Accelergy: An Architecture-Level Energy Estimation Methodology for Accelerator Designs

Yannan Nellie Wu\textsuperscript{1}, Joel S. Emer\textsuperscript{1,2}, Vivienne Sze\textsuperscript{1}

\textsuperscript{1} MIT  \quad \textsuperscript{2} NVIDIA
Accelergy Overview

• An architecture-level energy estimator
• Flexibly characterizes various basic building blocks of different technologies
• Succinctly models diverse and complicated designs
• Improves estimation accuracy via fine-grained classification of operations
• Achieves 95% accuracy in evaluating a deep neural network (DNN) accelerator – Eyeriss [ISSCC 2016]
Energy Consumption Concerns

Data and computation-intensive applications are power hungry

- Object Detection
- Database Processing
- DNN Accelerator
- Database Accelerator

We must quickly evaluate energy efficiency of arbitrary potential designs in the large design space
Energy Estimation and Design Exploration

- global buffer (GLB)
- buffer
- MAC
- PE*
- component
- abstract hierarchy

Arch. Description
Energy Estimation and Design Exploration

- Physical-Level Energy Estimator (Synopsys Prime Power, Cadence Joules)

Synthesize the design, place standard cells, and route the wires

Develop the register transfer level (RTL) details

Requires physical layout of the design
Energy Estimation and Design Exploration

• Physical-Level Energy Estimator (Synopsys Prime Power, Cadence Joules)

Requires physical layout of the design
Slow design space exploration
Accelergy Overview

• An architecture-level energy estimator
• Flexibly characterizes various basic building blocks in the design
• Succinctly models diverse and complicated designs
• Improves estimation accuracy via fine-grained classification of operations
• Achieves 95% accuracy in evaluating a deep neural network (DNN) accelerator – Eyeriss [ISSCC 2016]
Energy Estimation and Design Exploration

• Architecture-Level Energy Estimators

Only requires architecture-level design
Fast design space exploration
Existing Architecture-Level Energy Estimators

- **Design-Specific Accelerator Estimators: Aladdin** [ISCA2014], **fixed-cost** [Asilomar2017]

![Architecture Description](image)

- **Energy Estimator**
  - Description with **primitive components** (basic building blocks)
  - GLB, buffer, MAC, PE
Existing Architecture-Level Energy Estimators

- **Design-Specific Accelerator Estimators: Aladdin** [ISCA2014], **fixed-cost** [Asilomar2017]

---

**Architecture Description**

- GLB
- buffer
- MAC

**Energy Reference Table (ERT)**

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Action</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>access()</td>
<td>100pJ</td>
</tr>
<tr>
<td>buffer</td>
<td>access()</td>
<td>10pJ</td>
</tr>
<tr>
<td>MAC</td>
<td>compute()</td>
<td>5pJ</td>
</tr>
</tbody>
</table>
Existing Architecture-Level Energy Estimators

• Design-Specific Accelerator Estimators: Aladdin[ISCA2014], fixed-cost[Asilomar2017]

Architecture Description

Energy Estimator

Energy Reference Table (ERT)

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Action</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>access()</td>
<td>100pJ</td>
</tr>
<tr>
<td>buffer</td>
<td>access()</td>
<td>10pJ</td>
</tr>
<tr>
<td>MAC</td>
<td>compute()</td>
<td>5pJ</td>
</tr>
</tbody>
</table>

Action Counts

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Action</th>
<th>Counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>access()</td>
<td>10</td>
</tr>
<tr>
<td>buffer</td>
<td>access()</td>
<td>800</td>
</tr>
<tr>
<td>MAC</td>
<td>compute()</td>
<td>400</td>
</tr>
</tbody>
</table>
Existing Architecture-Level Energy Estimators

- Design-Specific Accelerator Estimators: Aladdin [ISCA2014], fixed-cost [Asilomar2017]

**Architecture Description**

- GLB
- Buffer
- MAC

**Energy Estimator**

**Energy Reference Table (ERT)**

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Action</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>access()</td>
<td>100pJ</td>
</tr>
<tr>
<td>Buffer</td>
<td>access()</td>
<td>10pJ</td>
</tr>
<tr>
<td>MAC</td>
<td>compute()</td>
<td>5pJ</td>
</tr>
</tbody>
</table>

**Energy Calculations**

**Action Counts**

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Action</th>
<th>Counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>access()</td>
<td>10</td>
</tr>
<tr>
<td>Buffer</td>
<td>access()</td>
<td>800</td>
</tr>
<tr>
<td>MAC</td>
<td>compute()</td>
<td>400</td>
</tr>
</tbody>
</table>

**Energy Estimations**

<table>
<thead>
<tr>
<th>Name</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>1000pJ</td>
</tr>
<tr>
<td>Buffer</td>
<td>8000pJ</td>
</tr>
<tr>
<td>MAC</td>
<td>2000pJ</td>
</tr>
</tbody>
</table>
Existing Architecture-Level Energy Estimators

- **Design-Specific Accelerator Estimators: Aladdin** [ISCA2014], **fixed-cost** [Asilomar2017]

### Architecture Description

- **GLB’**
- **buffer**
- **MAC**

### Action Counts

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Action</th>
<th>Counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB’</td>
<td>access()</td>
<td>10</td>
</tr>
<tr>
<td>buffer</td>
<td>access()</td>
<td>800</td>
</tr>
<tr>
<td>MAC</td>
<td>compute()</td>
<td>400</td>
</tr>
</tbody>
</table>

### Energy Reference Table (ERT)

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Action</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>access()</td>
<td>100pJ</td>
</tr>
<tr>
<td>buffer</td>
<td>access()</td>
<td>10pJ</td>
</tr>
<tr>
<td>MAC</td>
<td>compute()</td>
<td>5pJ</td>
</tr>
</tbody>
</table>

### Energy Calculator

Not generalizable to other designs
Accelergy Overview

• An architecture-level energy estimator

• **Flexibly characterizes various primitive components of different technologies**

• Succinctly models diverse and complicated designs

• Improves estimation accuracy via fine-grained classification of operations

• Achieves 95% accuracy in evaluating a deep neural network (DNN) accelerator – Eyeriss [ISSCC 2016]
Accelergy: Flexibly Model Various Primitive Components

Architecture Description

GLB

SRAM

MAC

buffer

ERT Generator

Primitive Component Library

CACTI Estimation Plug-in

40nm Estimation Plug-in

...
Accelergy: Flexibly Model Various Primitive Components

Architecture Description

GLB
buffer
MAC

SRAM

ERT Generator

Primitive Component Library

ERT (in progress)

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Action</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>access()</td>
<td>100pJ</td>
</tr>
</tbody>
</table>

SRAM type has associated action “access”
Accelergy: Flexibly Model Various Primitive Components

Architecture Description

- GLB
- Buffer
- MAC

Accelergy

- ERT Generator
- Primitive Component Library

ERT

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Action</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>access()</td>
<td>100pJ</td>
</tr>
<tr>
<td>Buffer</td>
<td>access()</td>
<td>10pJ</td>
</tr>
<tr>
<td>MAC</td>
<td>compute()</td>
<td>5pJ</td>
</tr>
</tbody>
</table>

CACTI Estimation Plug-in

40nm Estimation Plug-in
Accelergy: Flexibly Model Various Primitive Components

**Architecture Description**
- GLB
- buffer
- MAC

**ERT Generator**
- Primitive Component Library

**Energy Calculator**
- CACTI Estimation Plug-in
- 40nm Estimation Plug-in

**ERT**
<table>
<thead>
<tr>
<th>Comp.</th>
<th>Action</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>access()</td>
<td>100pJ</td>
</tr>
<tr>
<td>buffer</td>
<td>access()</td>
<td>10pJ</td>
</tr>
<tr>
<td>MAC</td>
<td>compute()</td>
<td>5pJ</td>
</tr>
</tbody>
</table>

...
Accelergy: Flexibly Model Various Primitive Components

Architecture Description

ERT Generator

Primitive Component Library

Energy Calculator

GLB
buffer
MAC

ERT

Comp. | Action | Energy
--- | --- | ---
GLB | access() | 100pJ
buffer | access() | 10pJ
MAC | compute() | 5pJ

Action Counts

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Action</th>
<th>Counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>access()</td>
<td>10</td>
</tr>
<tr>
<td>buffer</td>
<td>access()</td>
<td>800</td>
</tr>
<tr>
<td>MAC</td>
<td>compute()</td>
<td>400</td>
</tr>
</tbody>
</table>

Name | Energy
--- | ---
GLB | 1000pJ
Buffer | 8000pJ
MAC | 2000pJ

CACTI Estimation Plug-in

40nm Estimation Plug-in

Energy Estimates
Use energy estimation plug-ins to characterize primitive components

- CACTI Estimation Plug-in
- 40nm Estimation Plug-in
- Proprietary plug-ins
- Emerging technology plug-ins

Traditional open-source plug-ins*

*available at http://accelergy.mit.edu
Accelergy: Flexibly Model Various Primitive Components

Use energy estimation plug-ins to characterize primitive components

Detailed plug-in interface in open-source repo

Traditional open-source plug-ins*

CACTI Estimation Plug-in

Emerging technology plug-ins

*available at http://accelergy.mit.edu

Detailed plug-in interface in open-source repo

NVSIM [TCAD 2012]
Modeling Complicated Designs

- Practical architecture designs involve much more details
  - Example: storage units with local address generators (AGs)

- AG_SRAM is an abstract hierarchy
- Buffer is of SRAM type
- AGs is of counter type

Example: storage units with local address generators (AGs)
Modeling Complicated Designs

• Practical architecture designs involve much more details
  – Example: storage units with local address generators (AG)

Let’s construct a more practical design!
Modeling Complicated Designs

- Practical architecture designs involve much more details
  - Example: storage units with local address generators (AG)

*Let’s construct a more practical design!*
Modeling Complicated Designs

- Architecture description is tedious
- Hard to make modifications

• Action counts are even more tedious
• Small modification requires new action counts

<table>
<thead>
<tr>
<th>Name</th>
<th>Action</th>
<th>Counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>PE[0]. AG[0]</td>
<td>count</td>
<td>50</td>
</tr>
<tr>
<td>PE[0]. AG[1]</td>
<td>count</td>
<td>50</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Existing Work - Modeling Complicated Designs

• Existing work that aims to succinctly model complicated architectures
  – Wattch [ISCA 2000], McPAT [MICRO 2009]

**CPU-Centric Architecture Model**

Use a fixed set of **compound components** to represent the architecture

Components that can be decomposed into lower level components
Existing Work - Modeling Complicated Designs

- Existing work that aims to succinctly model complicated architectures
  
  - Wattch [ISCA 2000], McPAT [MICRO 2009]

**CPU-Centric Architecture Model**

The fixed set of compound components is not sufficient to describe arbitrary accelerator designs.
Accelergy Overview

• An architecture-level energy estimator
• Flexibly characterizes various primitive components of different technologies
• Succinctly models diverse and complicated designs
• Improves estimation accuracy via fine-grained classification of operations
• Achieves 95% accuracy in evaluating a deep neural network (DNN) accelerator – Eyeriss [ISSCC 2016]
Accelergy: Succinctly Model Arbitrary Architecture

AG_SRAM is an user-defined compound component

Architecture described with compound components and primitive components
Accelergy: Succinctly Model Arbitrary Action Counts

Tedious action counts in terms of primitive component actions
Accelergy: Succinctly Model Arbitrary Action Counts

User-defined compound actions

- AG_SRAM.read()
- AGs[0].count()
- buffer.read()
Accelergy: Succinctly Model Arbitrary Action Counts

Succinct action counts with compound component actions

User-defined compound actions

Action Counts

<table>
<thead>
<tr>
<th>Name</th>
<th>Action</th>
<th>Counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>read()</td>
<td>50</td>
</tr>
<tr>
<td>GLB</td>
<td>write()</td>
<td>20</td>
</tr>
</tbody>
</table>
Accelergy: Succinctly Model Complex Designs

Architecture Description

Compound Component Description

Accelergy

ERT Generator

Primitive Component Library

Energy Calculator

ERT

<table>
<thead>
<tr>
<th>Comp.</th>
<th>Actions</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>read(), ...</td>
<td>120pJ,</td>
</tr>
<tr>
<td>PE[0].buffer</td>
<td>read(), ...</td>
<td>12pJ,</td>
</tr>
<tr>
<td>PE[0].MAC</td>
<td>compute(), ...</td>
<td>5pJ,</td>
</tr>
</tbody>
</table>
Accelergy: Succinctly Model Complex Designs

---

**Architecture Description**

- GLB
  - AG_SRAM
- MAC FIFO
- MAC FIFO
- MAC FIFO
- PE
  - buffer AG_SRAM
  - buffer AG_SRAM
  - buffer AG_SRAM

**Compound Component Description**

- AG[0]
  - read address
  - counter
- AG[1]
  - write address
- buffer MAC FIFO

**Accelergy**

- ERT Generator
  - Primitive Component Library
  - Energy Calculator

**Energy Calculations**

- CACTI Estimation Plug-in
- 40nm Estimation Plug-in

---

**ERT**

<table>
<thead>
<tr>
<th>Component</th>
<th>Action</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>read()</td>
<td>120pJ,</td>
</tr>
<tr>
<td>PE[0].buffer</td>
<td>read()</td>
<td>12pJ,</td>
</tr>
<tr>
<td>PE[0].MAC</td>
<td>compute()</td>
<td>5pJ,</td>
</tr>
</tbody>
</table>

**Action Counts**

<table>
<thead>
<tr>
<th>Name</th>
<th>Action</th>
<th>Counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>read()</td>
<td>50</td>
</tr>
<tr>
<td>PE[0].buffer</td>
<td>read()</td>
<td>60</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>
Additional Challenge: Inaccurate Modeling of Energy/Action

- Existing architecture-level energy estimators only model coarse action types.

<table>
<thead>
<tr>
<th>Component</th>
<th>Action</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLB</td>
<td>access()</td>
<td>100pJ</td>
</tr>
<tr>
<td>Buffer</td>
<td>access()</td>
<td>10pJ</td>
</tr>
<tr>
<td>ALU</td>
<td>compute()</td>
<td>5pJ</td>
</tr>
</tbody>
</table>

Coarse-grained Actions

Coarse-grained estimations reduce estimation accuracies

Energy-Per-Actions of a Register File (normalized to idle)

- Random Read: 1.8
- Repeated Read: 1.0
- Random Write: 4.7 (~5x)
- Repeated Write: 2.1
- Constant Data Write: 2.4

Fine-grained Actions
Accelergy Overview

• An architecture-level energy estimator
• Flexibly characterizes various primitive components of different technologies
• Succinctly models diverse and complicated designs
• Improves estimation accuracy via fine-grained actions
• Achieves 95% accuracy in evaluating a deep neural network (DNN) accelerator – Eyeriss [ISSCC 2016]
Accelergy: Fine-grained Action Classification

• Accurate estimation with a primitive component library

Defines the fine-grained actions for each primitive component

Fine-grained memory action types

- Random Read
- Repeated Read
- Random Write
- Repeated Write
- Constant Data Write

Fine-grained multiplier action types

- Random Mult
- Reused Mult
- Gated Mult

~5x
~20x
Accelergy: Fine-grained Action Classification

• Accurate estimation with a primitive component library

Detailed methodology for generating fine-grained action types in paper
Accelergy Overview

• An architecture-level energy estimator
• Flexibly characterizes various primitive components of different technologies
• Succinctly models diverse and complicated designs
• Improves estimation accuracy via fine-grained actions
• Achieves 95% accuracy in evaluating a deep neural network (DNN) accelerator – Eyeriss [ISSCC 2016]
Energy Evaluations on Eyeriss

• Experimental Setup:
  – Workload: Alexnet weights & ImageNet input feature maps
  – Ground Truth: Energy obtained from post-layout simulations

Eyeriss Architecture

Ifmap = input feature map
Psum = partial sum
PE = processing element
*_spad = *_scratchpad
Energy Evaluations on Eyeriss

• Experimental Setup:
  – Workload: Alexnet weights & ImageNet input feature maps
  – Ground Truth: Energy obtained from post-layout simulations

Zero-gating optimization
If there is a 0 ifmap data
• Gate on reading the weights data => gated-read
• Gate on computing the MAC => gated-MAC
Total Energy and Coarse Energy Breakdown

- Total energy estimation is 95% accurate of the post-layout energy.
- Estimated relative breakdown of the important units in the design is within 8% of the post-layout energy.

*Total energy might not add up to exact 100.0% due to rounding*
• Comparisons with existing work: Aladdin and fixed-cost
PE Array Energy Breakdown

• Comparisons with existing work: Aladdin and fixed-cost

Not aware of the fine-grained actions related to zero-gating optimization
PE Array Energy Breakdown

- Comparisons with existing work: Aladdin and fixed-cost

Inaccurate energy characterization of components
• Comparisons with existing work: Aladdin and fixed-cost

Energy Breakdown of components inside a PE

- ground truth
- Accelergy
- Aladdin
- fixed-cost

Zero-gating action type not reflected
PE Energy Breakdown

- Comparisons with existing work: Aladdin and fixed-cost

Energy Breakdown of components inside a PE

- All local scratchpads share the same energy reference table
Conclusion

• Accelergy is an architecture-level energy estimator that
  – Accelerates accelerator design space exploration
  – Provides flexibility to
    • Describe a diverse range of accelerator designs
    • Support estimation of different technologies, e.g., CMOS, RRAM, optical
  – Achieves high accuracy energy estimations
    • 95% accurate for the Eyeriss accelerator

• Open-source code available at: http://accelergy.mit.edu

Acknowledgement: DARPA, Facebook, MIT Presidential Fellowship