Timeloop

Accelergy

Angshuman Parashar	NVIDIA
Yannan Nellie Wu	MIT
Po-An Tsai	NVIDIA
Vivienne Sze	MIT
Joel S. Emer	NVIDIA, MIT

ISPASS Tutorial

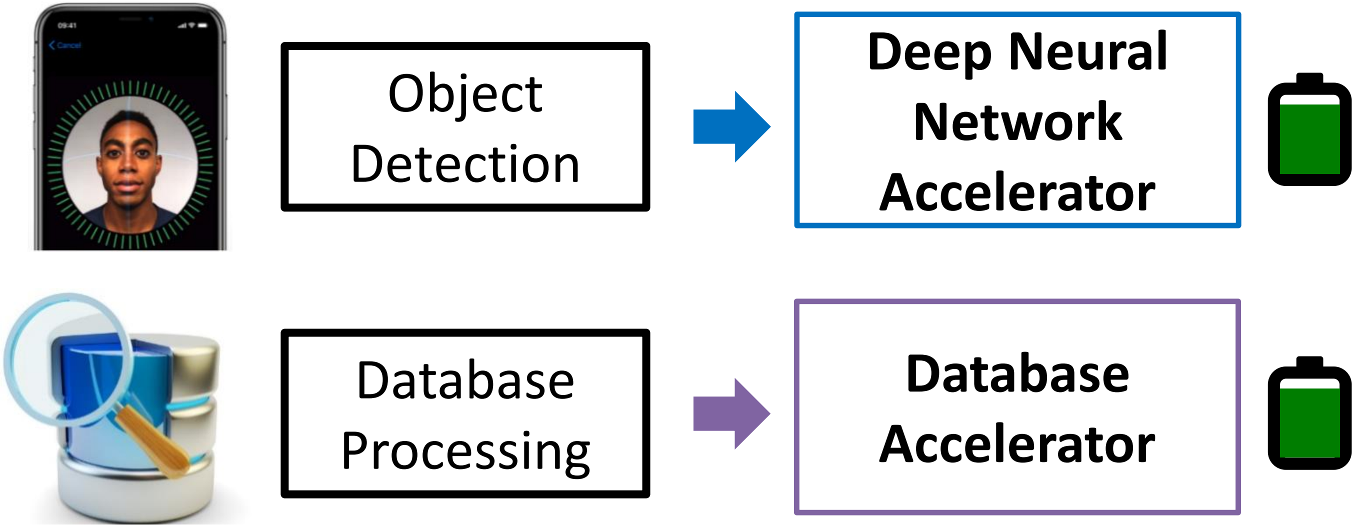
August 2020

Resources

- **Tutorial Related**
  - **Tutorial Website: http://accelergy.mit.edu/tutorial.html**
  - **Tutorial Docker: https://github.com/Accelergy-Project/timeloop-accelergy-tutorial**
    - **Various exercises and example designs and environment setup for the tools**
- **Other**
  - **Infrastructure Docker: https://github.com/Accelergy-Project/accelergy-timeloop-infrastructure**
    - **Pure environment setup for the tools without exercises and example designs**
  - **Open Source Tools**
    - **Accelergy: http://accelergy.mit.edu/**
    - **Timeloop: https://github.com/NVlabs/timeloop**
  - **Papers:**
    - A. Parashar, et al. "Timeloop: A systematic approach to DNN accelerator evaluation," ISPASS, 2019.
    - Y. N. Wu, V. Sze, J. S. Emer, “An Architecture-Level Energy and Area Estimator for Processing-In-Memory Accelerator Designs,” *ISPASS,* 2020.
    - Y. N. Wu, J. S. Emer, V. Sze, “Accelergy: An Architecture-Level Energy Estimation Methodology for Accelerator Designs,” *ICCAD*, 2019.

Domain-Specific Accelerators Improve Energy Efficiency

Data and computation-intensive applications are power hungry

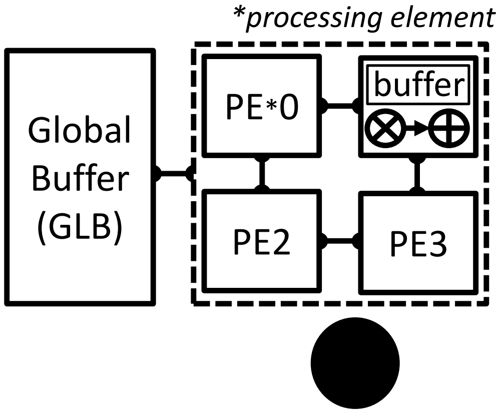


Object Detection → **Deep Neural Network Accelerator**

Database Processing → **Database Accelerator**

We must quickly evaluate energy efficiency of arbitrary potential designs in the large design space

From Architecture Blueprints to Physical Systems

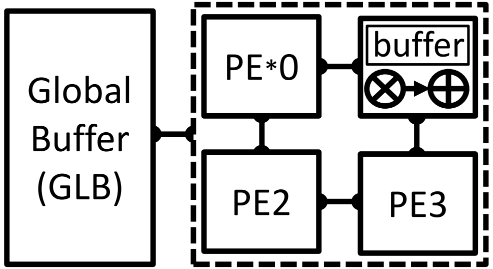


Architecture Stage

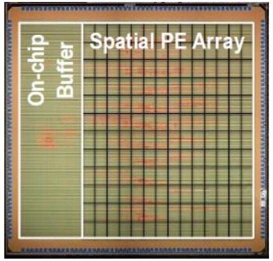
- How many levels in the memory hierarchy?
- How large are the memories at each level?
- How many PEs are there?
- What are the X and Y dimensions of the PE array?
- …

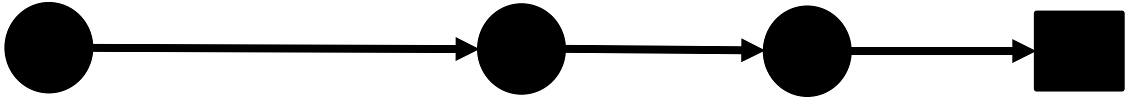
From Architecture Blueprints to Physical Systems

**processing element*



[Chen, ISSCC 2016]





Architecture Stage

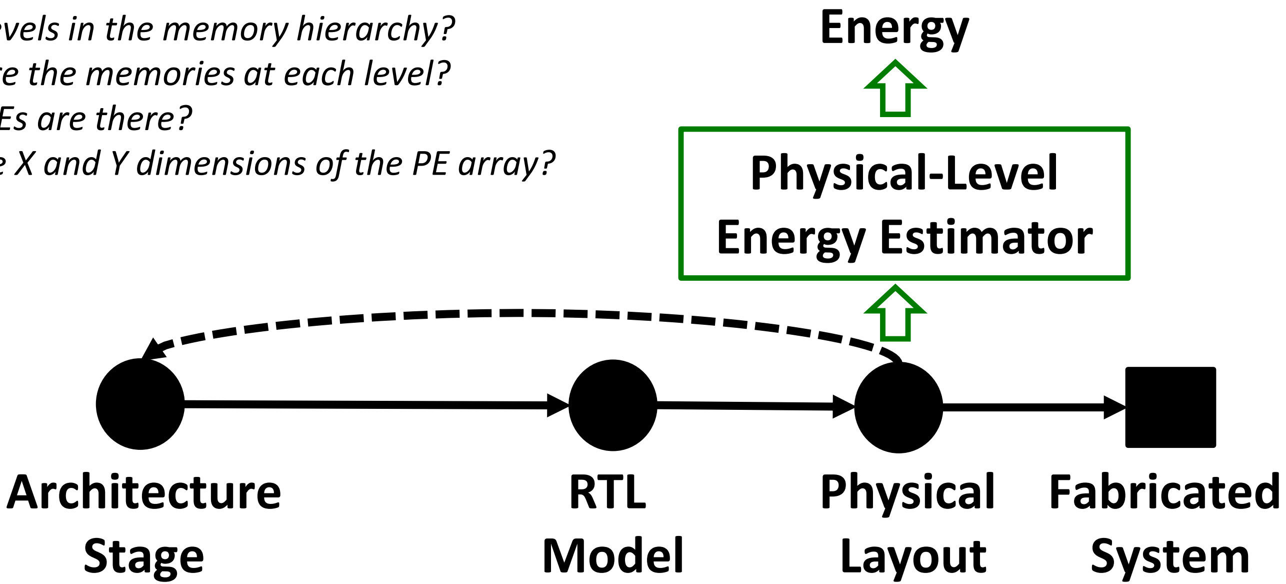
RTL Model

Physical Layout

Fabricated System

Physical-Level Energy Estimation and Design Exploration

- *How many levels in the memory hierarchy?*
- *How large are the memories at each level?*
- *How many PEs are there?*
- *What are the X and Y dimensions of the PE array?*
- …

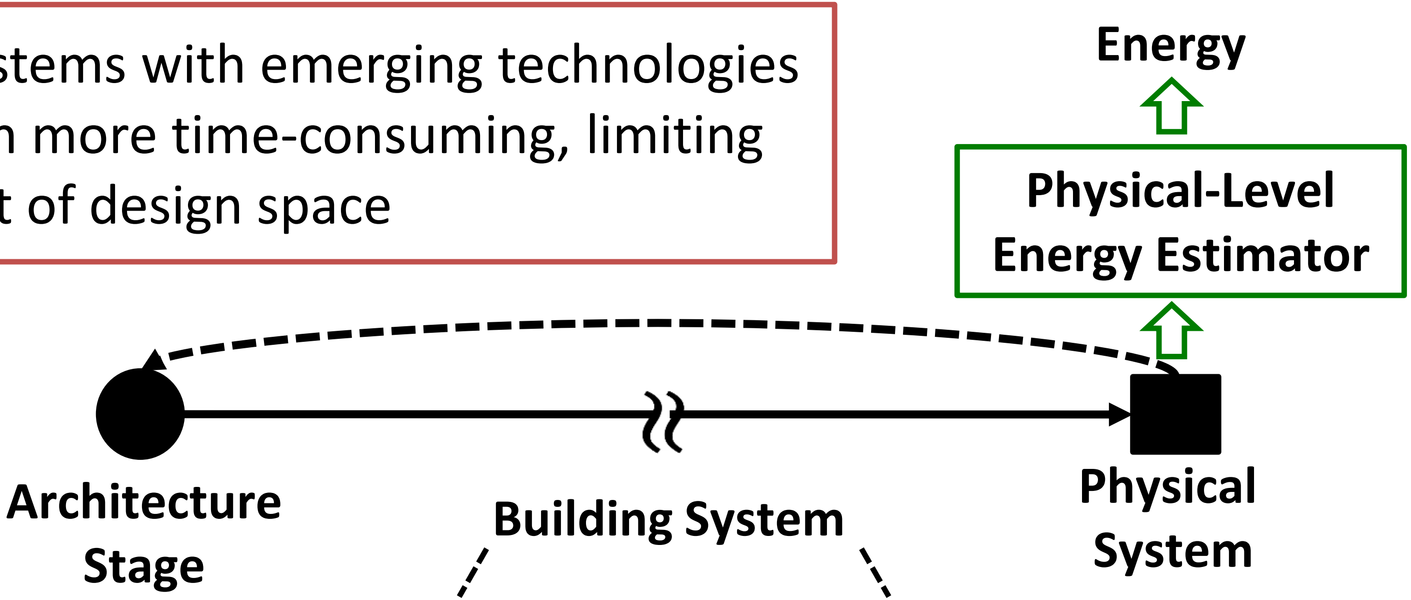


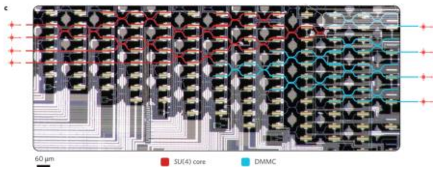
Slow design space exploration

- Long simulations on gate-level components
- Long turn-around time for each potential design

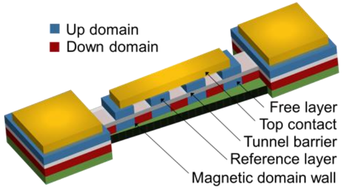
Physical-Level Energy Estimation and Design Exploration

Building systems with emerging technologies can be even more time-consuming, limiting the amount of design space





Optical Computation *[Nature Photonics 2017]*

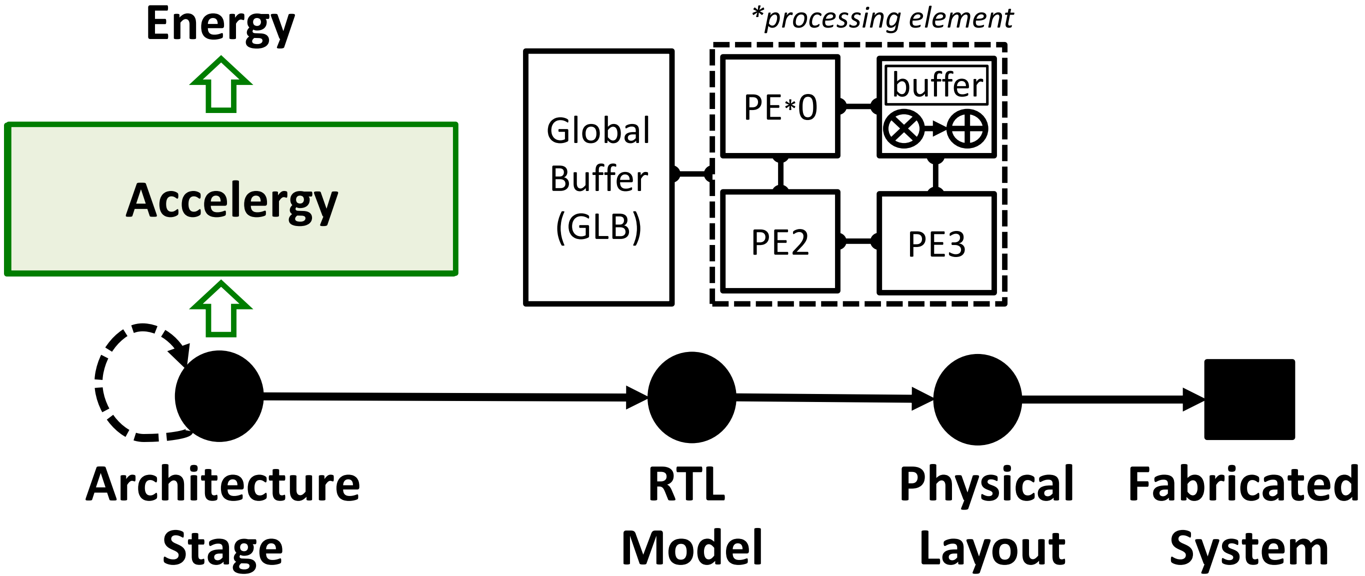


Non-volatile Memory Computation *[NANOARCH 2017]*

Accelergy Overview

- **Accelergy Infrastructure**
  - **Performs architecture-level estimations to enable rapid design space exploration**
  - Supports modeling of diverse architectures with various underlying technologies
  - Improves estimation accuracy by allowing fine-grained classification of components' runtime behaviors
  - Supports succinct modeling of complicated architectures
- Validation on various accelerator designs
  - 95% accurate on a conventional digital accelerator design
  - Modeling of processing in memory (PIM) based DNN accelerator designs

Architecture-Level Energy Estimation and Design Exploration

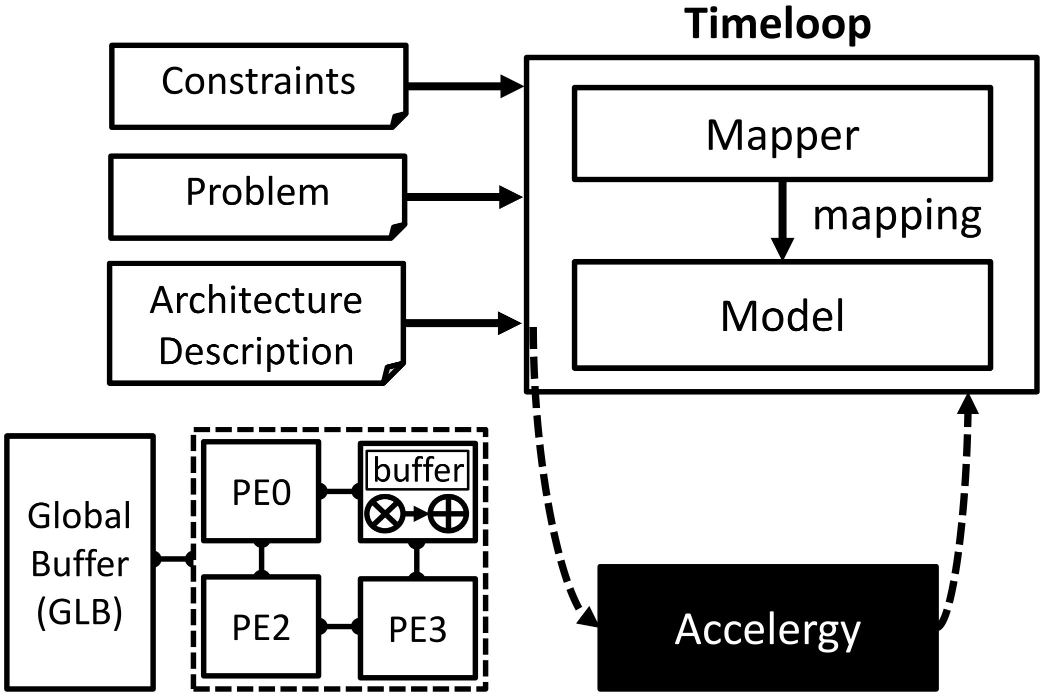


Fast design space exploration

- Short simulations on architecture-level components
- Short turn-around time for each potential design

Connect Back to Timeloop

Timeloop requires energy reference tables (ERTs) to evaluate the energy efficiency of a potential mapping



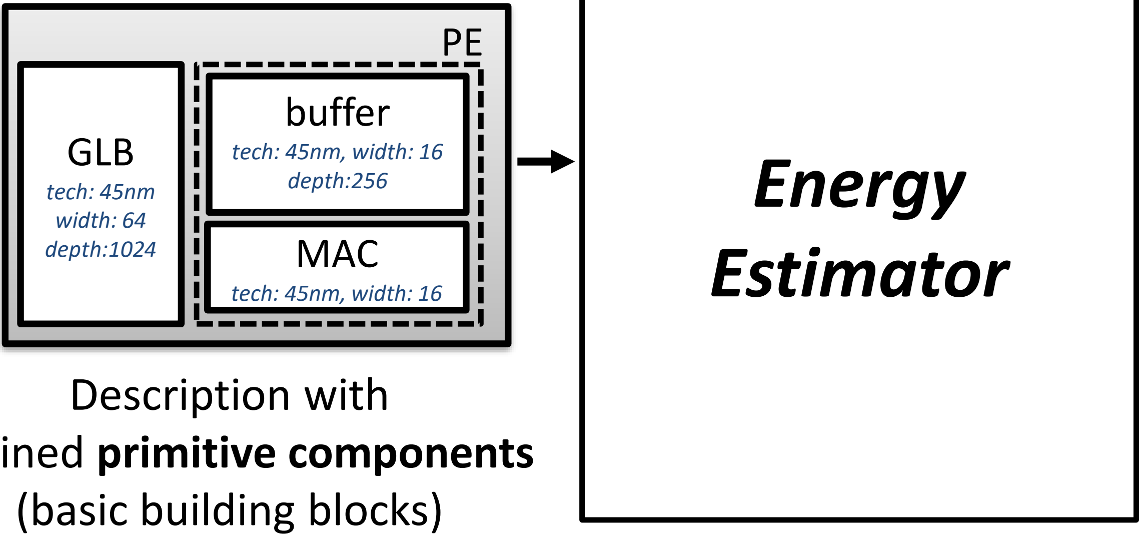
Energy Reference Table (ERT)

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

Existing Accelerator Estimators Lack Flexibility

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

Architecture Description

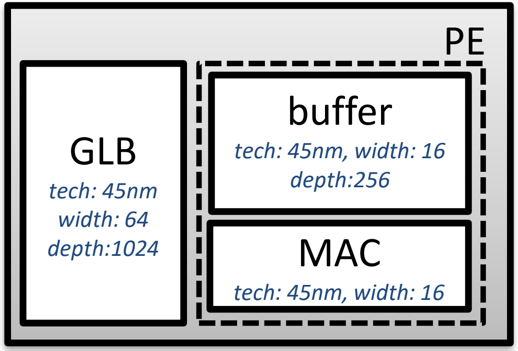


Description with
defined **primitive components**
(basic building blocks)

Existing Accelerator Estimators Lack Flexibility

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

Architecture Description



Energy Estimator

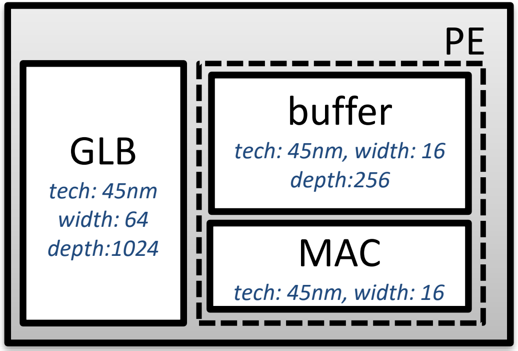
Energy Reference Table (ERT)

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

Existing Accelerator Estimators Lack Flexibility

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

Architecture Description



PE

GLB
tech: 45nm
width: 64
depth:1024

buffer
tech: 45nm, width: 16
depth:256

MAC
tech: 45nm, width: 16

Comp.	Action	Counts
GLB	access()	10
buffer	access()	800
MAC	compute()	400

Action Counts *Comes from a performance model (e.g., cycle accurate simulator)*

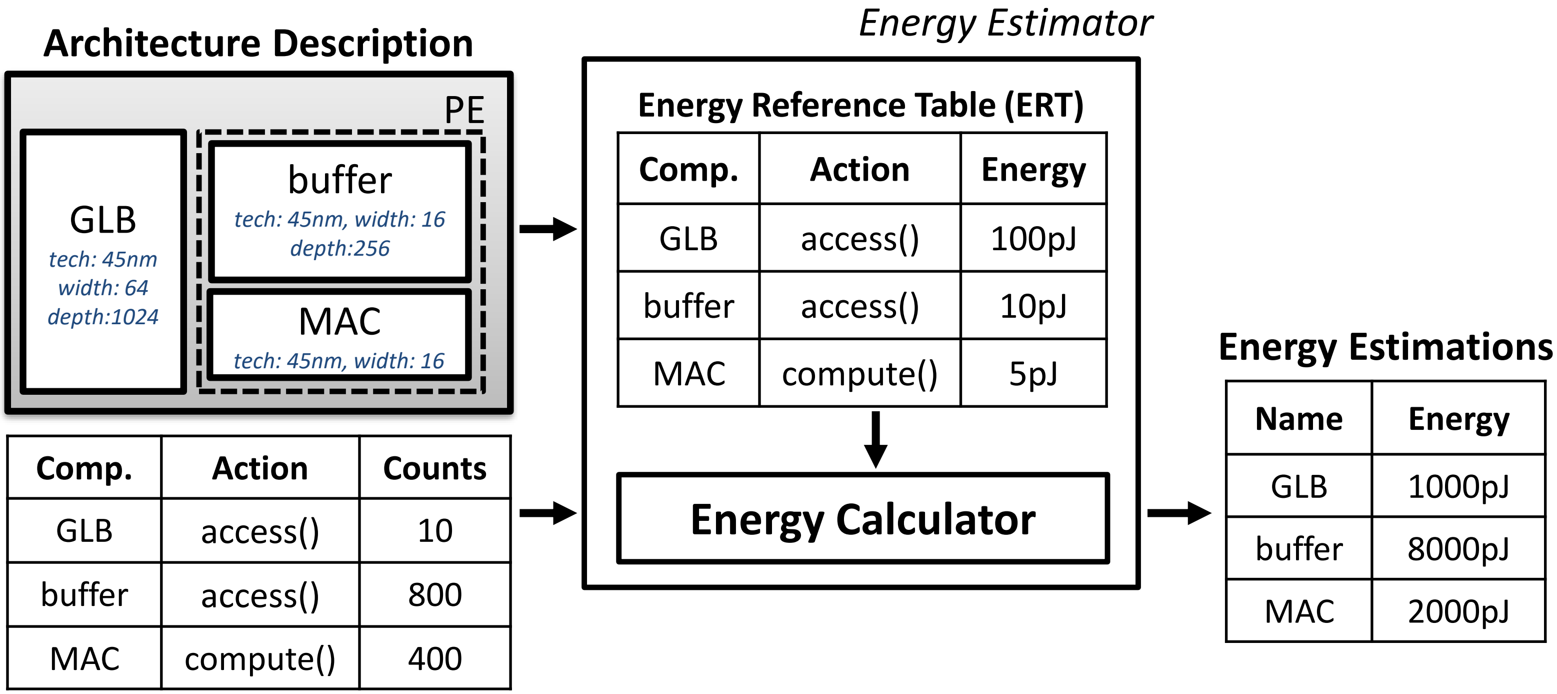
Energy Estimator

Energy Reference Table (ERT)

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

Existing Accelerator Estimators Lack Flexibility

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



Architecture Description

PE

GLB
tech: 45nm
width: 64
depth:1024

buffer
tech: 45nm, width: 16
depth:256

MAC
tech: 45nm, width: 16

Comp.	Action	Counts
GLB	access()	10
buffer	access()	800
MAC	compute()	400

Energy Estimator

Energy Reference Table (ERT)

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

Energy Calculator

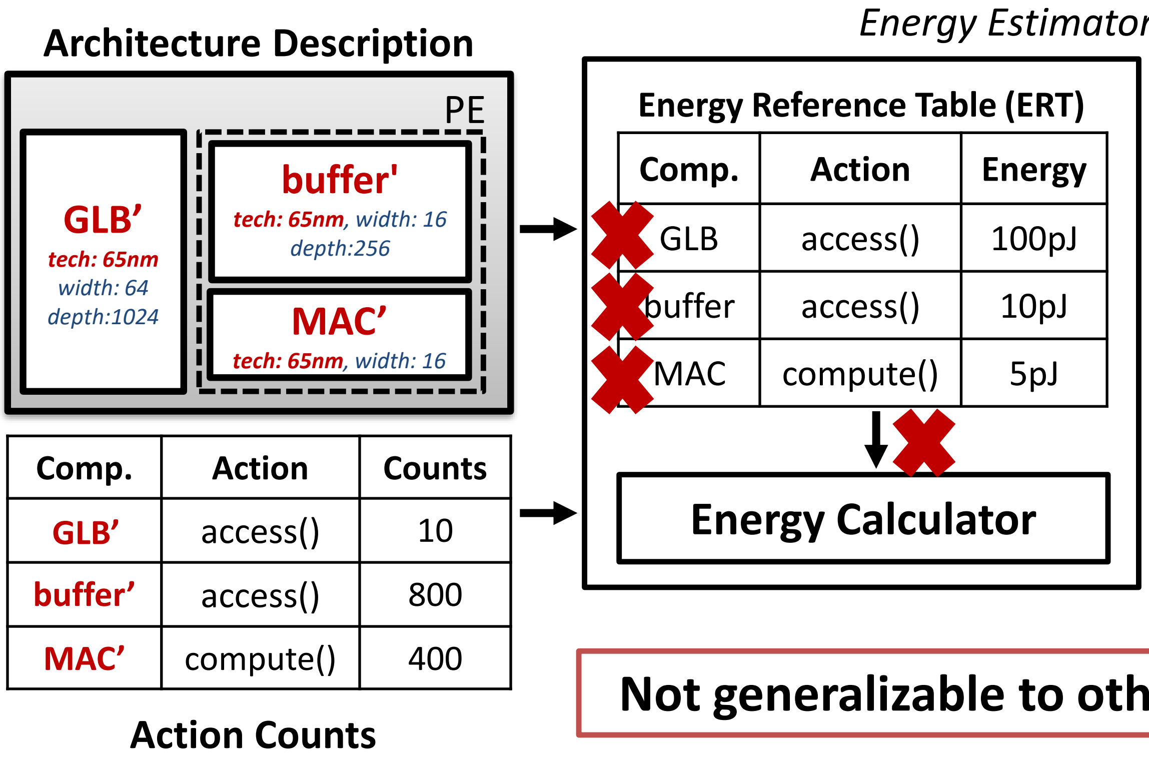
Energy Estimations

Name	Energy
GLB	1000pJ
buffer	8000pJ
MAC	2000pJ

Action Counts *Comes from a performance model*
(e.g., cycle accurate simulator)

Existing Accelerator Estimators Lack Flexibility

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



Architecture Description

Comp.	Action	Counts
GLB’	access()	10
buffer’	access()	800
MAC’	compute()	400

Action Counts

Energy Estimator

Energy Reference Table (ERT)

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

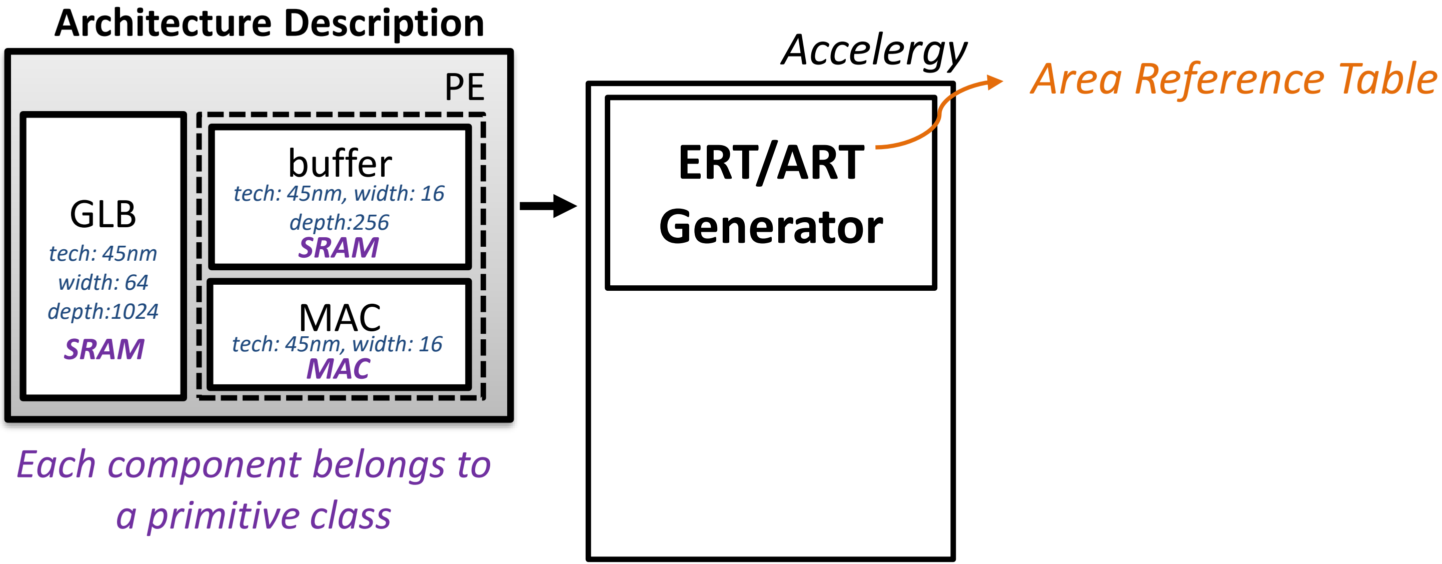
Energy Calculator

Not generalizable to other designs

Accelergy Overview

- **Accelergy Infrastructure**
  - Performs architecture-level estimations to enable rapid design space exploration
  - **Supports modeling of diverse architectures with various underlying technologies**
  - Improves estimation accuracy by allowing fine-grained classification of components runtime behaviors
  - Supports succinct modeling of complicated architectures
- Validation on various accelerator designs
  - 95% accurate on a conventional digital accelerator design
  - Modeling of processing in memory (PIM) based DNN accelerator designs

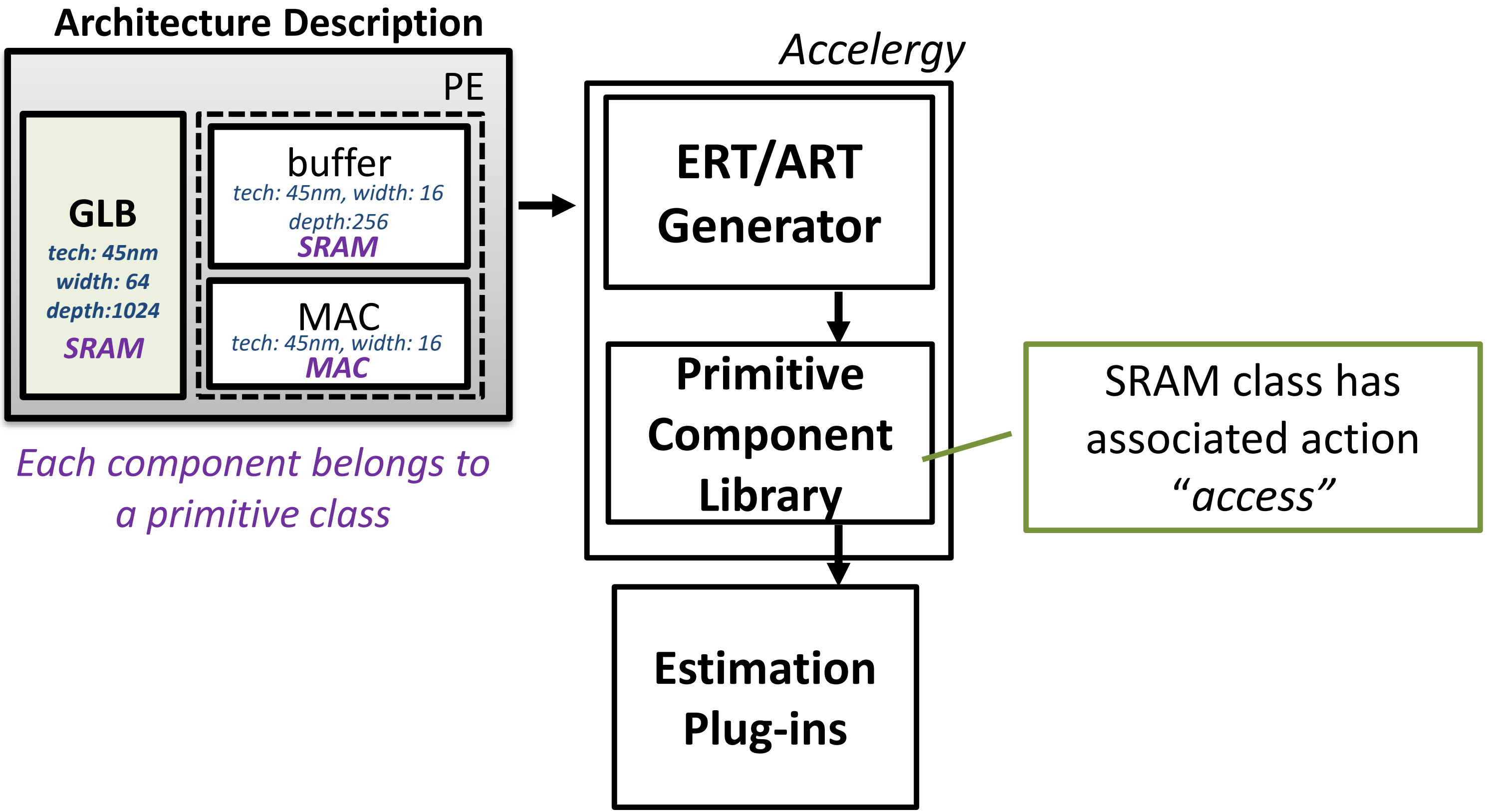
Accelergy: Flexibly Model Various Primitive Components



Each component belongs to a primitive class

Available at http://accelergy.mit.edu/

Accelergy: Flexibly Model Various Primitive Components

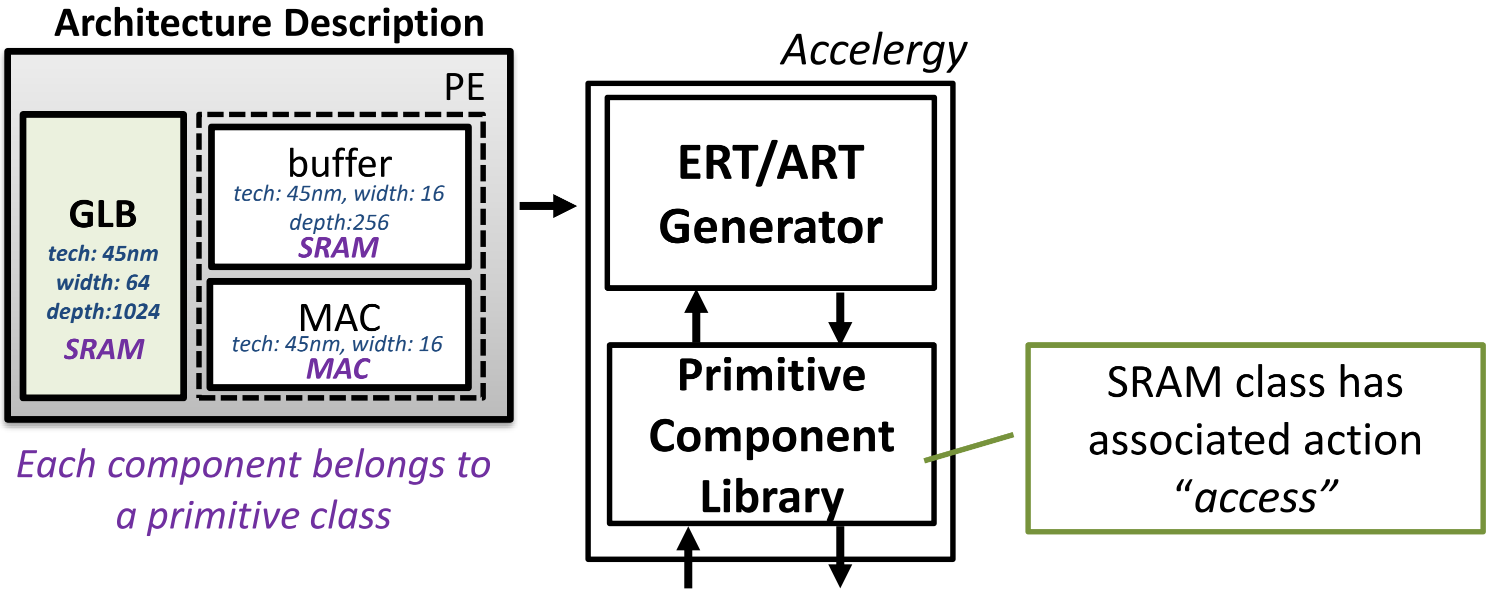


Each component belongs to a primitive class

SRAM class has associated action *"access"*

Available at http://accelergy.mit.edu/

Accelergy: Flexibly Model Various Primitive Components



Each component belongs to a primitive class

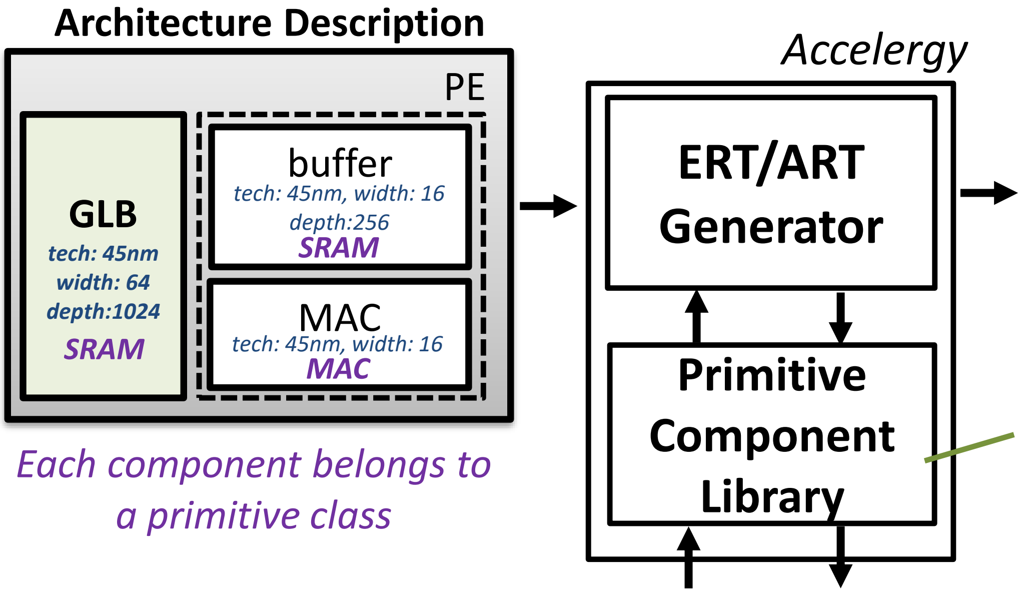
SRAM class has associated action *“access”*

Simple Example Estimation Plug-in

class	tech.	width	depth	action	energy (pJ)	area (um^2)
MAC	45nm	16b	N/A	compute	5	0.4
SRAM	**45nm**	**64b**	**1024**	**access**	**100**	**20**
SRAM	45nm	16b	256	access	10	2

Available at http://accelergy.mit.edu/

Accelergy: Flexibly Model Various Primitive Components



Each component belongs to a primitive class

ERT/ART (in progress)

comp.	action	energy	area
GLB	access()	100pJ	20um^2

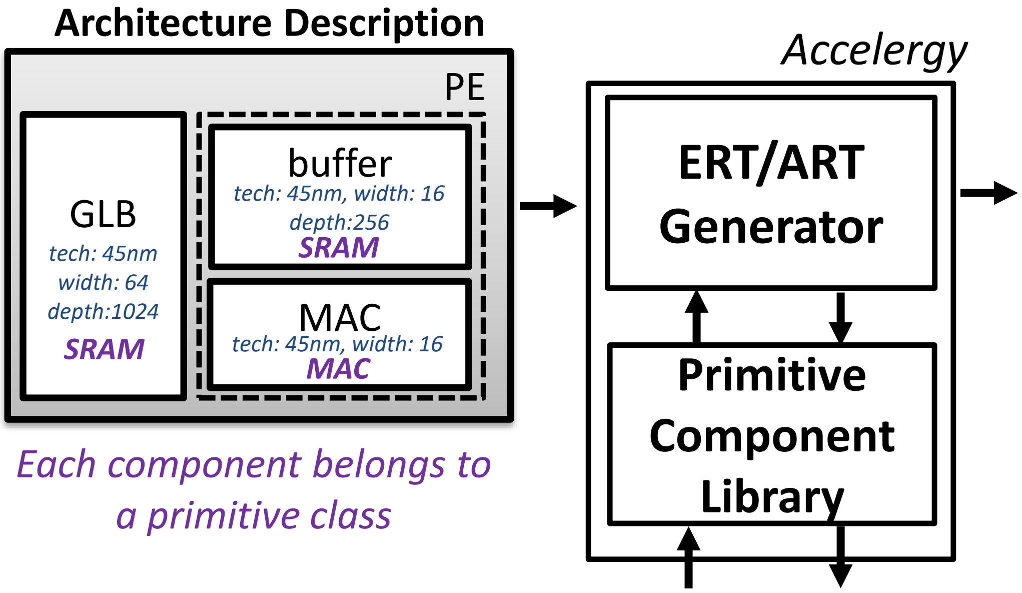
SRAM class has associated action “access”

Simple Example Estimation Plug-in

class	tech.	width	depth	action	energy (pJ)	area (um^2)
MAC	45nm	16b	N/A	compute	5	0.4
SRAM	**45nm**	**64b**	**1024**	**access**	**100**	**20**
SRAM	45nm	16b	256	access	10	2

Available at http://accelergy.mit.edu/

Accelergy: Flexibly Model Various Primitive Components



Each component belongs to a primitive class

ERT/ART

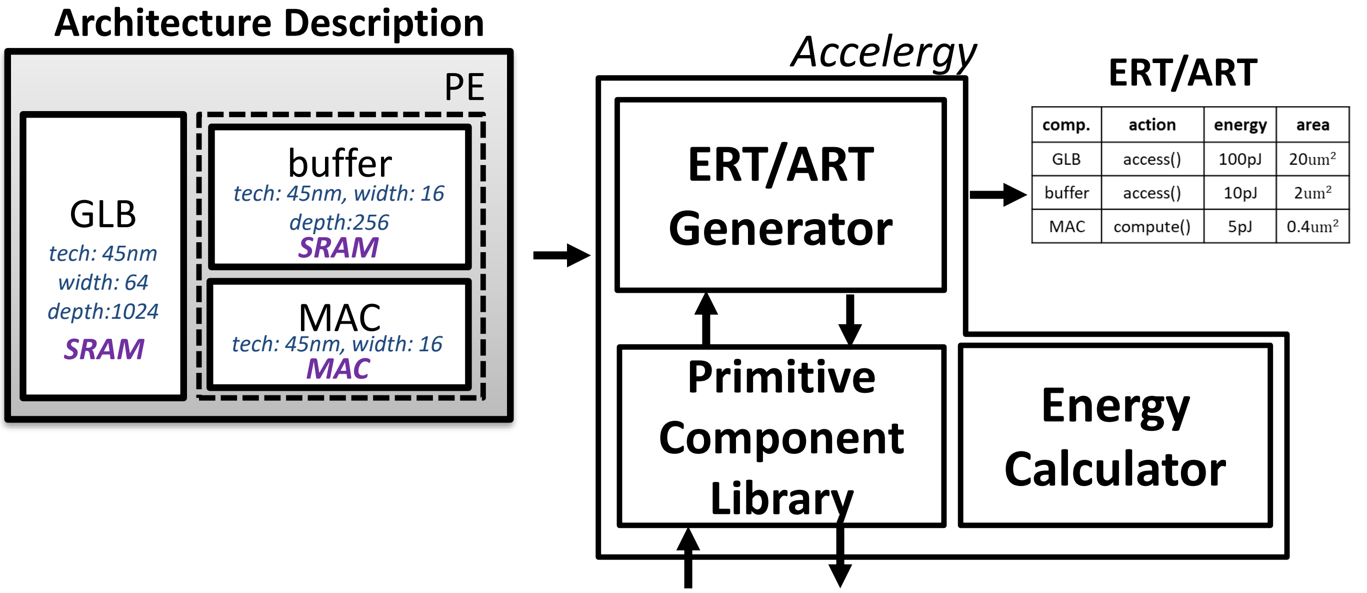
comp.	action	energy	area
GLB	access()	100pJ	20um^2
buffer	access()	10pJ	2um^2
MAC	compute()	3pJ	0.3um^2

Simple Example Estimation Plug-in

class	tech.	width	depth	action	energy (pJ)	area (um^2)
MAC	45nm	16b	N/A	compute	5	0.4
SRAM	45nm	64b	1024	access	100	20
SRAM	45nm	16b	256	access	10	2

Available at http://accelergy.mit.edu/

Accelergy: Flexibly Model Various Primitive Components



comp.	action	energy	area
GLB	access()	100pJ	20um^2
buffer	access()	10pJ	2um^2
MAC	compute()	5pJ	0.4um^2

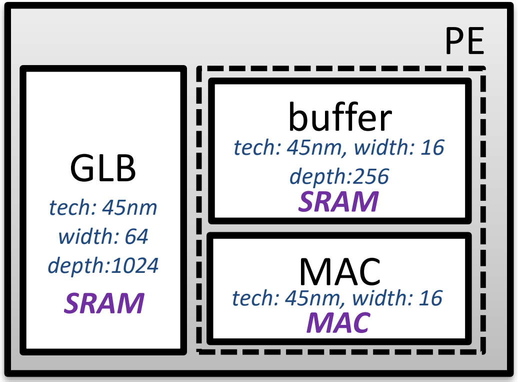
Simple Example Estimation Plug-in

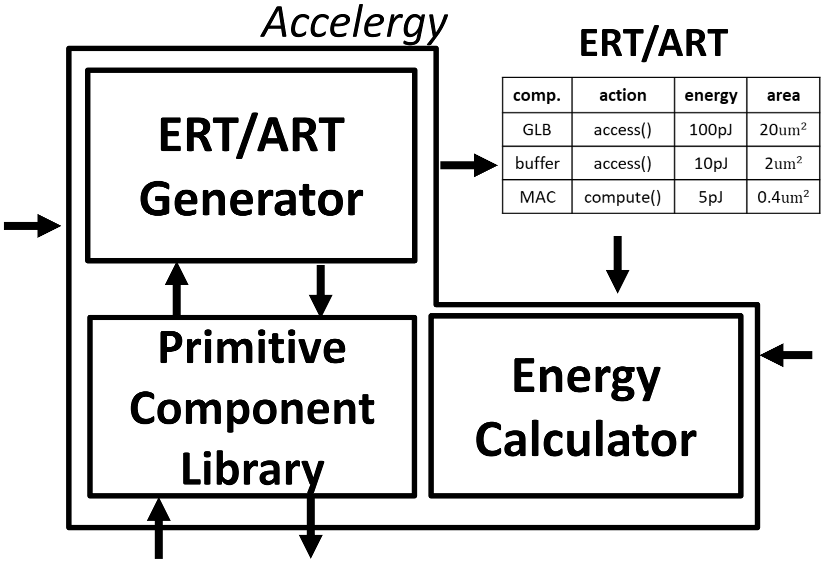
class	tech.	width	depth	action	energy (pJ)	area (um^2)
MAC	45nm	16b	N/A	compute	5	0.4
SRAM	45nm	64b	1024	access	100	20
SRAM	45nm	16b	256	access	10	2

Available at http://accelergy.mit.edu/

Accelergy: Flexibly Model Various Primitive Components

Architecture Description





ERT/ART

comp.	action	energy	area
GLB	access()	100pJ	20um^2
buffer	access()	10pJ	2um^2
MAC	compute()	5pJ	0.4um^2

Comes from a performance model (e.g., cycle accurate simulator)

Action Counts

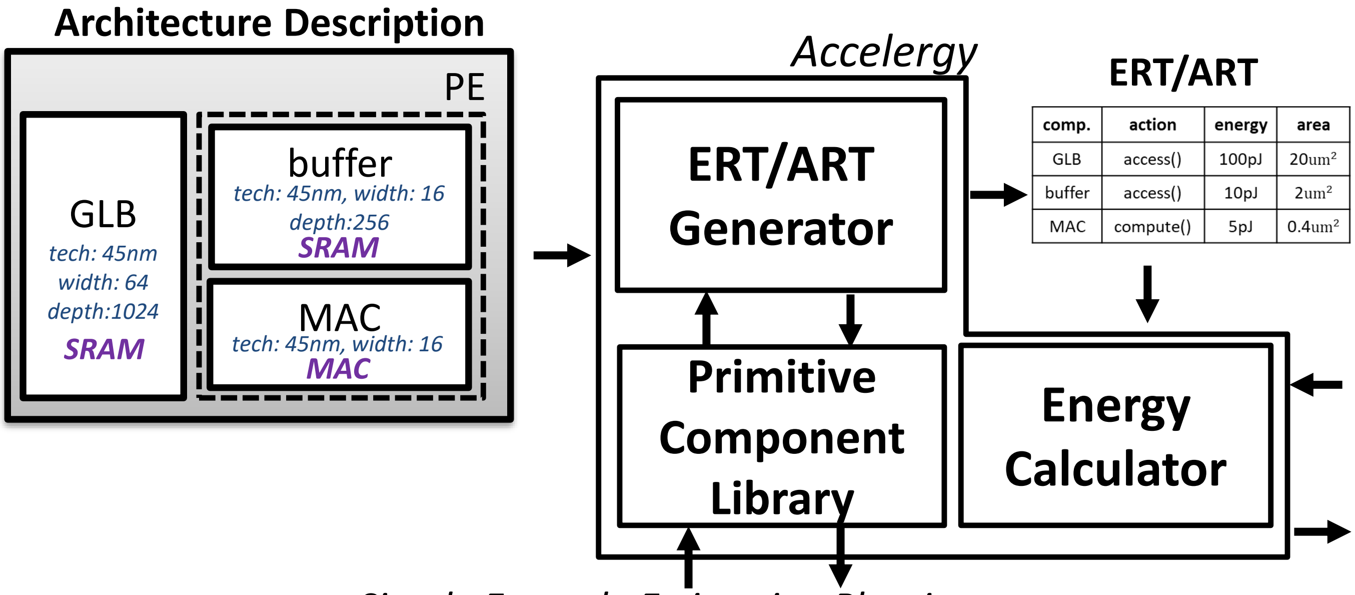
Comp.	Action	Counts
GLB	access()	10
buffer	access()	800
MAC	compute()	400

Simple Example Estimation Plug-in

class	tech.	width	depth	action	energy (pJ)	area (um^2)
MAC	45nm	16b	N/A	compute	5	0.4
SRAM	45nm	64b	1024	access	100	20
SRAM	45nm	16b	256	access	10	2

Available at http://accelergy.mit.edu/

Accelergy: Flexibly Model Various Primitive Components



ERT/ART

comp.	action	energy	area
GLB	access()	100pJ	$20um^2$
buffer	access()	10pJ	$2um^2$
MAC	compute()	5pJ	$0.4um^2$

Comes from a performance model (e.g., cycle accurate simulator)

Action Counts

Comp.	Action	Counts
GLB	access()	10
buffer	access()	800
MAC	compute()	400

Name	Energy
GLB	1000pJ
buffer	8000pJ
MAC	2000pJ

Energy Estimates

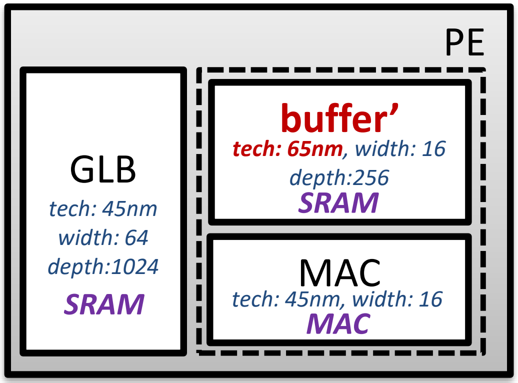
Simple Example Estimation Plug-in

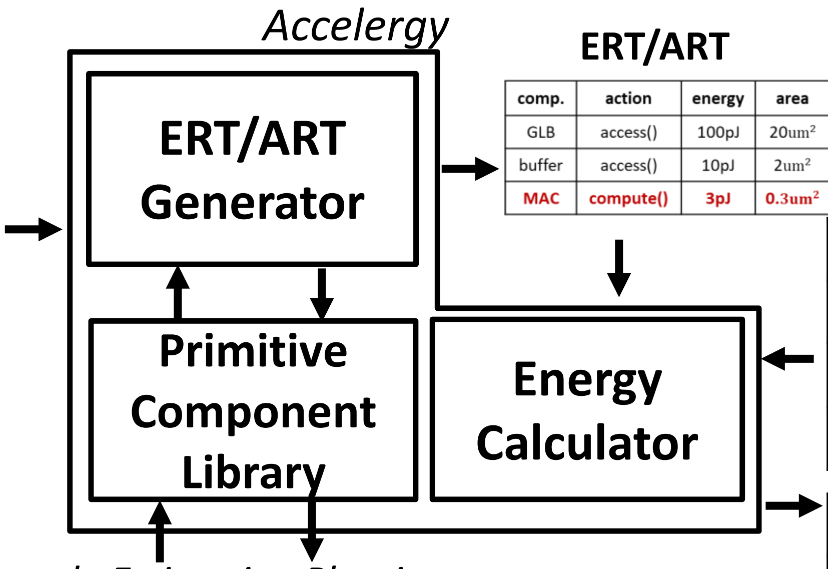
class	tech.	width	depth	action	energy (pJ)	area (um^2)
MAC	45nm	16b	N/A	compute	5	0.4
SRAM	45nm	64b	1024	access	100	20
SRAM	45nm	16b	256	access	10	2

Available at http://accelergy.mit.edu/

Accelergy: Flexibly Model Various Primitive Components

Architecture Description





ERT/ART

comp.	action	energy	area
GLB	access()	100pJ	$20um^2$
buffer	access()	10pJ	$2um^2$
MAC	compute()	3pJ	$0.3um^2$

Comes from a performance model (e.g., cycle accurate simulator)

Action Counts

Comp.	Action	Counts
GLB	access()	10
buffer	access()	800
MAC	compute()	400

Name	Energy
GLB	1000pJ
buffer	8000pJ
MAC	1200pJ

Energy Estimates

Simple Example Estimation Plug-in

class	tech.	width	depth	action	energy (pJ)	area (um^2)
MAC	45nm	16b	N/A	compute	5	0.4
SRAM	45nm	64b	1024	access	100	20
SRAM	45nm	16b	256	access	10	2
MAC	65nm	16b	N/A	compute	3	0.3

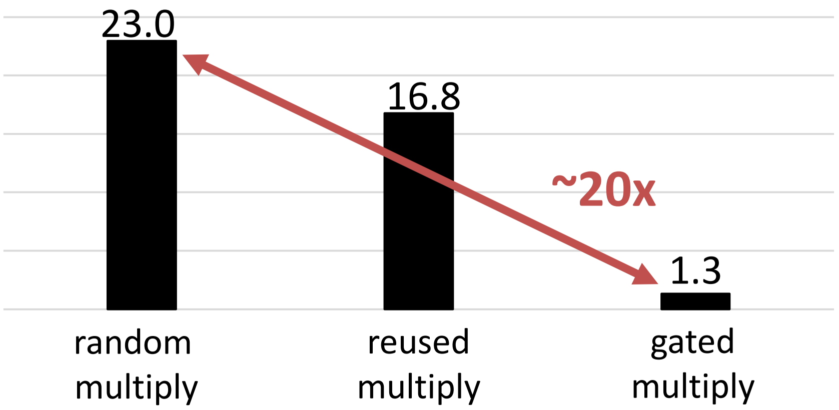
Accelergy Overview

- **Accelergy Infrastructure**
  - Performs architecture-level estimations to enable rapid design space exploration
  - Supports modeling of diverse architectures with various underlying technologies
  - **Improves estimation accuracy by allowing fine-grained classification of components runtime behaviors**
  - Supports succinct modeling of complicated architectures
- Validation on various accelerator designs
  - 95% accurate on a conventional digital accelerator design
  - Modeling of processing in memory (PIM) based DNN accelerator designs

Plug-ins for Fine-Grain Action Energy Estimation

- **External energy/area models that accurately reflect the properties of a macro**
  - **e.g., multiplier with zero-gating**

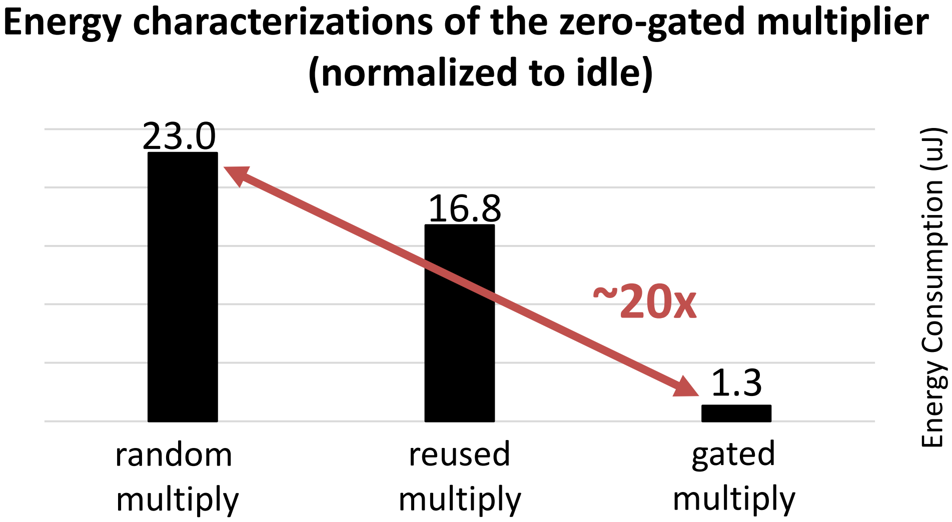
Energy characterizations of the zero-gated multiplier (normalized to idle)

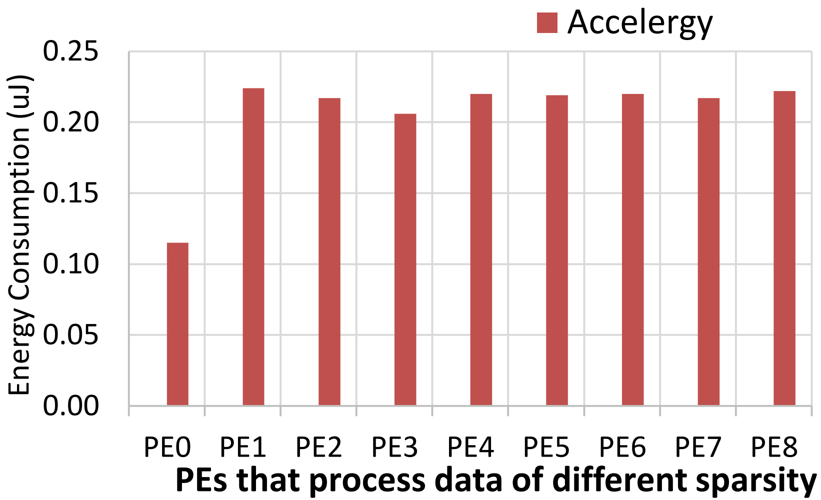


name	tech.	width	action	energy
multiplier	65nm	16b	random multiply	23.0
multiplier	65nm	16b	reused multiply	16.8
multiplier	65nm	16b	gated multiply	1.3

Plug-ins for Fine-Grain Action Energy Estimation

- External energy/area models that accurately reflect the properties of a macro
  - e.g., multiplier with zero-gating



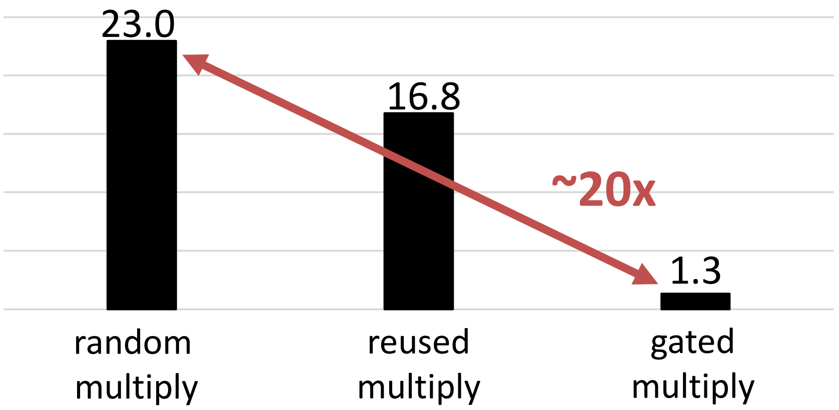


With the characterization provided in the plug-in, we can see significant energy savings for sparse workloads

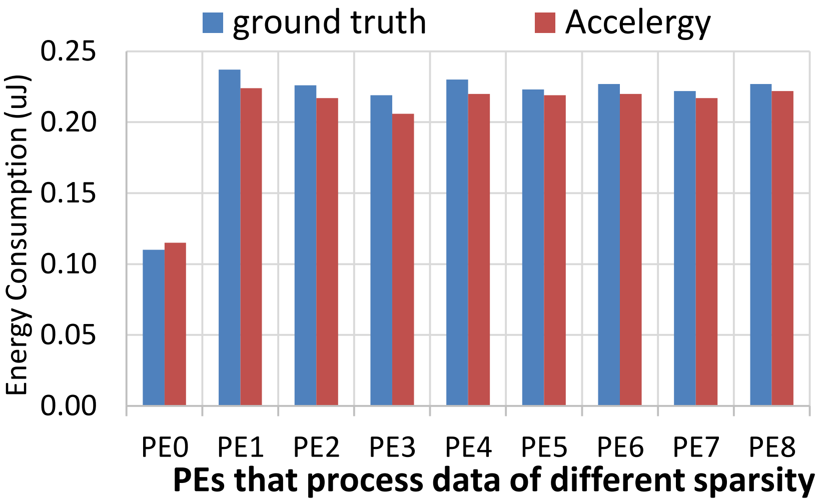
Plug-ins for Fine-Grain Action Energy Estimation

- External energy/area models that accurately reflect the properties of a macro
  - e.g., multiplier with zero-gating

Energy characterizations of the zero-gated multiplier (normalized to idle)



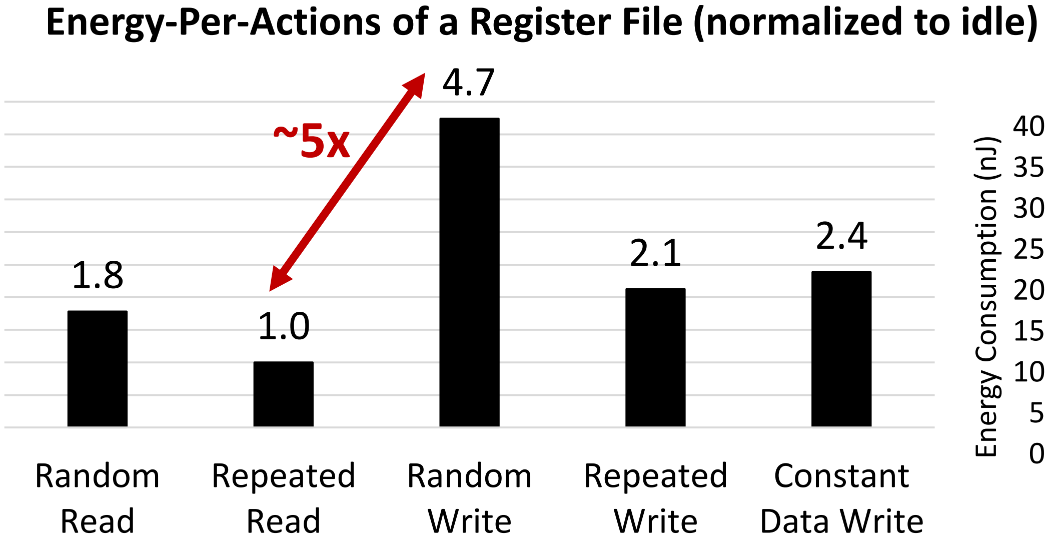
PnR simulations

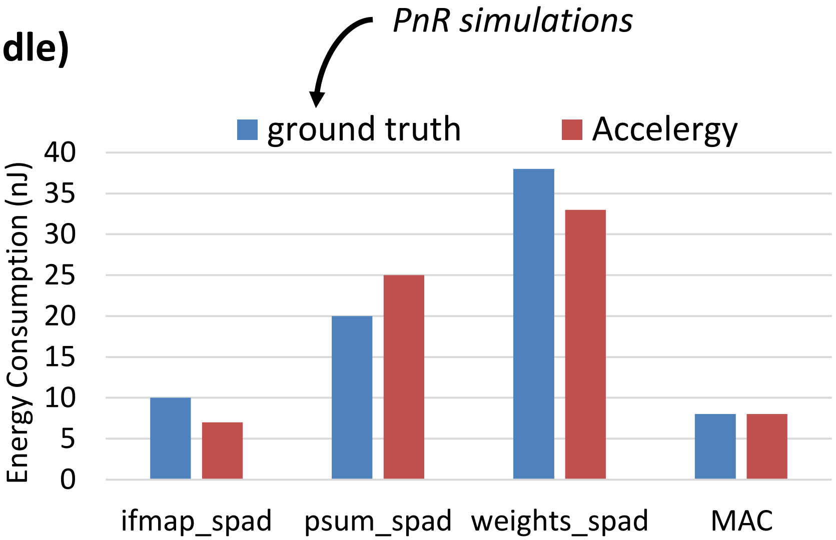


With the characterization provided in the plug-in, we can see significant energy savings for sparse workloads

Plug-ins for Fine-Grain Action Energy Estimation Plug-ins

- External energy/area models that accurately reflect the properties of a macro
  - e.g., register file with various access types





**With the characterization provided in the plug-in,
we can see accurate characterization for memories with different access patterns**

Flexibly Model Various Primitive Components

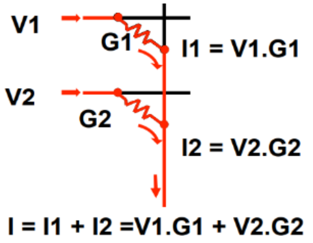
Use energy estimation plug-ins to characterize primitive components

CACTI Estimation Plug-in

45nm Estimation Plug-in

Traditional open-source plug-ins

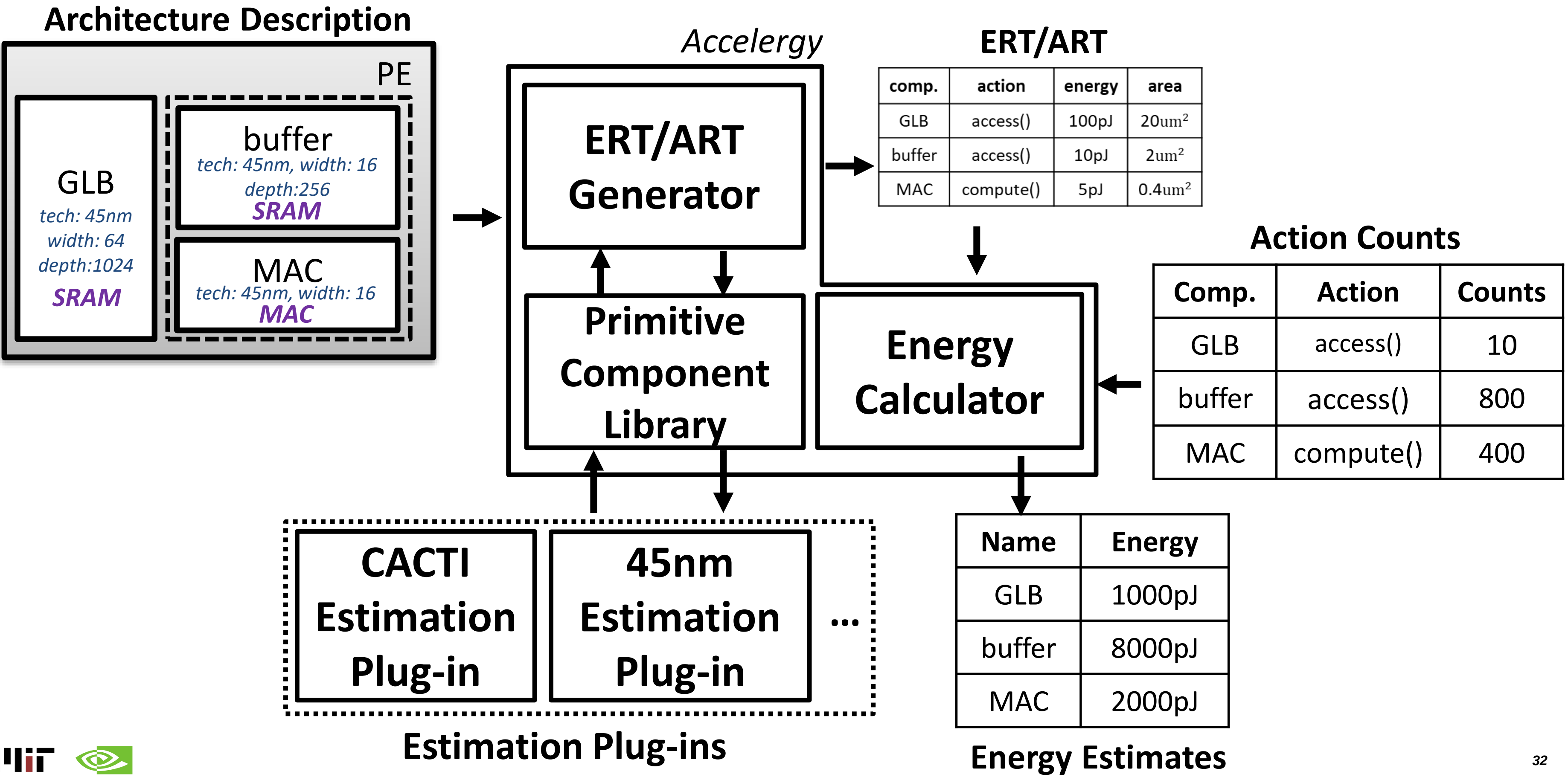
Proprietary plug-ins



NVSIM [TCAD 2012]

Emerging technology plug-ins

Accelergy: Flexibly Model Various Primitive Components



Architecture Description

Estimation Plug-ins

ERT/ART

comp.	action	energy	area
GLB	access()	100pJ	$20um^2$
buffer	access()	10pJ	$2um^2$
MAC	compute()	5pJ	$0.4um^2$

Action Counts

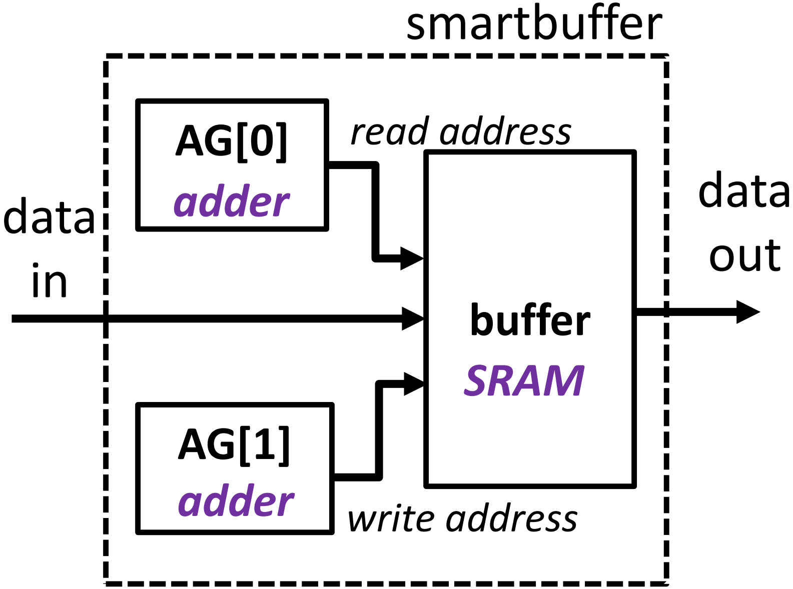
Comp.	Action	Counts
GLB	access()	10
buffer	access()	800
MAC	compute()	400

Name	Energy
GLB	1000pJ
buffer	8000pJ
MAC	2000pJ

Energy Estimates

Modeling Complicated Designs

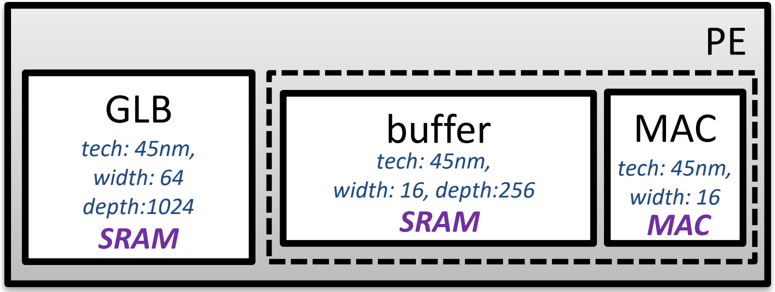
- **Practical designs involve many more primitive components**
  - **Example: smartbuffer – a storage unit with preprogrammed address generators (AGs)**
    - Domain-specific applications have predictable storage access patterns, allowing offline access stream generation, e.g., general matrix multiply applications.



- ○ buffer belongs to *SRAM* class
- ○ AGs belongs to *adder* class

Modeling Complicated Designs

- **Practical designs involve many more primitive components**

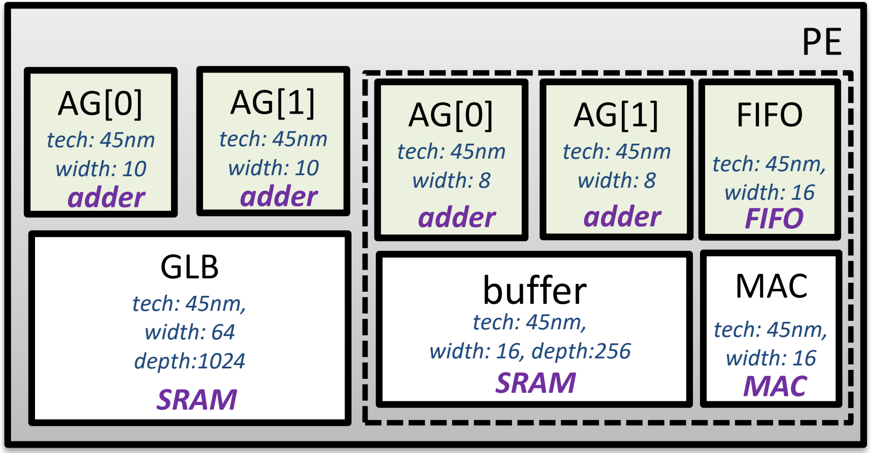


Simple Architecture Design

Let's construct a more practical design!

Modeling Complicated Designs

- **Practical designs involve many more primitive components**

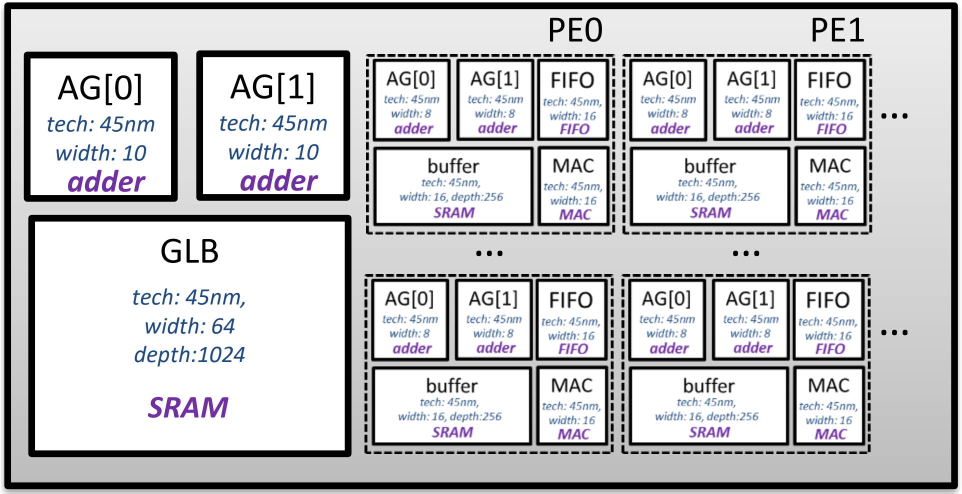


Practical Architecture Design

Let's construct a more practical design!

Modeling Complicated Designs

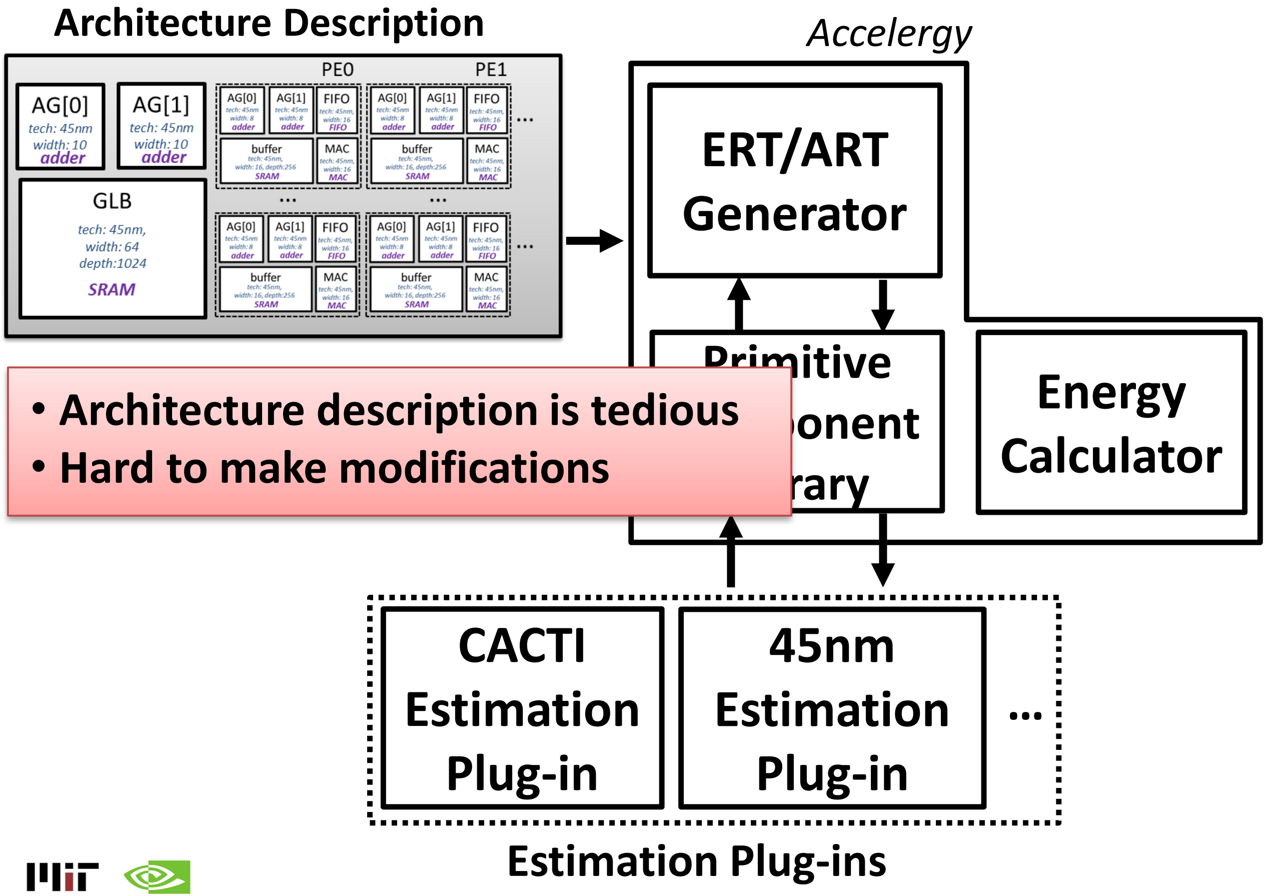
- **Practical designs involve many more primitive components**



Practical Architecture Design

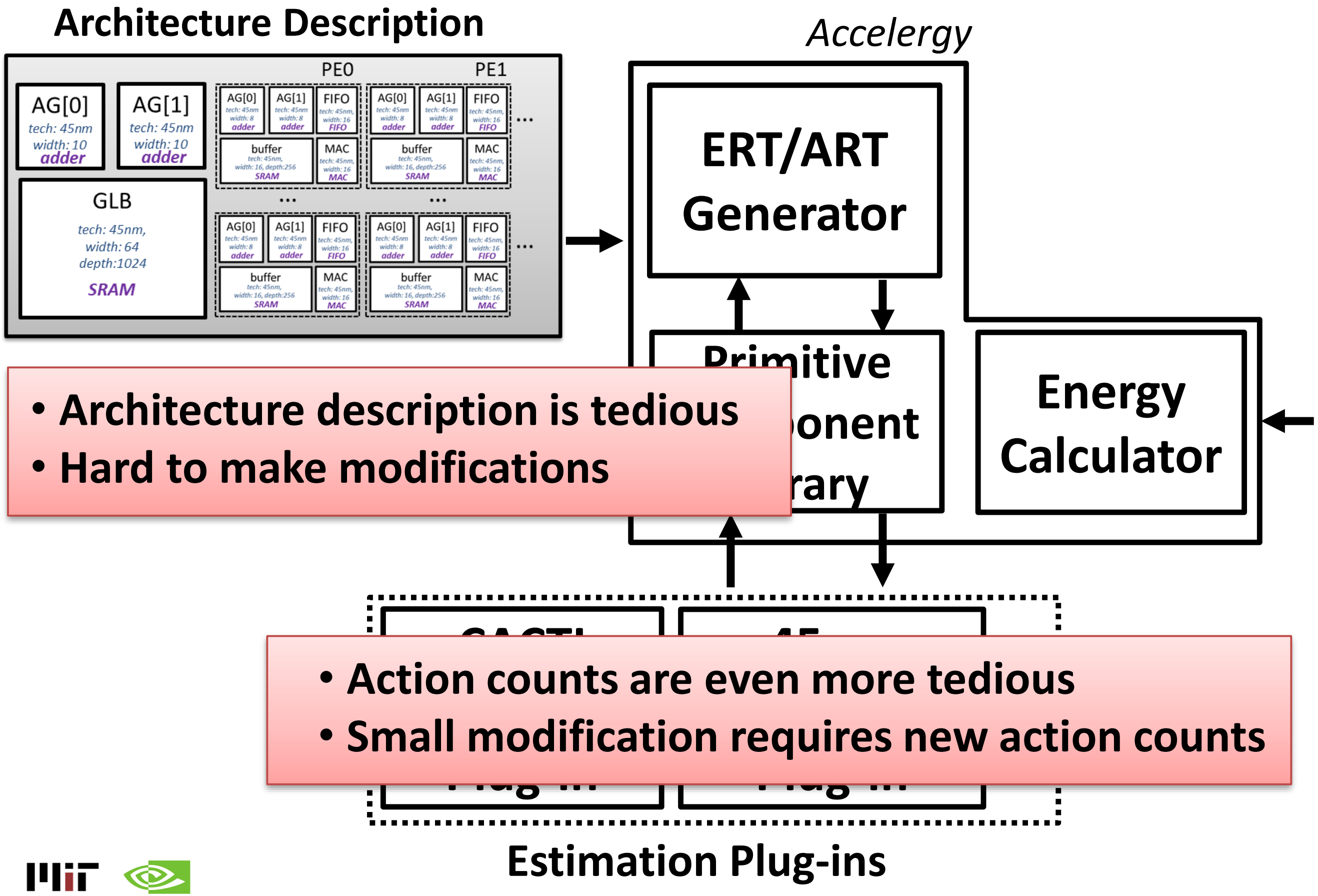
Let's construct a more practical design!

Modeling Complicated Designs



- Architecture description is tedious
- Hard to make modifications

Modeling Complicated Designs



- Architecture description is tedious
- Hard to make modifications

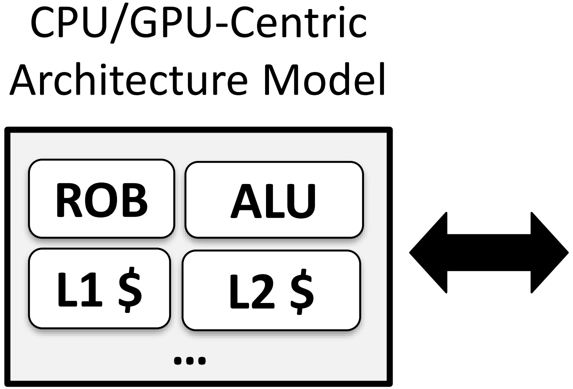
- Action counts are even more tedious
- Small modification requires new action counts

Action Counts

component	action	counts
GLB	access()	10
AG[0]	add()	7
AG[1]	add()	3
PE0.buffer	access()	800
PE0.AG[0]	add()	680
PE0.AG[1]	add()	120
PE0.MAC	compute()	370
PE0.FIFO	access()	370
PE1.buffer	access()	830
PE1.AG[0]	add()	690
…		

Existing Architecture-Level Energy Estimators

- **Architecture-level energy modeling for general purpose processors**
  - **Wattch[Brooks, ISCA2000], McPAT[Li, MICRO2009], GPUWattch[Leng, ISCA2013], PowerTrain[Lee, ISLPED2015]**

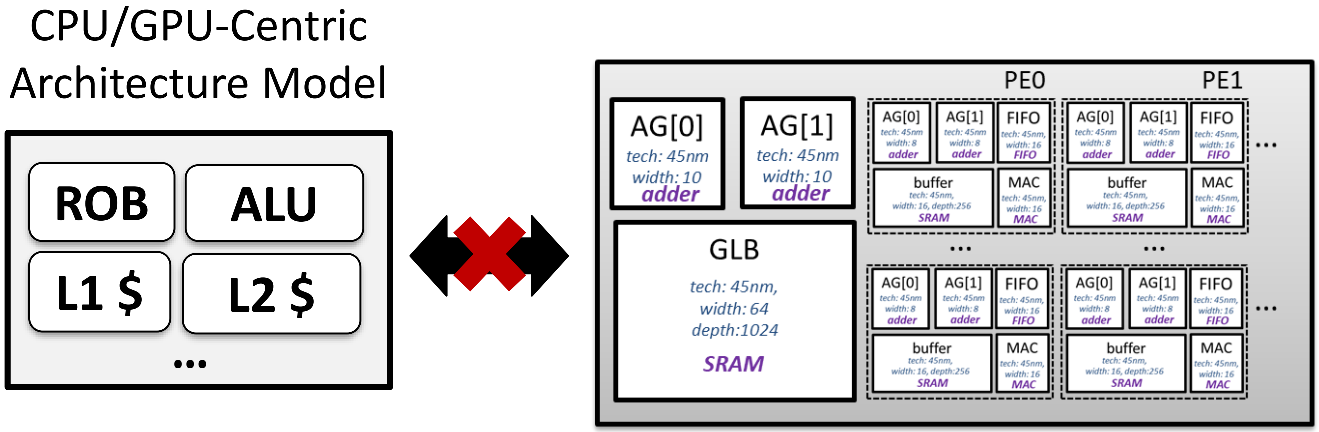

CPU/GPU-Centric Architecture Model

Use a fixed set of compound components to represent the architecture

Components that can be decomposed into lower level components

Existing Architecture-Level Energy Estimators

- **Architecture-level energy modeling for general purpose processors**
  - **Wattch[Brooks, ISCA2000], McPAT[Li, MICRO2009], GPUWattch[Leng, ISCA2013], PowerTrain[Lee, ISLPED2015]**



CPU/GPU-Centric Architecture Model

The fixed set of compound components is not sufficient to describe various optimizations in the diverse accelerator design space

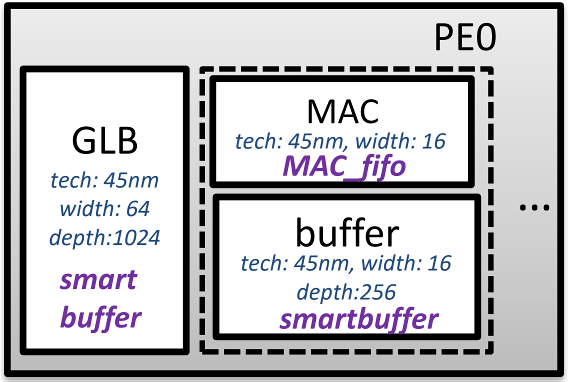
Accelergy Overview

- **Accelergy Infrastructure**
  - Performs architecture-level estimations to enable rapid design space exploration
  - Supports modeling of diverse architectures with various underlying technologies
  - Improves estimation accuracy by allowing fine-grained classification of components runtime behaviors
  - **Supports succinct modeling of complicated architectures**
- Validation on various accelerator designs
  - 95% accurate on a conventional digital accelerator design
  - Modeling of processing in memory (PIM) based DNN accelerator designs

Accelergy: Succinctly Model Arbitrary Architecture

- **Allow succinct architecture description with user-defined compound component classes**

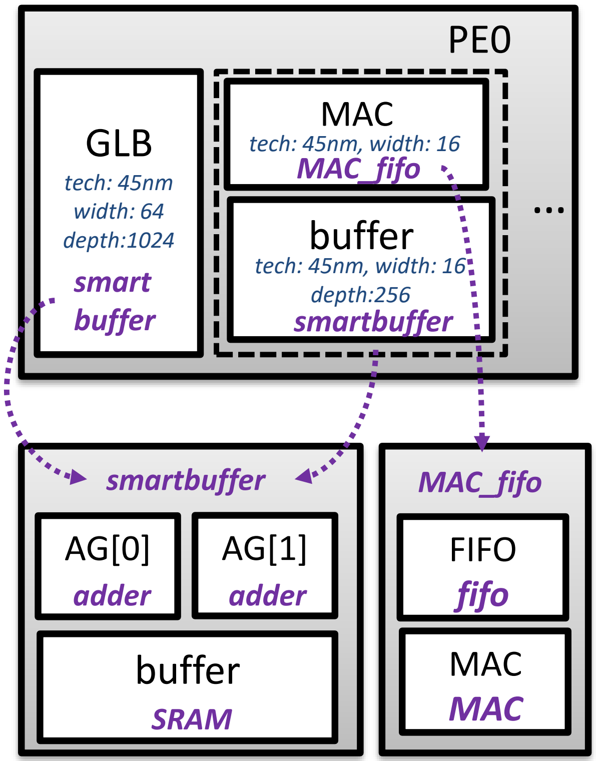
Architecture Description



Accelergy: Succinctly Model Arbitrary Architecture

- **Allow succinct architecture description with user-defined compound component classes**
- **Allow user-defined compound component hardware structure using primitive components**

Architecture Description

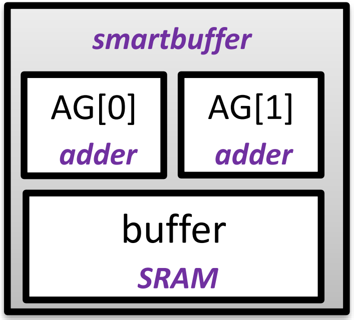


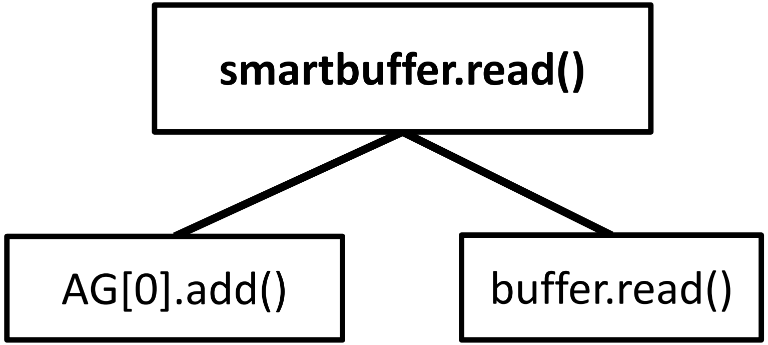
Compound component description

Accelergy: Succinctly Model Arbitrary Architecture

- **Allow succinct architecture description with user-defined compound component classes**
- **Allow user-defined compound component hardware structure using primitive components**
- **Allow user-defined compound component actions using primitive component actions**

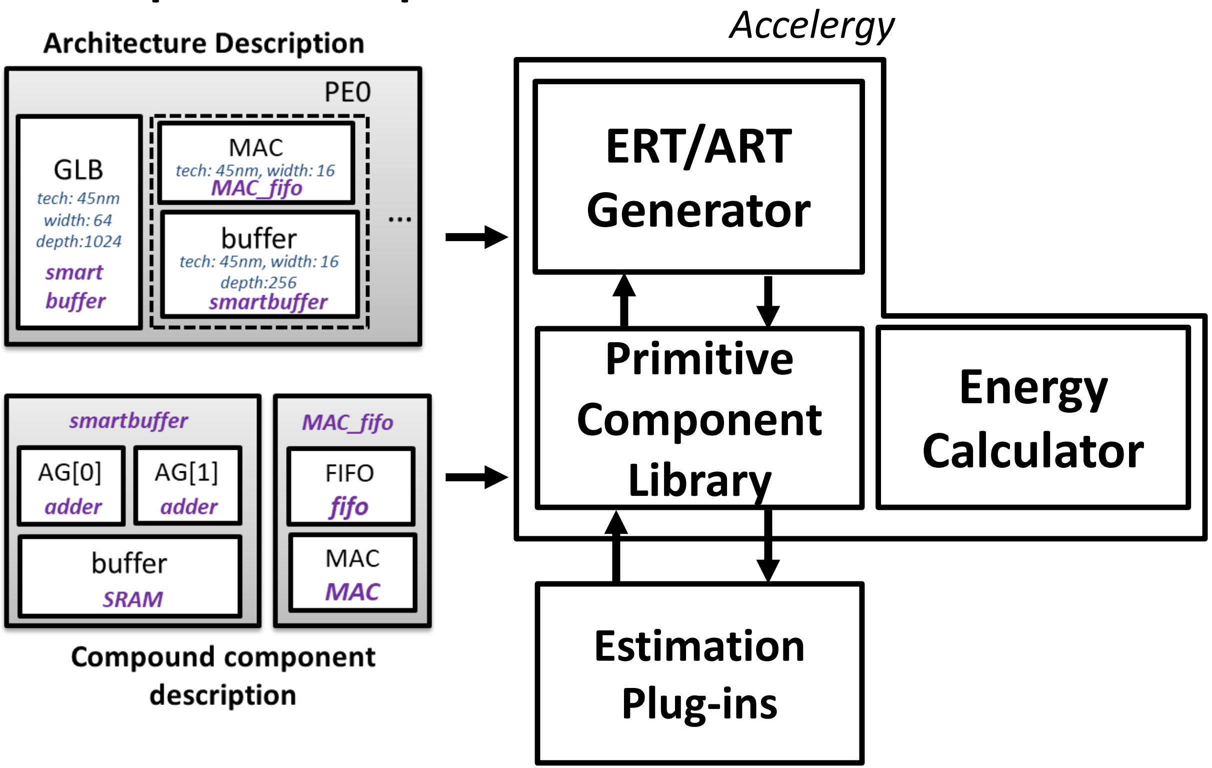
Compound component description





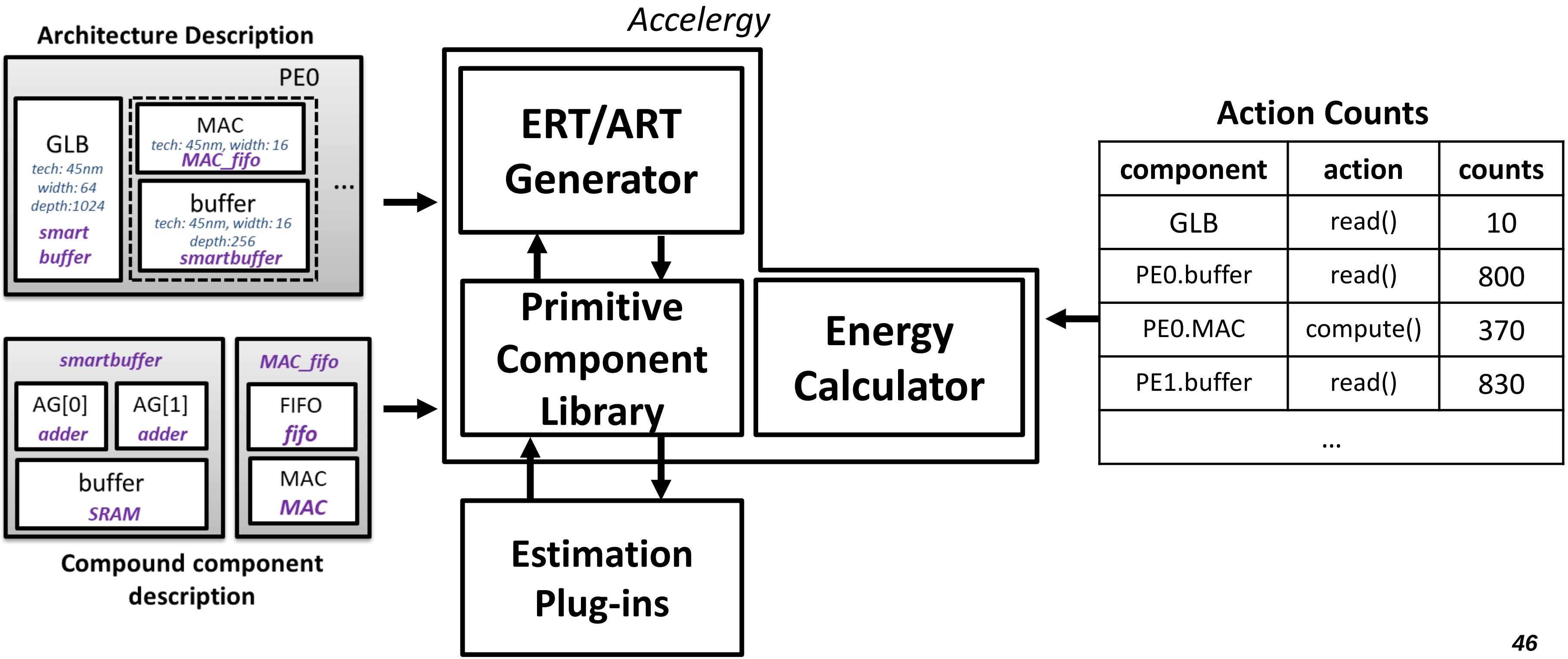
Accelergy: Succinctly Model Arbitrary Architecture

- Flexible and succinct architecture representations using user-defined compound components



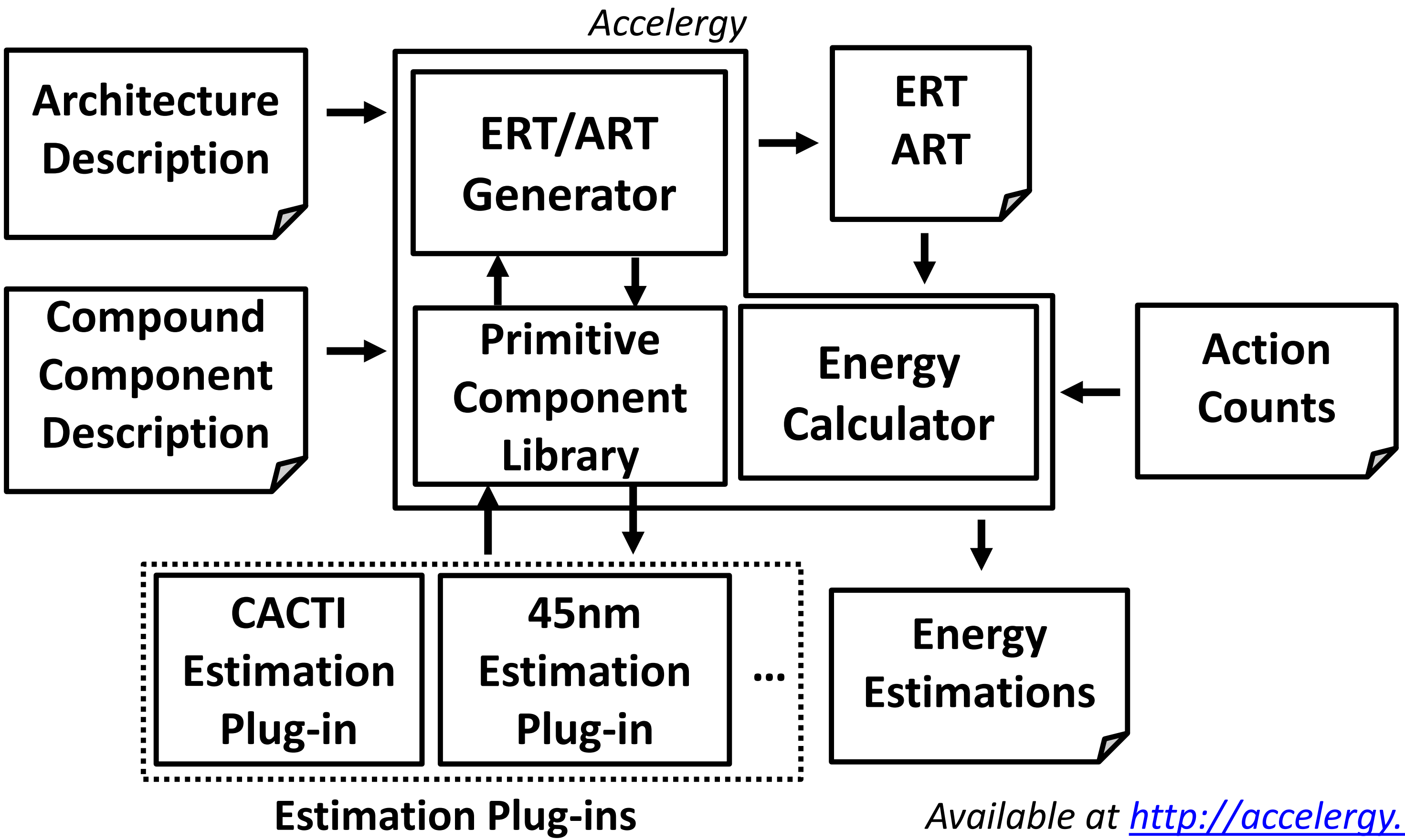
Accelergy: Succinctly Model Arbitrary Architecture

- **Flexible and succinct action counts using compound actions**



component	action	counts
GLB	read()	10
PE0.buffer	read()	800
PE0.MAC	compute()	370
PE1.buffer	read()	830
…		

Accelergy High-Level Infrastructure



Accelergy

Architecture Description

Compound Component Description

ERT/ART Generator

Primitive Component Library

ERT ART

Energy Calculator

Action Counts

CACTI Estimation Plug-in

45nm Estimation Plug-in

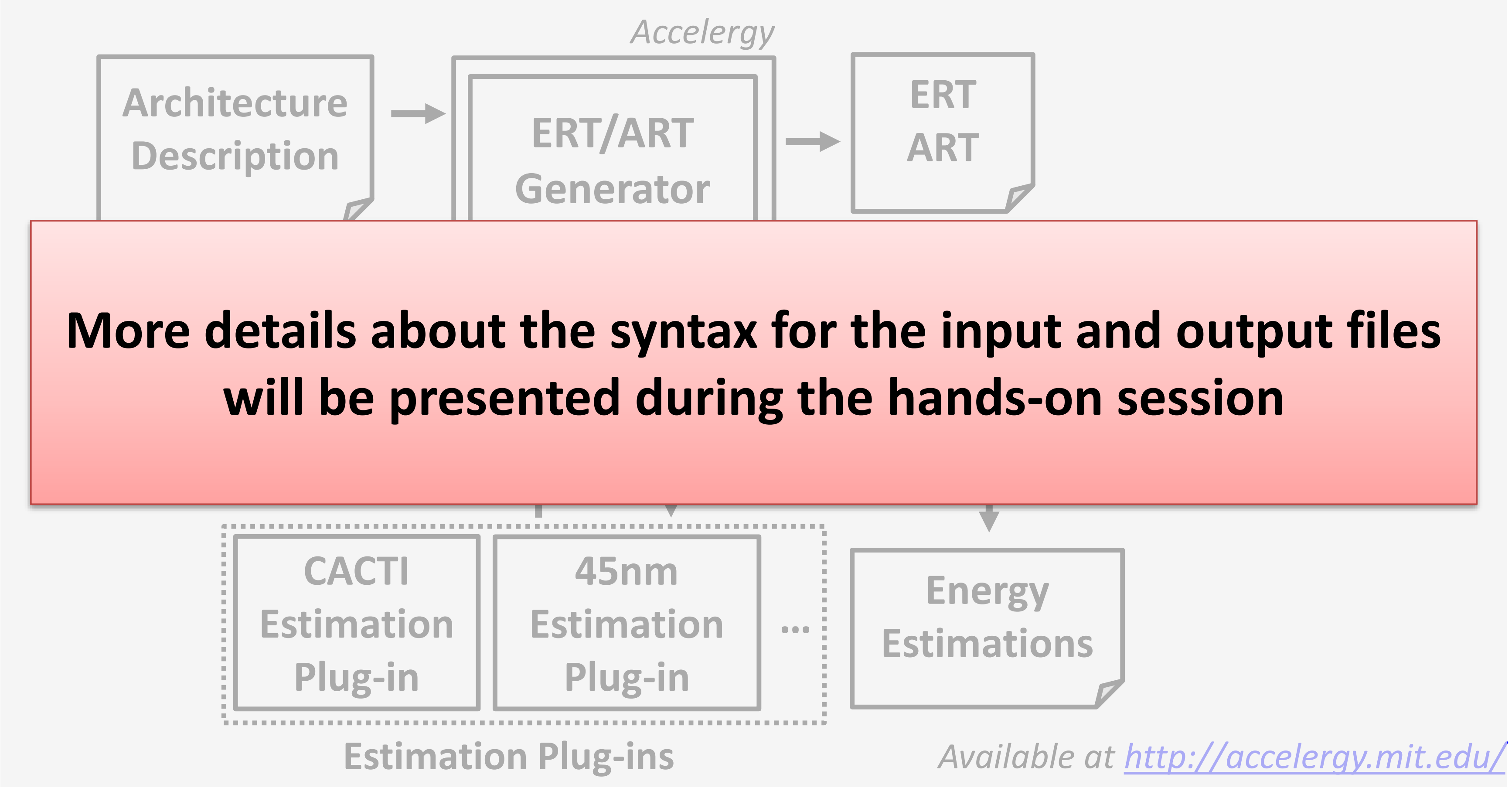
…

Estimation Plug-ins

Energy Estimations

Available at http://accelergy.mit.edu/

Accelergy High-Level Infrastructure



More details about the syntax for the input and output files will be presented during the hands-on session

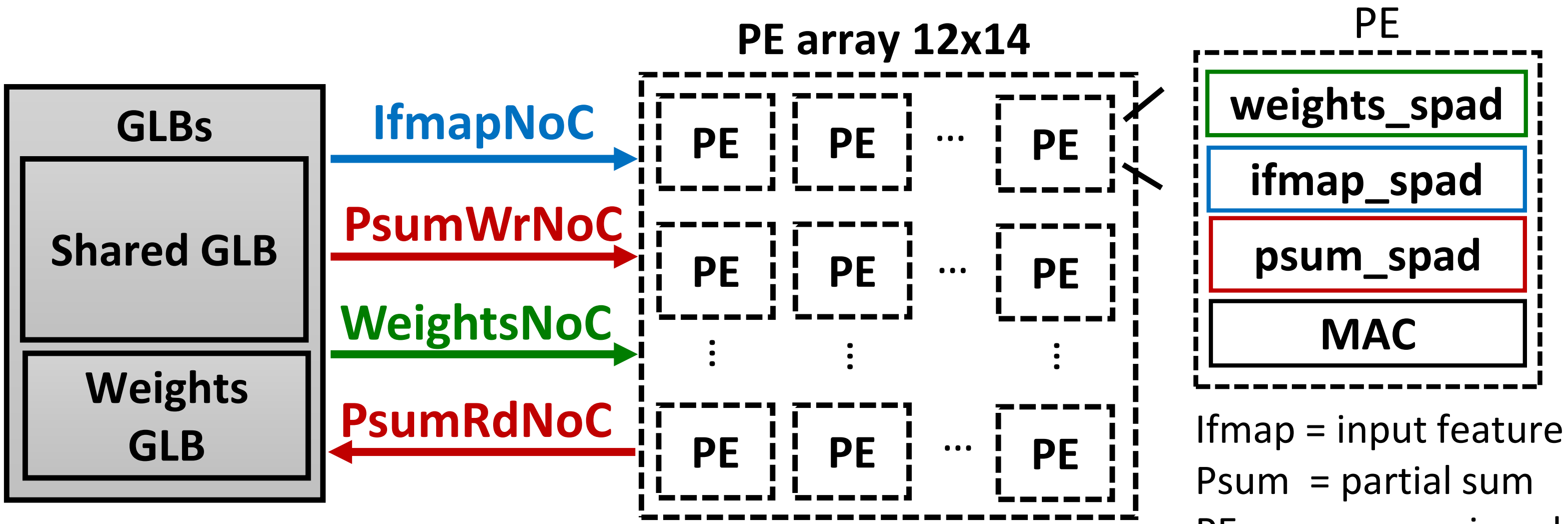
Available at http://accelergy.mit.edu/

Accelergy Overview

- Accelergy Infrastructure
  - Performs architecture-level estimations to enable rapid design space exploration
  - Supports modeling of diverse architectures with various underlying technologies
  - Improves estimation accuracy by allowing fine-grained classification of components runtime behaviors
  - Supports succinct modeling of complicated architectures
- **Validation on various accelerator designs**
  - **95% accurate on a conventional digital accelerator design**
  - Modeling of processing in memory (PIM) based DNN accelerator designs

Energy Validation on Eyeriss [Chen, ISSCC 2016]

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

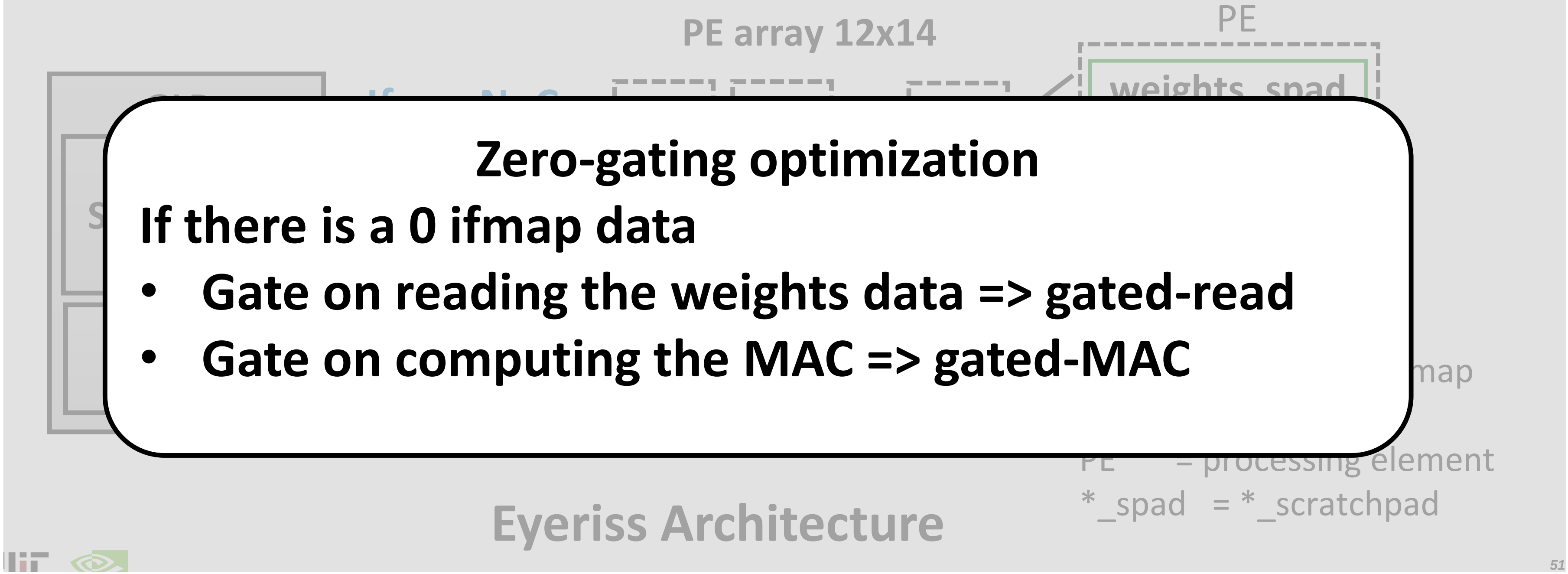


Ifmap = input feature map
Psum = partial sum
PE = processing element
*_spad = *_scratchpad

Eyeriss Architecture

Energy Validation on Eyeriss [Chen, ISSCC 2016]

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



PE = processing element
*_spad = *_scratchpad

Eyeriss Architecture

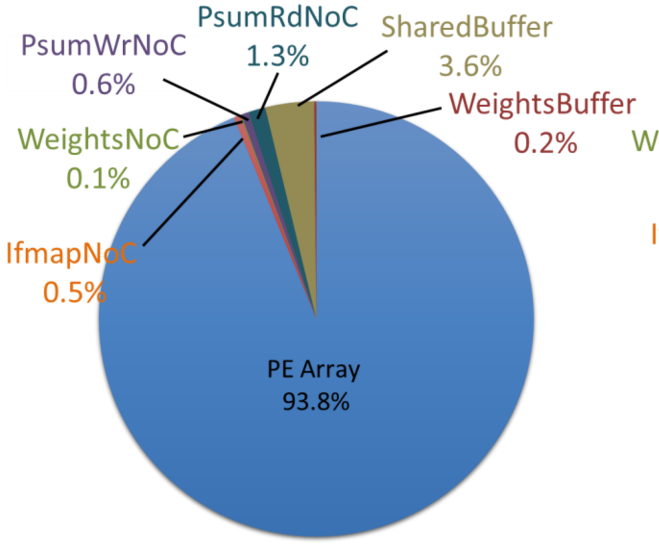
Zero-gating optimization

If there is a 0 ifmap data

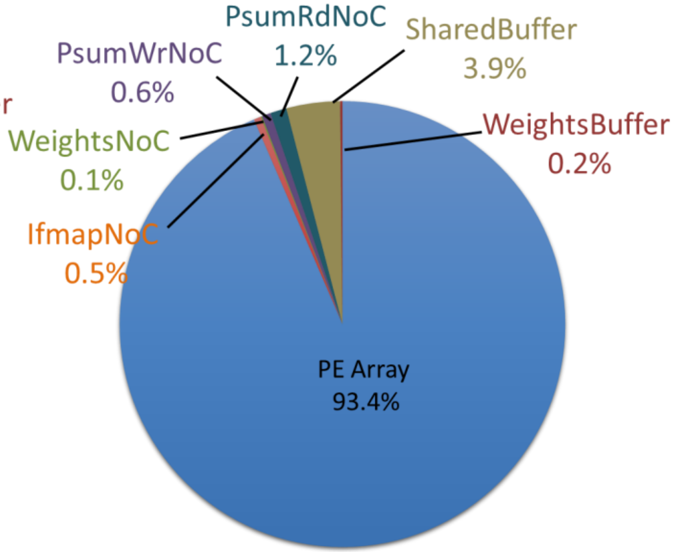
- **Gate on reading the weights data => gated-read**
- **Gate on computing the MAC => gated-MAC**

Energy Validation on Eyeriss [Chen, ISSCC 2016]

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



Ground Truth Energy Breakdown



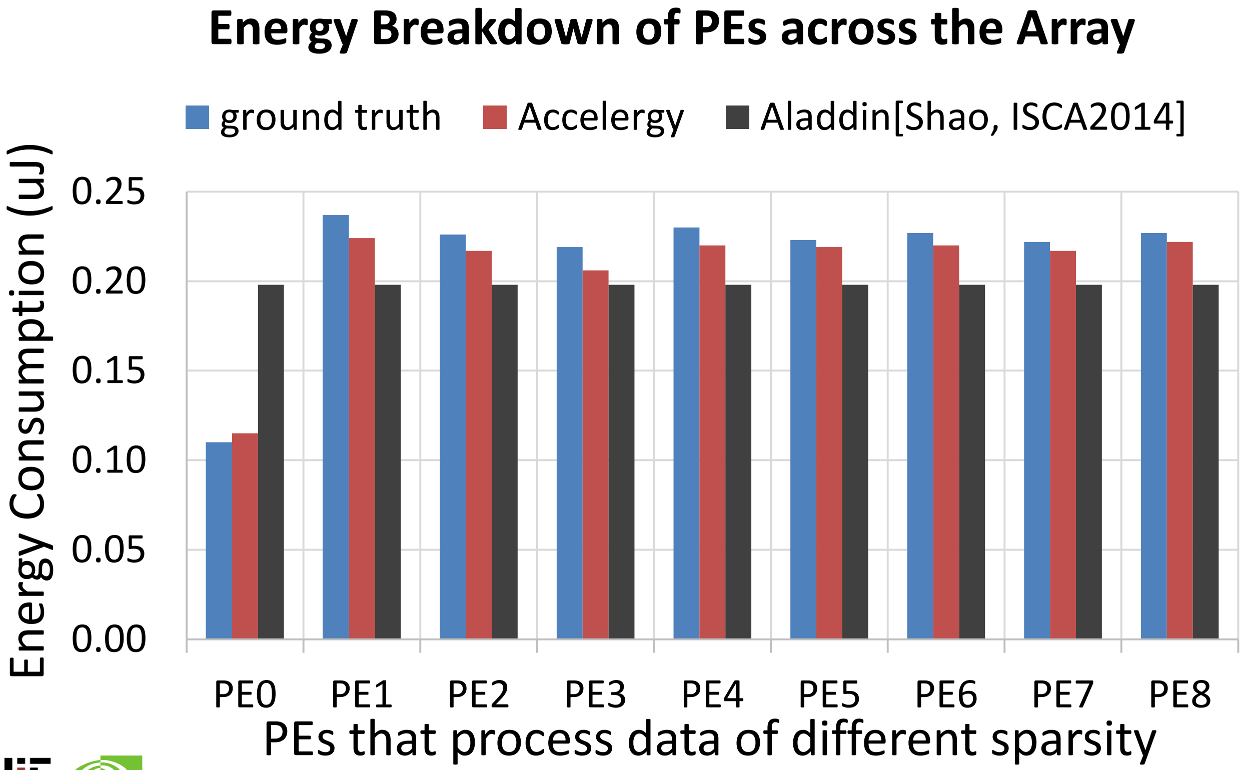
Accelergy Energy Breakdown

Published at [Wu, ICCAD 2019]

**Total energy might not add up to exact 100.0% due to rounding*

PE Array Energy Breakdown

- **Comparisons with existing work: Aladdin[Shao, ISCA2014]**



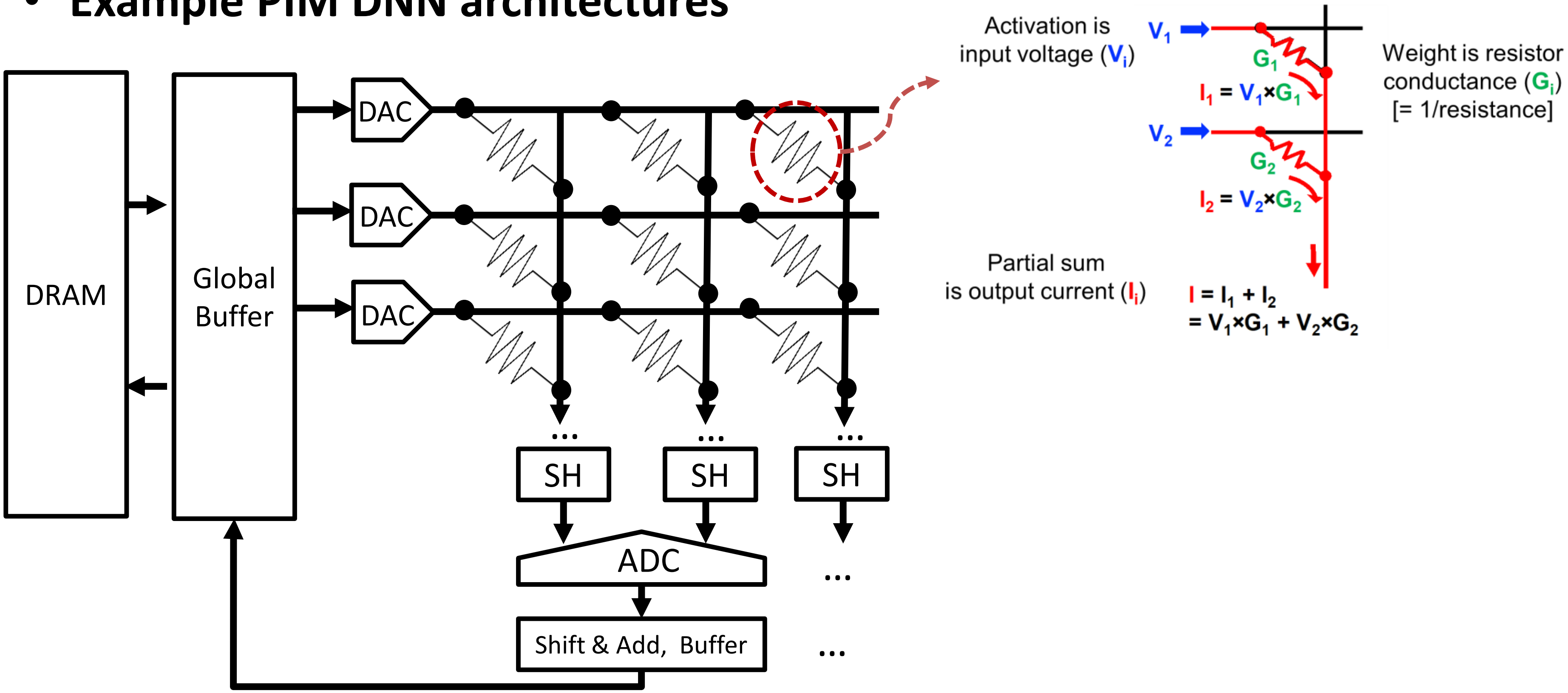
Energy impact of sparsity is accurately captured with sparsity-aware estimation plug-ins

Accelergy Overview

- Accelergy Infrastructure
  - Performs architecture-level estimations to enable rapid design space exploration
  - Supports modeling of diverse architectures with various underlying technologies
  - Improves estimation accuracy by allowing fine-grained classification of components runtime behaviors
  - Supports succinct modeling of complicated architectures
- Validation on various accelerator designs
  - 95% accurate on a conventional digital accelerator design
  - **Modeling of processing in memory (PIM) based DNN accelerator designs**

Accelergy Modeling of PIM Architectures

- **Example PIM DNN architectures**



DRAM

Global Buffer

DAC

DAC

DAC

SH

SH

SH

ADC

Shift & Add, Buffer

...

Activation is input voltage (V_i)

Weight is resistor conductance (G_i) [= 1/resistance]

Partial sum is output current (I_i)

V_1

V_2

G_1

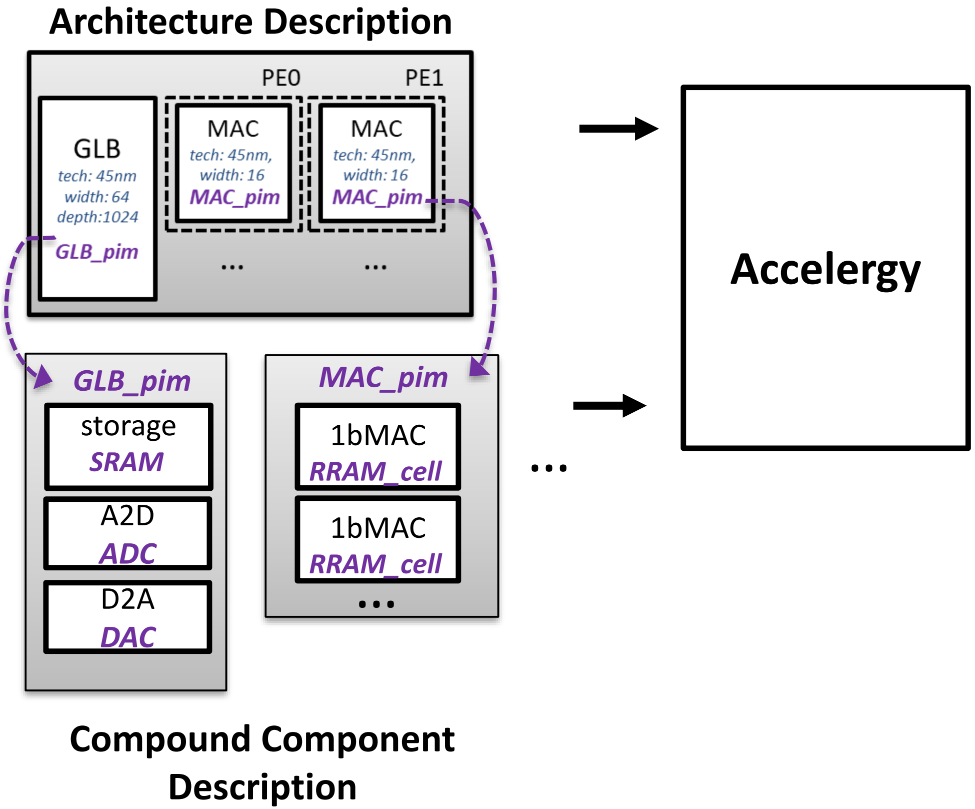
G_2

$I_1 = V_1 \times G_1$

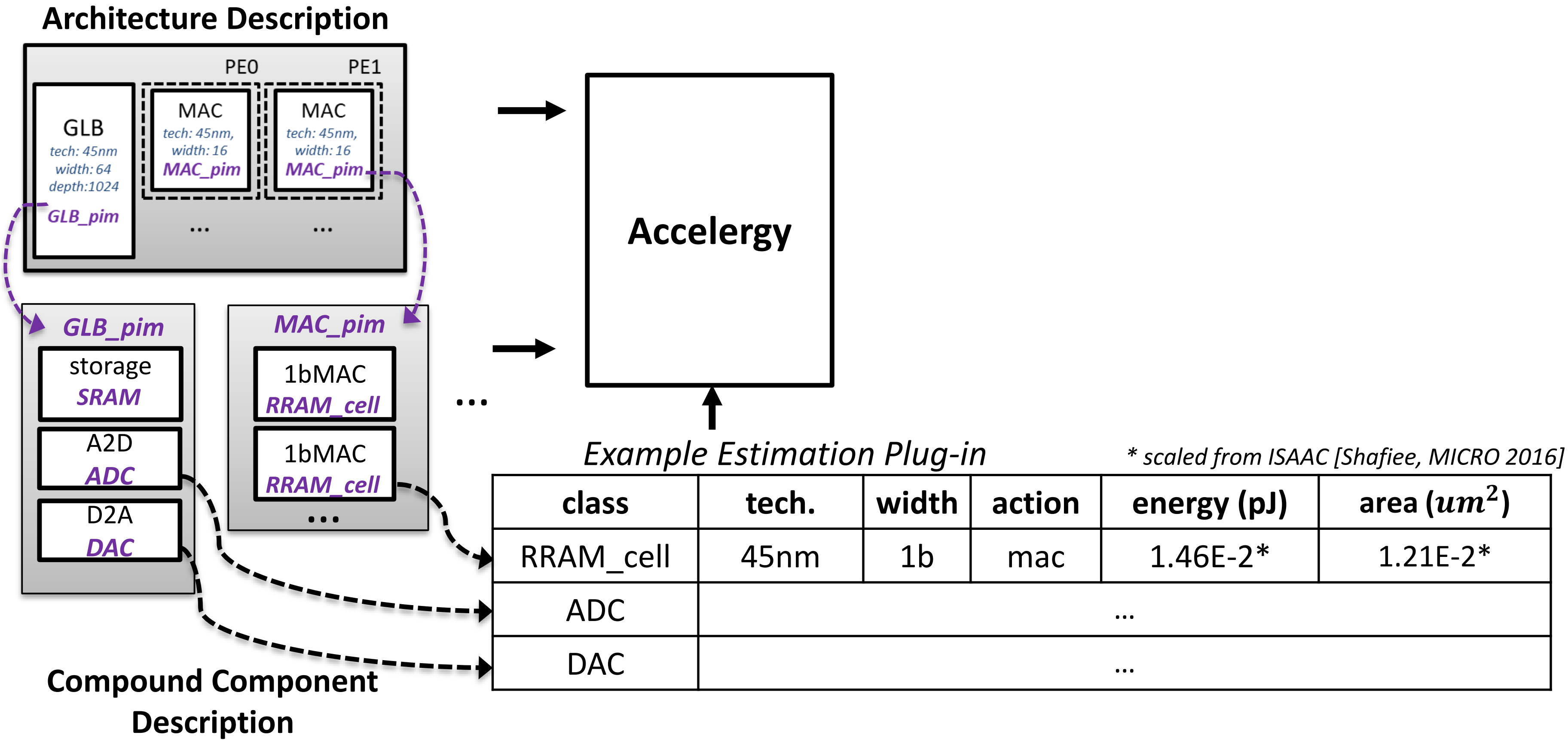
$I_2 = V_2 \times G_2$

$I = I_1 + I_2$
$= V_1 \times G_1 + V_2 \times G_2$

Estimation for PIM Accelerators



Estimation for PIM Accelerators

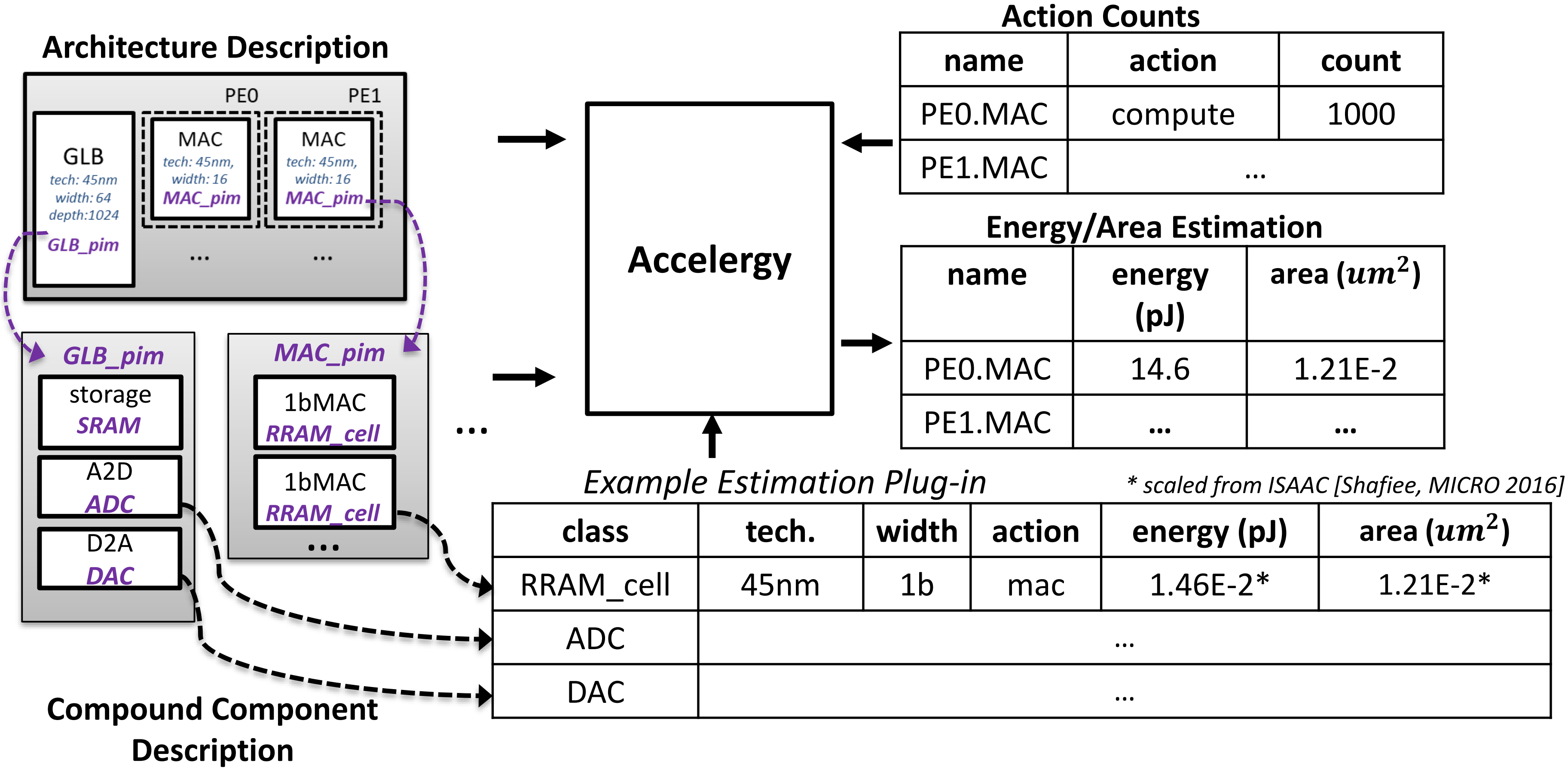


Example Estimation Plug-in

** scaled from ISAAC [Shafiee, MICRO 2016]*

class	tech.	width	action	energy (pJ)	area (um^2)
RRAM_cell	45nm	1b	mac	1.46E-2*	1.21E-2*
ADC	...				
DAC	...				

Estimation for PIM Accelerators



Action Counts

name	action	count
PE0.MAC	compute	1000
PE1.MAC	...	

Energy/Area Estimation

name	energy (pJ)	area (um^2)
PE0.MAC	14.6	1.21E-2
PE1.MAC	...	...

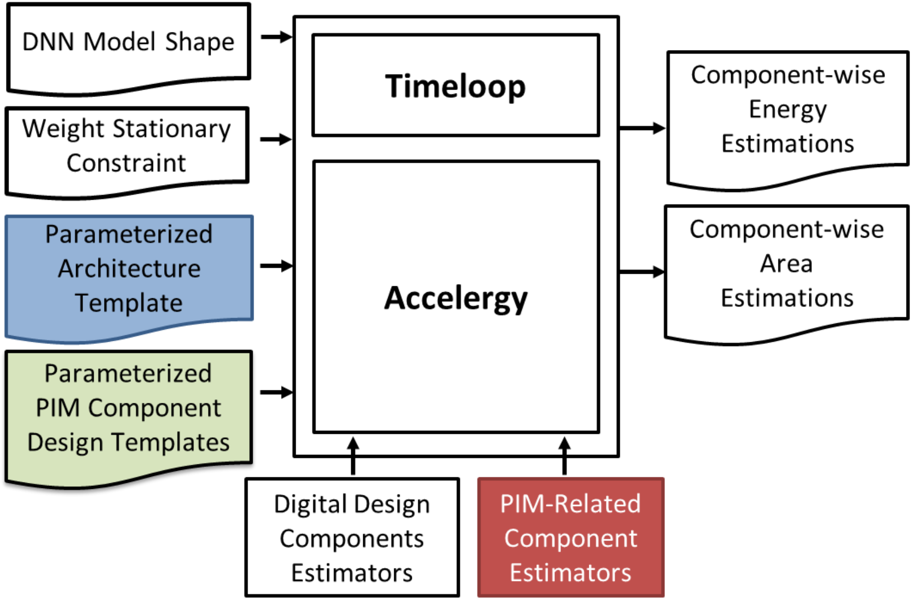
Example Estimation Plug-in

class	tech.	width	action	energy (pJ)	area (um^2)
RRAM_cell	45nm	1b	mac	1.46E-2*	1.21E-2*
ADC	...				
DAC	...				

** scaled from ISAAC [Shafiee, MICRO 2016]*

Accelergy Modeling of PIM Architectures

- **Parameterizable templates**
  - **Architecture Template allows architecture parameter sweeping, e.g.,**
    - number of PE rows
    - number of PE columns
    - size of global buffer, etc.
  - **Component design template allows implementation optimization, e.g.,**
    - optimize DAC-based D2A conversion system
    - optimize the design of the flash ADC in the A2D conversion system, etc.

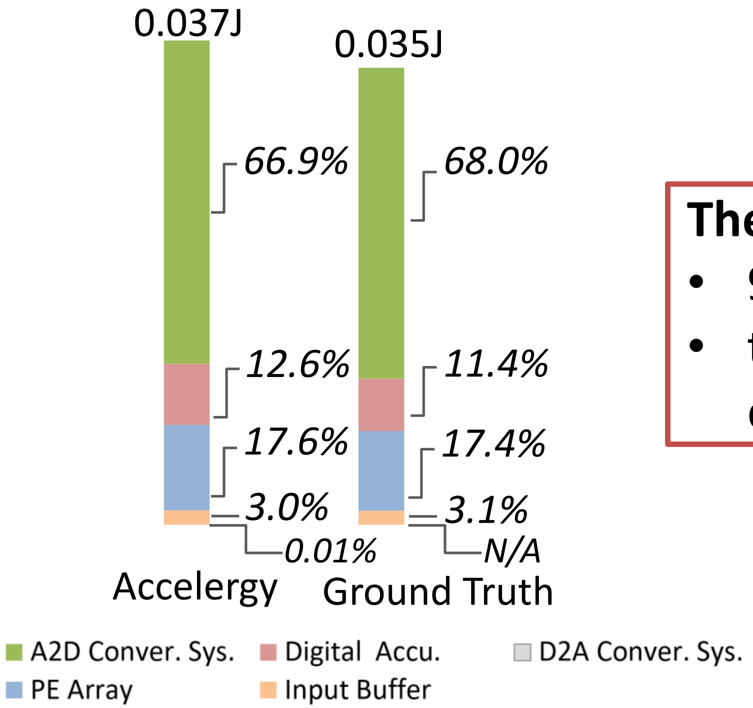

DNN Model Shape
Weight Stationary Constraint
Parameterized Architecture Template
Parameterized PIM Component Design Templates
Timeloop
Accelergy
Component-wise Energy Estimations
Component-wise Area Estimations
Digital Design Components Estimators
PIM-Related Component Estimators

Energy Modeling Validation on PIM Design

- Validation on the ADC-based design proposed in CASCADE [Chou, MICRO2019]
- Design Specs
  - 80 64x64 1-bit Memristor Arrays
  - 1-bit DACs
  - 6-bit ADCs
  - 16-bit data representations
- Workload: VGG Net convolutional layers
- Energy estimation tables: extracted numbers from the paper/cited sources

Energy Modeling Validation on PIM Design

Total Energy Estimation and Breakdown Validation



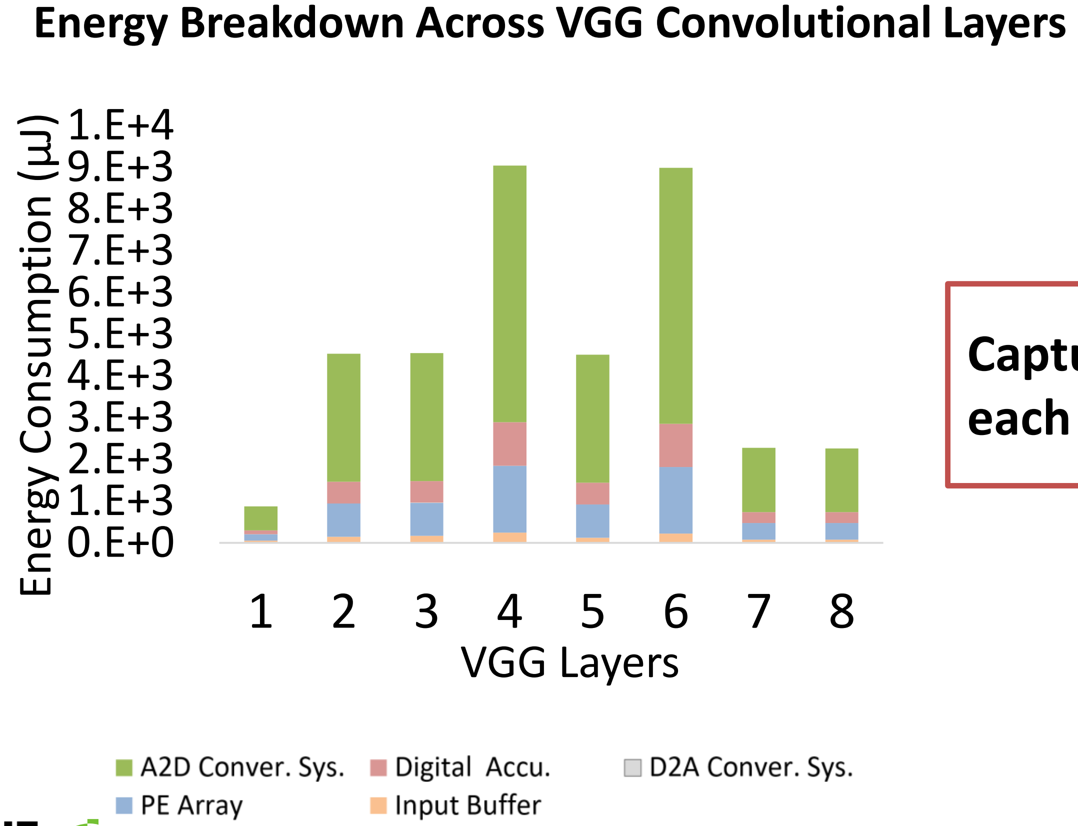
The architecture is correctly modeled:

- 95% accurate total energy estimation
- tracks the breakdown across different components

Published at [Wu, ISPASS 2020]

Total energy might not add up to exact 100.0% due to rounding

Energy Modeling Validation on PIM Design



Captures the energy breakdown of each convolutional layer

Published at [Wu, ISPASS 2020]

Summary

- Accelergy is an architecture-level energy estimator that
  - Accelerates accelerator design space exploration
  - Provides flexibility to
    - Describe and evaluate a wide range of accelerator designs
    - Support different technologies with user defined plug-ins, e.g., CMOS, RRAM, etc.
  - Achieves high accuracy energy estimations
    - 95% accurate for the Eyeriss accelerator and Cascade PIM accelerator
- The Timeloop-Accelergy system allows fast explorations on
  - High-level architecture properties, e.g., PE array size
  - Lower-level implementation optimizations on the components in the design, e.g., storage designs with local address generation

Acknowledgement: DARPA, Facebook, MIT Presidential Fellowship

Resources

- **Tutorial Related**
  - **Tutorial Website: http://accelergy.mit.edu/isca20_tutorial.html**
  - **Tutorial Docker: https://github.com/Accelergy-Project/timeloop-accelergy-tutorial**
    - **Various exercises and example designs and environment setup for the tools**
- **Other**
  - **Infrastructure Docker: https://github.com/Accelergy-Project/accelergy-timeloop-infrastructure**
    - **Pure environment setup for the tools without exercises and example designs**
  - **Open Source Tools**
    - **Accelergy: http://accelergy.mit.edu/**
    - **Timeloop: https://github.com/NVlabs/timeloop**
  - **Papers:**
    - A. Parashar, et al. "Timeloop: A systematic approach to DNN accelerator evaluation," ISPASS, 2019.
    - Y. N. Wu, V. Sze, J. S. Emer, “An Architecture-Level Energy and Area Estimator for Processing-In-Memory Accelerator Designs,” *ISPASS,* 2020.
    - Y. N. Wu, J. S. Emer, V. Sze, “Accelergy: An Architecture-Level Energy Estimation Methodology for Accelerator Designs,” *ICCAD*, 2019.