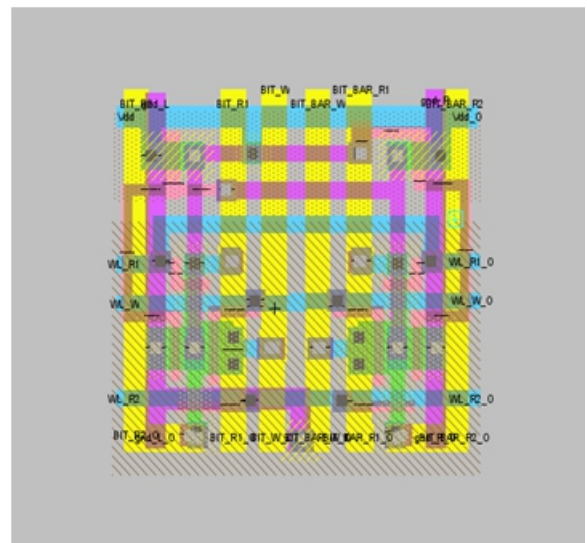
### IV IMPLEMENTATION OF SRAM BIT CELL USING 10 TRANSISTORS:

The SRAM cell stores a bit of data using six transistors. A Ten Transistor (10T) SRAM bit cell is as shown in Figure 6.



**Fig 6.Ten transistor BIT cell**

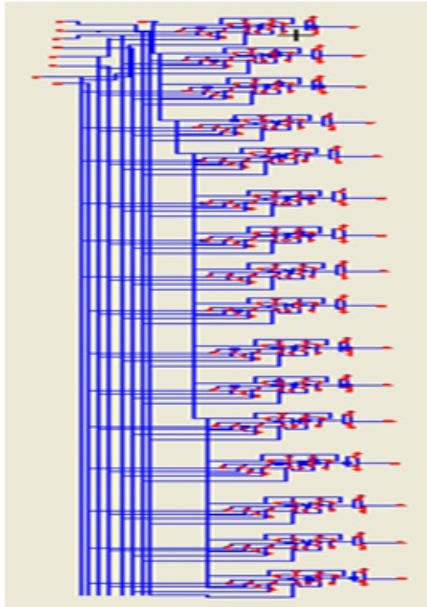
The layout is given by



**Fig 7 Layout of Ten transistor BIT Cell**

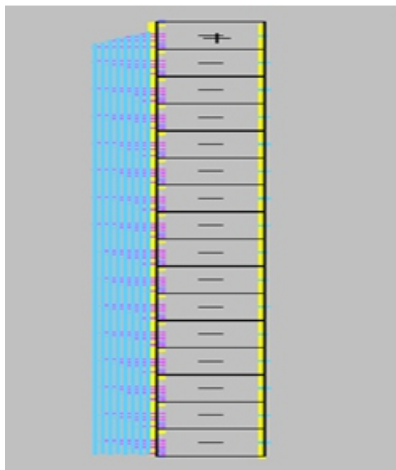
**4.1 Main Module Of 4 to 16 Decoder:**

The Schematic is given by



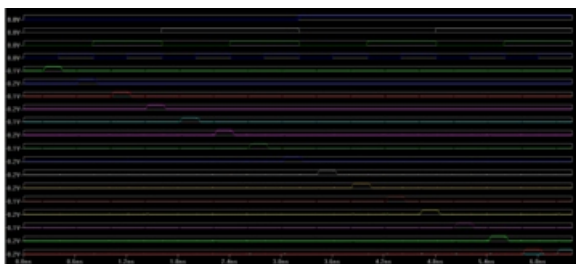
**Fig 8 Schematic of 4 to 16 decoder**

The Layout is given by

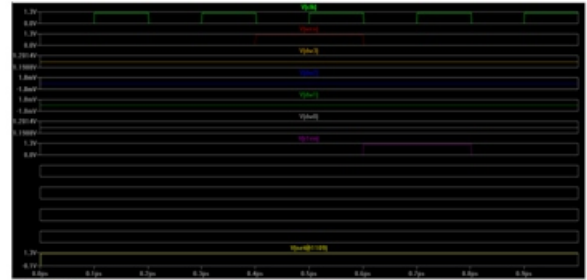


**Fig 9 Layout of 4 to 16 decoder**

**V RESULTS AND ANALYSIS:**



**Fig.10 Simulation of Decoder with 16 outputs for all combination of inputs**



**Fig.11.Simulation of SRAM register file using 10 transistors read operation followed by write operation.**

**5. CONCLUSION :**

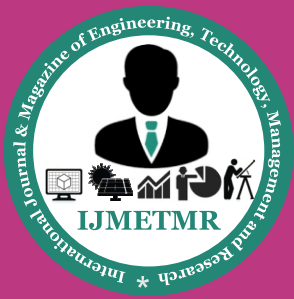
We present a reusable hybrid model in which the analytical formulas remain unchanged for all topologies while incorporating real design choices such as segmentation, gating, and timing impacts; a combination not in previous RF power models. The model is adaptable to other SRAM array structures by using a reference design and decomposing it into component stages, each stage characteristics captured independently by the reference design’s empirical data.

It allows any representative reference design to be used in estimation and exploration. Changes in topology and/or process require only the empirical data of a single reference to be updated. A typical modern microprocessor has a large number of unique RFs [2], most of which are manually designed and cannot be compiled, making design exploration across various combinations of bits, entries, gating, and segmentation intractable.

Using our model, individual unique RFs can be explored using their respective single reference designs without requiring large number of samples as needed for a curve fit approach. Our model is not tied to any specific technology or design style as we do not model the process, device level, or design environment dependent base values but rely on a reference design empirical data to capture these specifics.

**6. REFERENCES :**

[1] N. Kurd et al., “A Family of 32nm IA Processors”, IEEE Journal of Solid-State Circuits, 2011, vol 46, pp 119-130.



- [2] K. Anshumali et al. "Circuit And Process Innovations to Enable High-Performance, and Power and Area Efficiency on the Nehalem and Westmere Family of Intel processors" Intel Technology Journal, 2010, vol 14, pp 104-127 .
- [3] N. Vijaykrishnan, et al. "Energy-driven integrated hardware-software optimizations using SimplePower", ISCA 2000.
- [4] D. Brooks, et al, "Wattch: A Framework for Architectural-Level Power Analysis and Optimization," ISCA 2000.
- [5] Xuemei Zhao, et al: Design and Realization of a Low Power Register File Using Energy Model, PATMOS 2002: pp. 268-277.
- [6] Minh Q. Do, et al, "Parameterizable Architecture-Level SRAM Power Model Using Circuit-Simulation Backend for Leakage Calibration," pp.557-563, ISQED 2006.
- [7] S. L. Coumeri, D. E. Thomas Jr, "Memory Modeling for System Synthesis", IEEE Transactions on VLSI, June 2000.
- [8] Xiaoyao Liang, Kerem Turgay, David Brooks, "Architectural Power Models for SRAM and CAM Structures Based on Hybrid Analytical/Empirical Techniques", IC-CAD 2007.
- [9] S. Thoziyoor, et al., "CACTI 5.1," HP Technical Report,2008,.