tinyML® Asia

Enabling Ultra-low Power Machine Learning at the Edge

November 29 & 30, 2022

www.tinyML.org
An All-Digital Reconfigurable SRAM-Based Compute-in-Memory Macro for TinyML Devices

Runxi Wang, Xinfei Guo
U of Michigan – Shanghai Jiao Tong University Joint Institute
Shanghai Jiao Tong University, China
Nov. 2022
Outline

- Motivations
- Design Part I – A Multi-Logic SRAM Cell
- Design Part II – A Pipelined Bit-Serial Addition Supported Computing Peripheral
- Evaluations
- Summary
Intro – AI is ‘still’ booming…

Sources
- https://www.javatpoint.com/application-of-ai
- https://www.grandviewresearch.com/industry-analysis/artificial-intelligence-ai-market
Motivation I – TinyML on Edge

TinyML on Edge

- Limited hardware resources and power supply
- AI algorithms are more complex and more diverse
- Compute with data capture on edge
Motivation I – TinyML on Edge

- Edge device
- Edge data
- Edge AI model
- Out

Compute with data capture on edge

Limited hardware resources and power supply

AI algorithms are more complex and more diverse

TinyML on Edge
Motivation I – TinyML on Edge

- Edge device
- Edge data
- Edge AI model
- Out

Limited hardware resources and power supply

AI algorithms are more complex and more diverse

Compute with data capture on edge

TinyML on Edge

Save time and energy compared with cloud computing
Motivation I – TinyML on Edge

Limited hardware resources and power supply

AI algorithms are more complex and more diverse

TinyML on Edge

Compute with data capture on edge
Motivation I – TinyML on Edge

TinyML on Edge

- Limited hardware resources and power supply
  - Model scale and computing capability are limited
- AI algorithms are more complex and more diverse
- Compute with data capture on edge
Motivation I – TinyML on Edge

Limited hardware resources and power supply

Compute with data capture on edge

AI algorithms are more complex and more diverse

TinyML on Edge

Motivation II – TinyML on Edge

AI algorithms are more complex and more diverse

TinyML on Edge

Limited hardware resources and power supply

Compute with data capture on edge

AI algorithms are more complex and more diverse

TinyML on Edge
Motivation I – TinyML on Edge

TinyML on Edge

Limited hardware resources and power supply

AI algorithms are more complex and more diverse

Compute with data capture on edge

TinyML Asia 2022
Motivation I – TinyML on Edge

Limited hardware resources and power supply

Compute with data capture on edge

AI algorithms are more complex and more diverse

Need a way to both store and compute the data efficiently
Motivation II – TinyML & CIM

Conventional arch
e.g. Von Neumann arch

Computing Unit

frequent
data transfer

Data Storage

- cache
- main memory
- disk
Motivation II – TinyML & CIM

Conventional arch e.g. Von Neumann arch

- Computing Unit
- Data Storage
  - cache
  - main memory
  - disk

frequent data transfer

SRAM-based compute-in-memory (CIM) design

- Computing Unit
- Data Storage
  - SRAM-CIM
  - main memory
  - disk

less data transfer

Input Peripherals

Cell

Cell

Cell

Array

Output Peripherals

Cell

Cell

Cell
**Motivation II – TinyML & CIM**

Conventional arch e.g. Von Neumann arch

- Computing Unit
- Data Storage
  - cache
  - main memory
  - disk

frequent data transfer

SRAM-based compute-in-memory (CIM) design

- Computing Unit
- Data Storage
  - SRAM-CIM
  - main memory
  - disk

less data transfer

Lower energy consumption!

Lower latency!

Physically useful for NN calculations!!
Motivation III – Performance vs. Flexibility

- General-purpose processor
- Reconfigurable accelerator
- Fully customized accelerator

Flexibility vs. Cost & performance
Motivation III – Performance vs. Flexibility

Previous works mainly focus on

Single-level CIM design

or

Small reconfigurability scope

System

Architecture

Memory Array with Peripheral Circuit

Memory Cell
Motivation III – Performance vs. Flexibility

Previous works mainly focus on Single-level CIM design or Small reconfigurability scope.

This work focuses on Cross hardware level reconfigurability!
Design Part I – Multi-Logic SRAM Cell

WL separated 6T SRAM
Columnly shared computing logic

<table>
<thead>
<tr>
<th>Operation</th>
<th>Input</th>
<th>Q</th>
<th>RWL_PX</th>
<th>RWL_NX</th>
<th>RBLX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>AND</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>OR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>XOR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

TinyML Asia 2022
Design Part I – Multi-Logic SRAM Cell

WL separated 6T SRAM Columnly shared computing logic

<table>
<thead>
<tr>
<th>Operation</th>
<th>Input</th>
<th>Q</th>
<th>RWL_PX</th>
<th>RWL_NX</th>
<th>RBLX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read</td>
<td>- 0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>- 1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>AND</td>
<td>0 0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0 1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1 0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1 1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>OR</td>
<td>0 0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1 0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1 1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>XOR</td>
<td>0 0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1 0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1 1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Design Part I – Multi-Logic SRAM Cell

WL separated 6T SRAM
Columnly shared computing logic

<table>
<thead>
<tr>
<th>Operation</th>
<th>Input</th>
<th>Q</th>
<th>RWL_PX</th>
<th>RWL_NX</th>
<th>RBLX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>AND</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>OR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

TinyML Asia 2022
Design Part I – Multi-Logic SRAM Cell

WL separated 6T SRAM
Columnly shared computing logic

<table>
<thead>
<tr>
<th>Operation</th>
<th>Input</th>
<th>Q</th>
<th>RWL_PX</th>
<th>RWL_NX</th>
<th>RBLX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>AND</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>OR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>XOR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Support 3 Boolean operations
AND/OR/XOR with low area overheads
Design Part II – Reconfig. Computing Peripherals

Peripherals for Addition

Peripherals for Default Storage Function
Design Part II – Reconfig. Computing Peripherals
Design Part II – Reconfig. Computing Peripherals

Peripherals for Addition

Peripherals for Default Storage Function
Design Part II – Reconfig. Computing Peripherals

Peripherals for Addition

Peripherals for Default Storage Function

Intelligent Circuits, Architectures, and Systems (ICAS)
TinyML Asia 2022
Design Part II – Reconfig. Computing Peripherals

Support concurrent reading

Peripherals for Addition

Peripherals for Default Storage Function
Support concurrent reading

Support concurrent reading

Pipelined bit-serial addition

Design Part II – Reconfig. Computing Peripherals

Support concurrent reading

Pipelined bit-serial addition

✓ Lower latency
✓ Beneficial for accumulating process in MAC
✓ Components can be reused to do other Op.
Evaluation – Experiment Setup

- Simulation based evaluations
- Synthesis/Simulation tools: Genus, Virtuoso
- Technology node: TSMC28nm
Evaluation – Multi-Logic Cells

Input+config

Input+config

Output
Evaluation – Computing Peripherals

- Frequency: 3.125GHz
- Latency: Within 4 cycles
- Energy consumption: 0.041pJ
- Computing peripheral cell area: 12.348μm²
Summary

- Memory Cell
- Memory Array with Peripheral Circuit
- Architecture
- System
Summary

Multi-logic SRAM cell
Summary

Multi-logic SRAM cell

Reconfigurable computing peripheral

Memory Array with Peripheral Circuit

Memory Cell

System

Architectures, and Systems (ICAS)
Compared with previous works: More flexible with smaller overheads!
Summary

Compared with previous works:
More flexible with smaller overheads!

Future work:
➢ Explore more arithmetic Op. to be supported
➢ Extend the reconfigurability to architecture and system level

- Reconfigurable computing peripheral
- Multi-logic SRAM cell
- Memory Array with Peripheral Circuit
- Memory Cell
- System
Thank you!

https://sites.ji.sjtu.edu.cn/icas/

Contact Info

xinfei.guo@sjtu.edu.cn

+86-21-3420-6045
Ext. 4191

www.xinfeiguo.com

Office 419,
Longbin Building, Shanghai Jiao Tong University

Lab 212-11B,
Longbin Building, Shanghai Jiao Tong University
Copyright Notice

This presentation in this publication was presented at tinyML® Asia 2022. The content reflects the opinion of the author(s) and their respective companies. The inclusion of presentations in this publication does not constitute an endorsement by tinyML Foundation or the sponsors.

There is no copyright protection claimed by this publication. However, each presentation is the work of the authors and their respective companies and may contain copyrighted material. As such, it is strongly encouraged that any use reflect proper acknowledgement to the appropriate source. Any questions regarding the use of any materials presented should be directed to the author(s) or their companies.

tinyML is a registered trademark of the tinyML Foundation.

www.tinyML.org