The Era of LLMs on ASICs

Locally Intelligent Machines

Gregory Kielian
May, 2024
Google Research

Outline

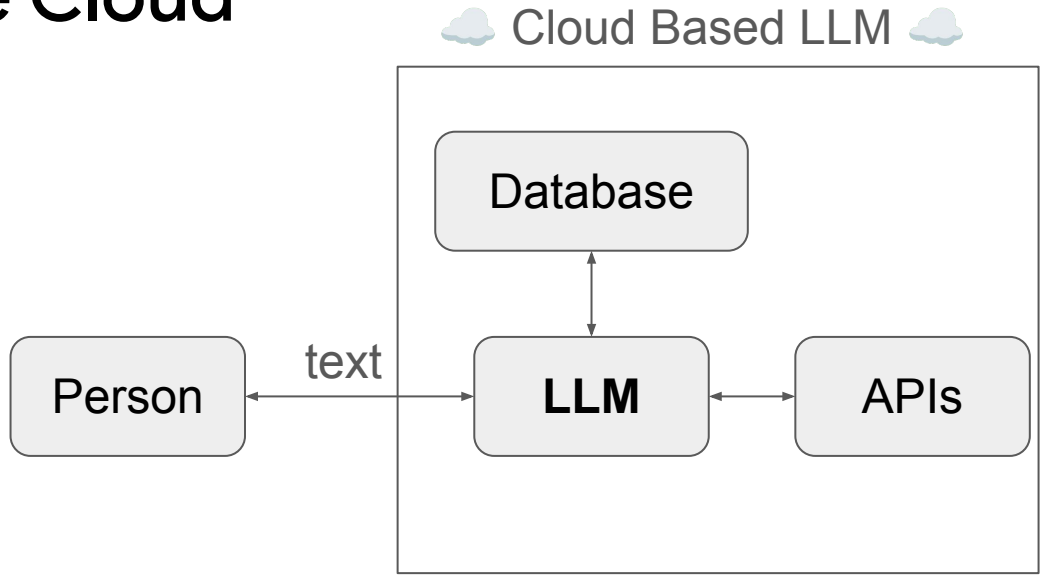
- Motivation
- Hardware Bottlenecks
- Directions and Opportunities for Collaboration

LLMs + Hardware = AI Robots

- Memory
- Multimodality
- Speech Capability

LLMs Currently in the Cloud

- AI Currently Resides in Remote Servers

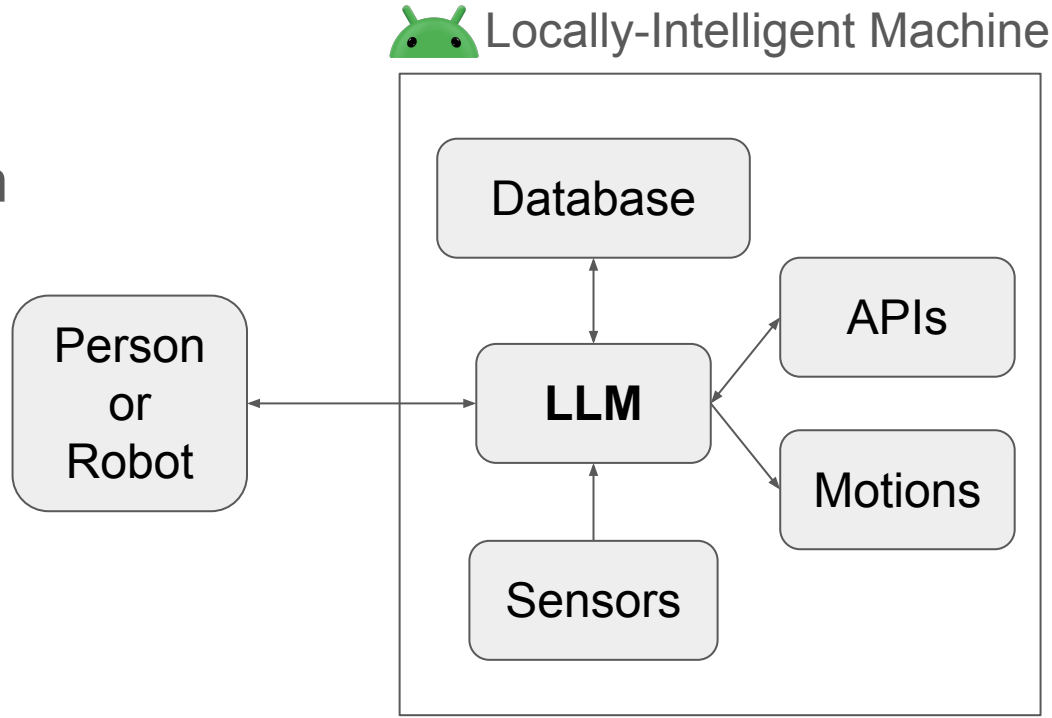


Simplified Modern LLM Diagram

Local Intelligence Enables *Future* Applications

Benefits:

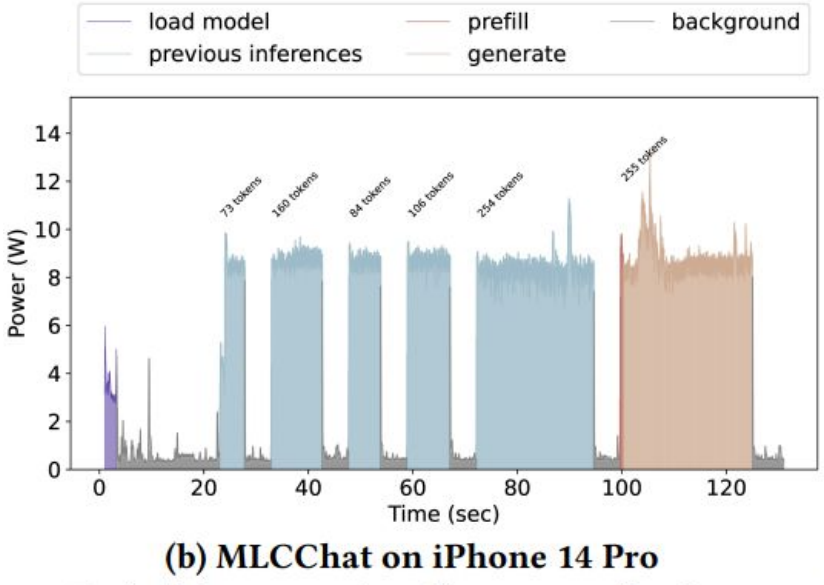
- Internet-Free Operation
- Real-Time Interaction
- Physical Interfaces



Enabling LLMs on the Edge

Efficiency Needed - Approaches From Both HW and SW

An IPhone 14 47.9C (118.22F) after just a 2 minute conversation with Zephyr-3B-4bit at 10 tokens/sec



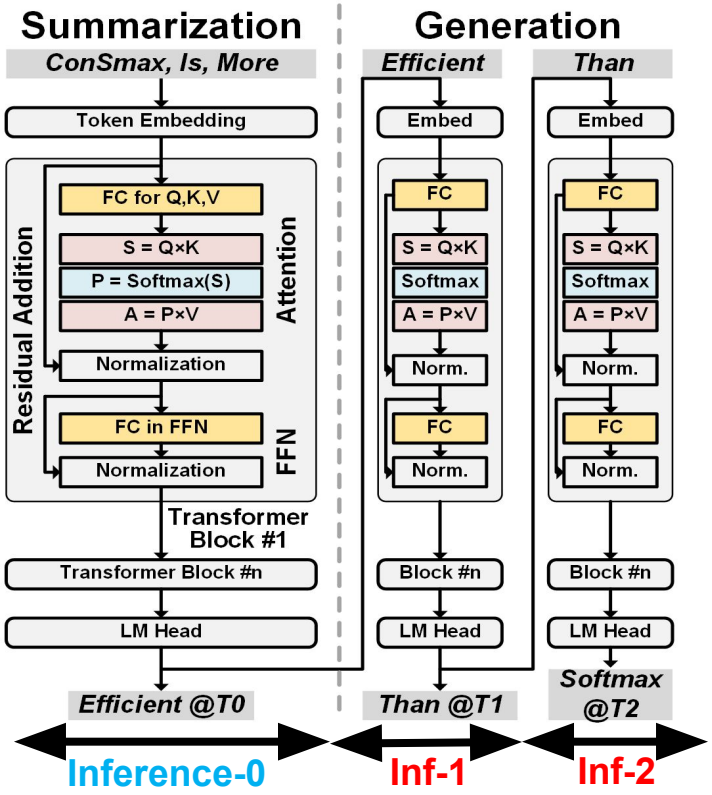
(b) MLCChat on iPhone 14 Pro

Present Hardware Cannot Run LLMs Efficiently

Google Research

Edge AI LLM Accelerators

❑ Summarization and Generation Stage in LLM

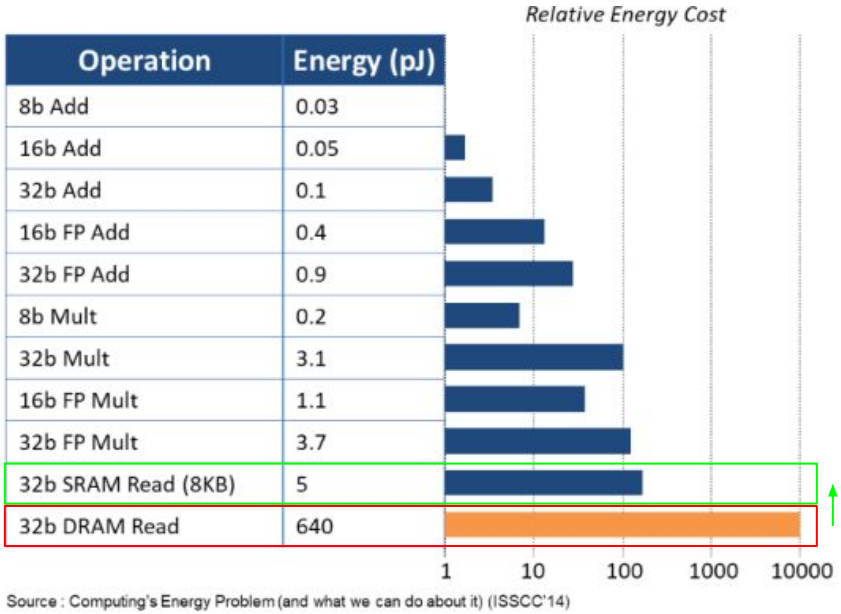


- Summarization Stage:
  - ❑ Summarize and extract features from input prompt.
  - ❑ Model receives multiple input tokens from the given prompt.
  - ❑ Compute-Bounded by general matrix-matrix multiplication.
- Generation Stage:
  - ❑ Generate new tokens.
  - ❑ Each inference produces one new token.
  - ❑ Memory-Bounded by general matrix-vector multiplication at small batch size (= 1).

Google Research

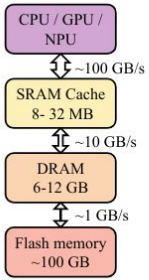
Memory Efficiency

DRAM and Off-Chip Memory Reads Largest Energy Sink



Operation	Energy (pJ)
8b Add	0.03
16b Add	0.05
32b Add	0.1
16b FP Add	0.4
32b FP Add	0.9
8b Mult	0.2
32b Mult	3.1
16b FP Mult	1.1
32b FP Mult	3.7
32b SRAM Read (8KB)	5
32b DRAM Read	640

Source : Computing's Energy Problem (and what we can do about it) (ISSCC'14)



Hardware	Device	SoC last level memory size	DRAM size
Apple A16	iPhone 15	24 MB	6 GB
Apple A15	iPhone 14	32 MB	6 GB
Google	Pixel 8	8 MB	8 GB / 12 GB (pro)
QCOM	Snapdragon 8	10 MB	8-12 GB

Figure 2: Memory hierarchy in prevalent mobile devices. Despite adequate Flash storage, the operational memory for executing high-speed applications predominantly resides in DRAM, typically constrained to 6-12 GB.

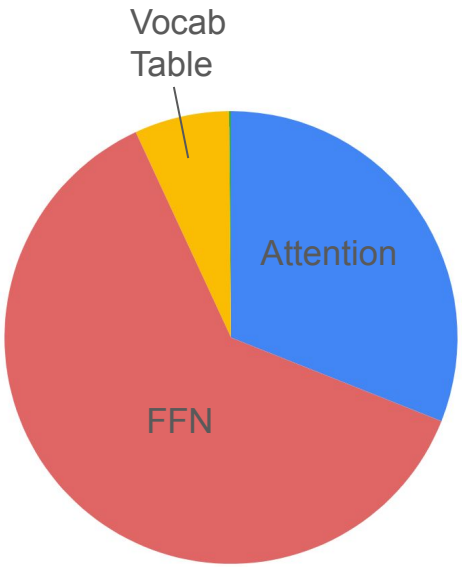
https://arxiv.org/abs/2402.14905

Iphone 14 47.9C (118.22F) after 2 minutes of conversation with Zephyr-3B-4bit at 10 tokens/sec

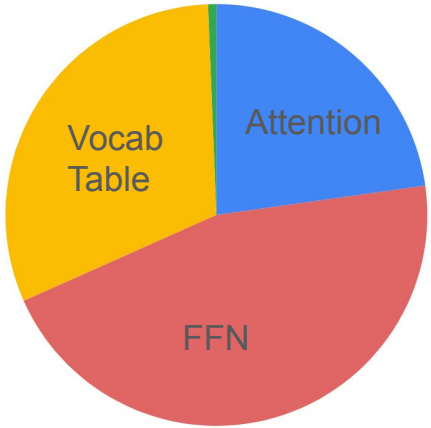
https://arxiv.org/pdf/2403.12844

https://github.com/karpathy/nanoGPT/blob/master/transformer_sizing.ipynb

Breakdown of Parameters are in the "Transformer Block"

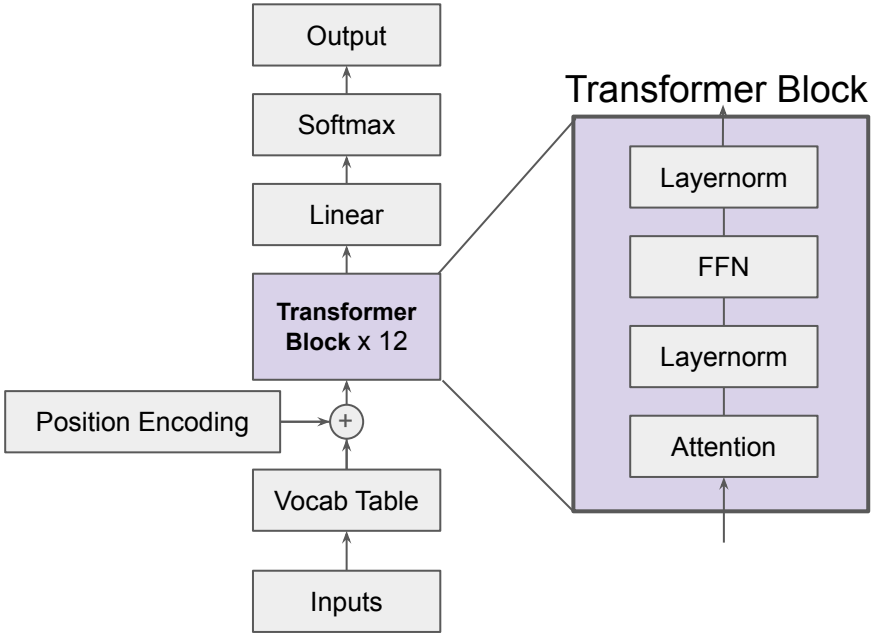


1.5B Parameters
GPT-2 XL



124M Parameters
GPT-2 Small

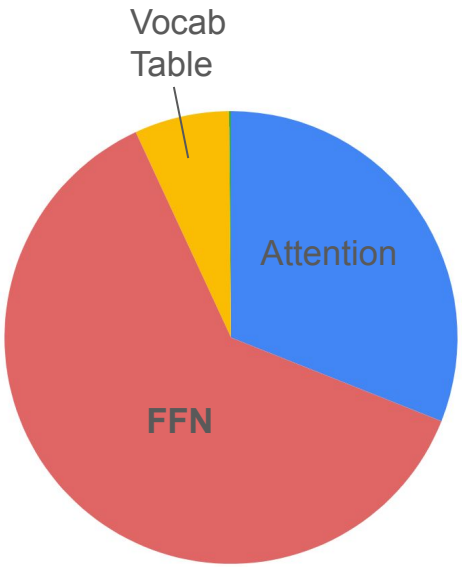
LLM Transformer Architecture



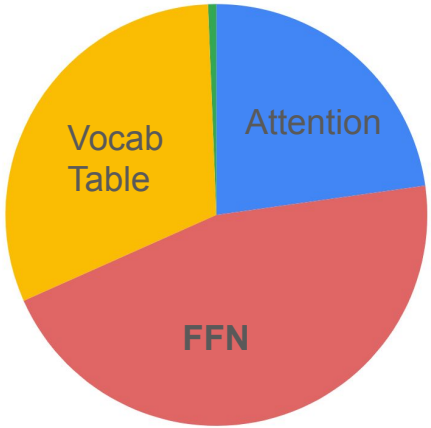
Google Research

https://github.com/karpathy/nanoGPT/blob/master/transformer_sizing.ipynb

FFN (aka MLP) Holds ~50-75% of Parameters

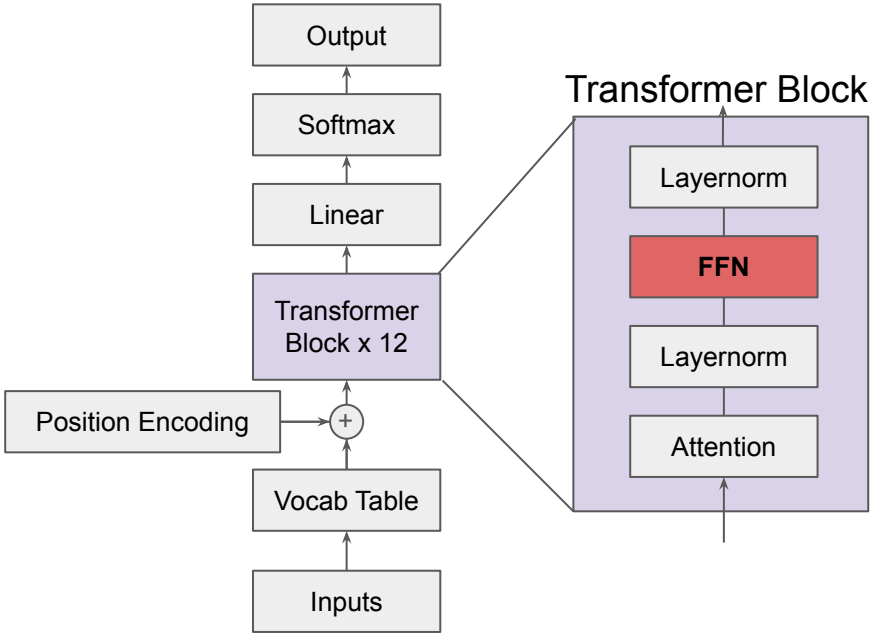


1.5B Parameters
GPT-2 XL



124M Parameters
GPT-2 Small

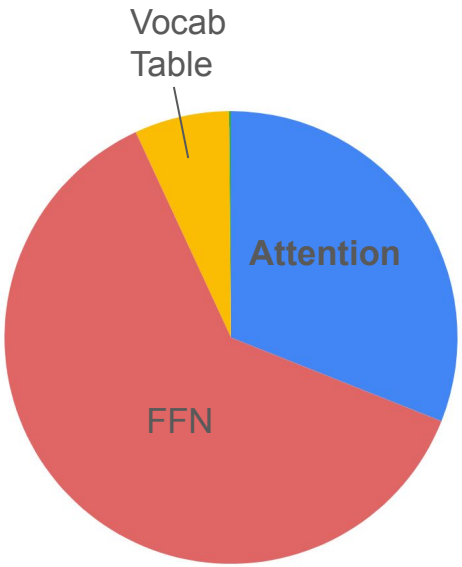
LLM Transformer Architecture



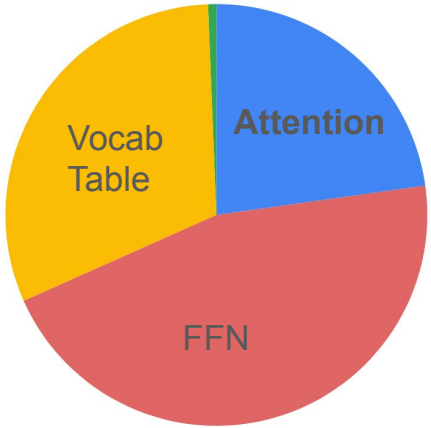
Google Research

https://github.com/karpathy/nanoGPT/blob/master/transformer_sizing.ipynb

Attention Layer Holds ~25-30% of Parameters

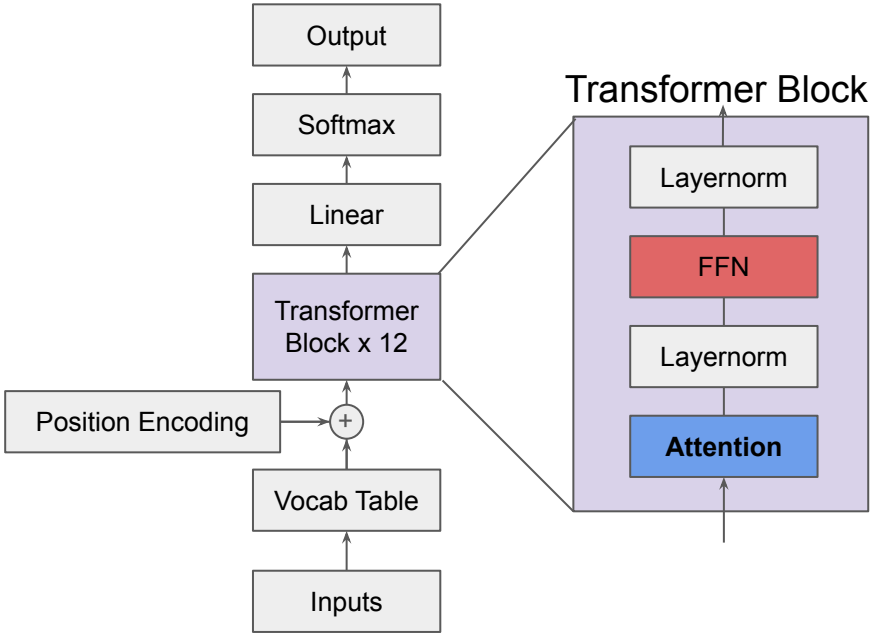


1.5B Parameters
GPT-2 XL



124M Parameters
GPT-2 Small

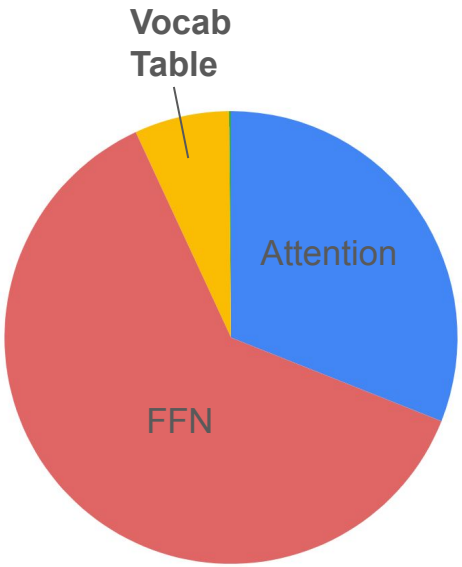
LLM Transformer Architecture



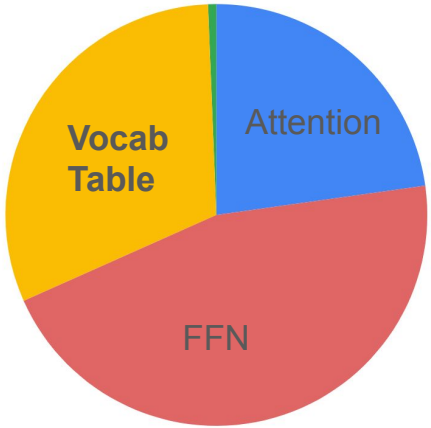
Google Research

https://github.com/karpathy/nanoGPT/blob/master/transformer_sizing.ipynb

Vocabulary Table, *25%+ for Smaller Networks*

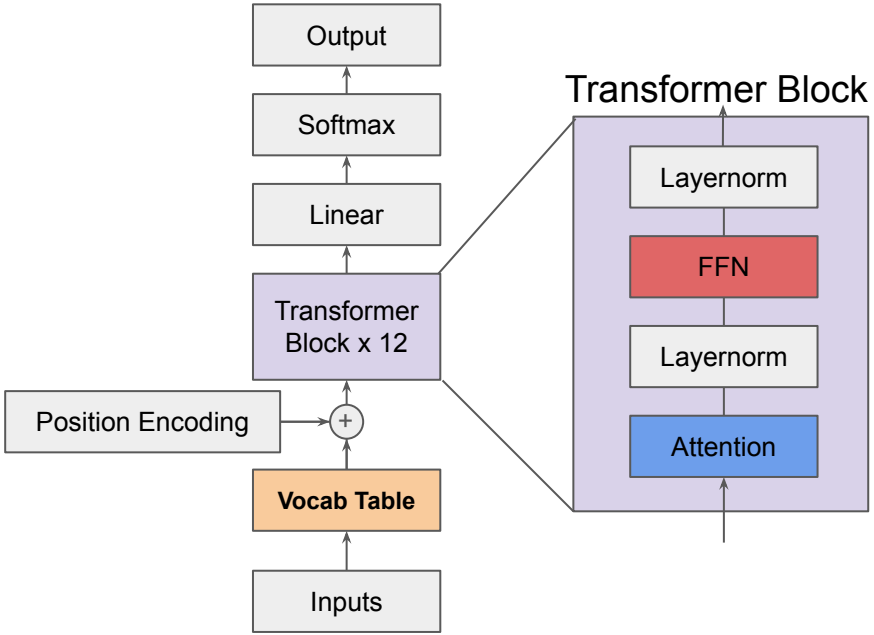


1.5B Parameters
GPT-2 XL



124M Parameters
GPT-2 Small

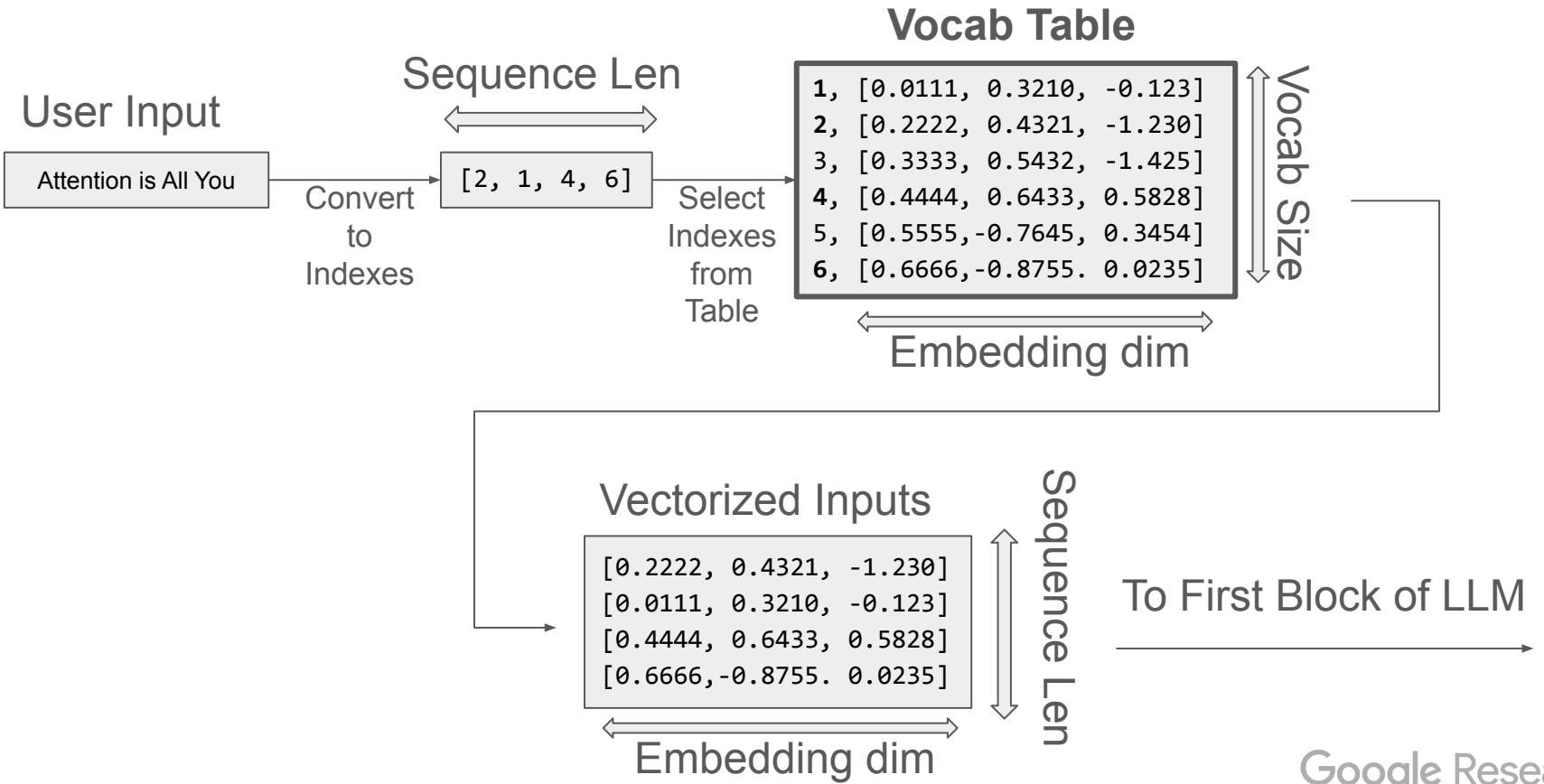
LLM Transformer Architecture



Google Research

Embedding Table Optimizations

A Close Look at the Input Stage of LLMs



Google Research

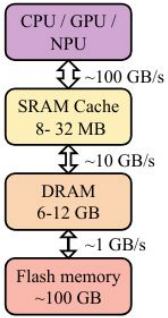
Vocab Table Size Estimation with Vocabulary of 50k

Vocab Table

```
  1, [0.0111, 0.3210, -0.123]
  2, [0.2222, 0.4321, -1.230]
  3, [0.3333, 0.5432, -1.425]
  4, [0.4444, 0.6433, 0.5828]
 …,              …
50k, [0.6666,-0.8755. 0.0235]
```

Vocab Size

Embedding dim



Hardware	Device	SoC last level memory size	DRAM size
Apple A16	iPhone 15	24 MB	6 GB
Apple A15	iPhone 14	32 MB	6 GB
Google	Pixel 8	8 MB	8 GB / 12 GB (pro)
QCOM	Snapdragon 8	10 MB	8-12 GB

Figure 2: Memory hierarchy in prevalent mobile devices. Despite adequate Flash storage, the operational memory for executing high-speed applications predominantly resides in DRAM, typically constrained to 6-12 GB.

https://arxiv.org/abs/2402.14905

Vocab Table Size = Vocab_Size * Embedding Vector Length * Bytes Per Param

GPT-2 Small Vocab Table Size, 8bit Quant = 50,000 * 768 * 1 = 38.59 MB

Reduced Vocabulary -> **More Efficiency**

name	params	ratio (%)
embedding/position	98304	0.3286
embedding/token	19200000	64.1708
embedding	19298304	64.4994
attention/ln	384	0.0013
attention/kqv	442368	1.4785
attention/proj	147456	0.4928
attention	590208	1.9726
mlp/ln	384	0.0013
mlp/ffw	589824	1.9713
mlp/proj	589824	1.9713
mlp	1180032	3.9439
block	295104	0.9863
transformer	10621440	35.4993
ln_f	384	0.0013
dense	0	0.0000
total	29920128	100.0000

name	params	ratio (%)
embedding/position	98304	0.9102
embedding/token	80640	0.7466
embedding	178944	1.6568
attention/ln	384	0.0036
attention/kqv	442368	4.0957
attention/proj	147456	1.3652
attention	590208	5.4645
mlp/ln	384	0.0036
mlp/ffw	589824	5.4609
mlp/proj	589824	5.4609
mlp	1180032	10.9254
block	295104	2.7323
transformer	10621440	98.3397
ln_f	384	0.0036
dense	0	0.0000
total	10800768	100.0000

50,000 Vocab Size → 210 Vocab Size

TikToken “gpt2” → Hiragana + Roman Alphabet

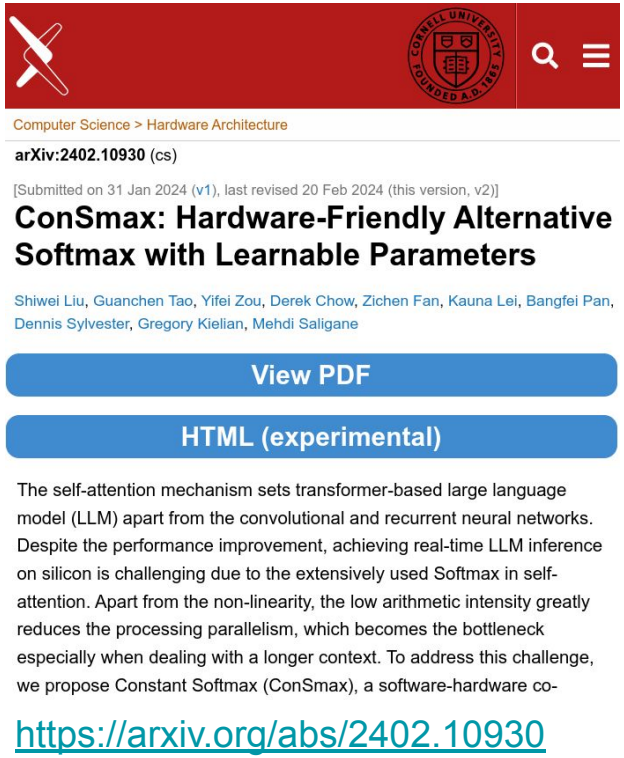
65% of Parameters → **1.7%** of Parameters

30M Params → **11M params**

Google Research

Hardware Efficient Softmax

Co-Design and Hardware And Software Collaborations



Computer Science > Hardware Architecture

arXiv:2402.10930 (cs)

[Submitted on 31 Jan 2024 (v1), last revised 20 Feb 2024 (this version, v2)]

ConSmax: Hardware-Friendly Alternative Softmax with Learnable Parameters

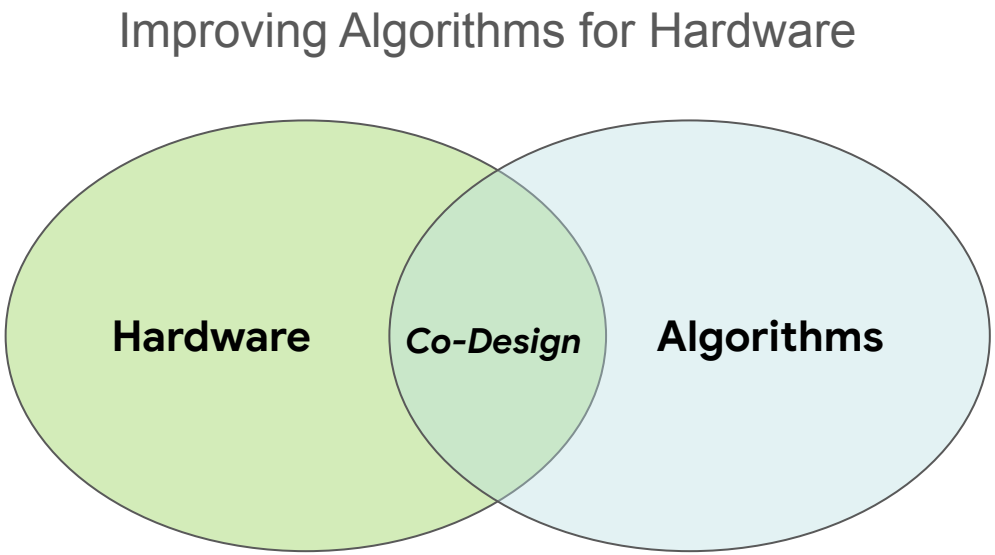
Shiwei Liu, Guanchen Tao, Yifei Zou, Derek Chow, Zichen Fan, Kauna Lei, Bangfei Pan, Dennis Sylvester, Gregory Kielian, Mehdi Saligane

View PDF

HTML (experimental)

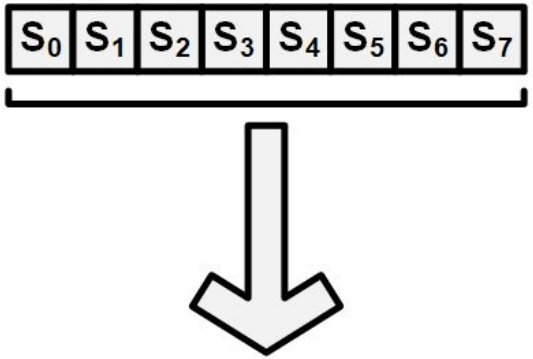
The self-attention mechanism sets transformer-based large language model (LLM) apart from the convolutional and recurrent neural networks. Despite the performance improvement, achieving real-time LLM inference on silicon is challenging due to the extensively used Softmax in self-attention. Apart from the non-linearity, the low arithmetic intensity greatly reduces the processing parallelism, which becomes the bottleneck especially when dealing with a longer context. To address this challenge, we propose Constant Softmax (ConSmax), a software-hardware co-

https://arxiv.org/abs/2402.10930



Optimization Achieved Via Parallelism

(a) Original Softmax



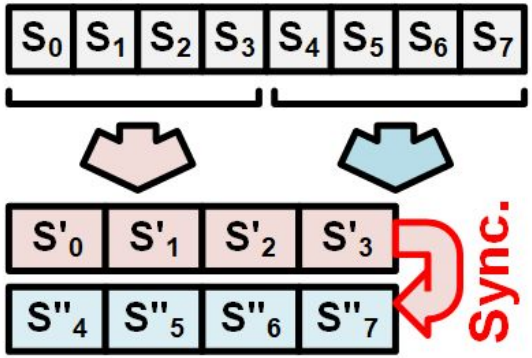
$$S_{MAX} = MAX(S_0, S_1, \dots, S_{N-1})$$

$$f(S_i) = e^{S_i - S_{MAX}}$$

$$Sum = \sum f(S_i)$$

$$Softmax = \frac{f(S_i)}{Sum}$$

(b) Partial Softmax



$$S_{MAX} = MAX(S'_{MAX}, S''_{MAX})$$

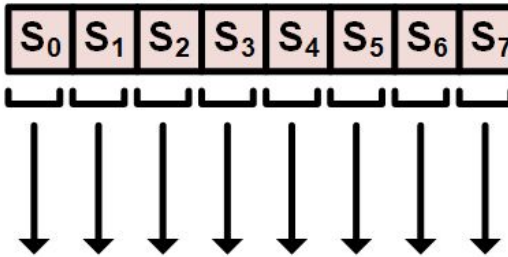
$$f(S'_i) = e^{S'_{MAX} - S_{MAX}} \times f(S'_i)$$

$$f(S''_i) = e^{S''_{MAX} - S_{MAX}} \times f(S''_i)$$

$$Sum = \sum f(S'_i) + \sum f(S''_i)$$

$$Softmax = \left[\frac{f(S'_i)}{Sum}, \frac{f(S''_i)}{Sum}\right]$$

(c) ConSmax



$$Softmax(S_i) = \frac{e^{S_i - \beta}}{\gamma}$$

β, γ are two constant values

Google Research

Hardware Efficient Normalization

The “Layernorm” Bottleneck

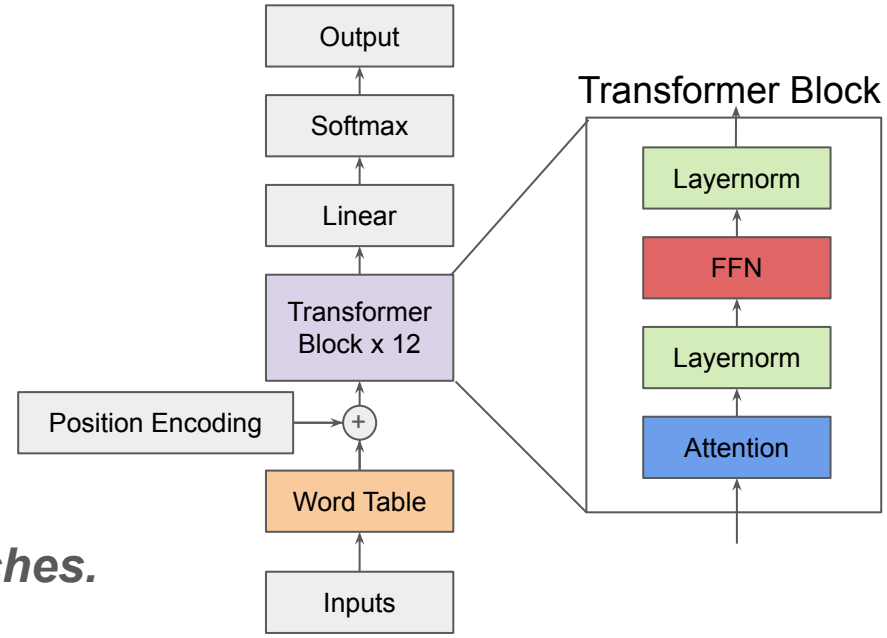
- **Normalizations**
  - Occur 2x per Transformer Block
  - Recenters and/or Rescales

Layernorm’s Mean StdDev Calculation

$$\bar{a}_i = \frac{a_i - \mu}{\sigma} g_i$$

$$\mu = \frac{1}{n}\sum_{i=1}^{n} a_i, \quad \sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(a_i - \mu)^2}$$

LLM Transformer Architecture

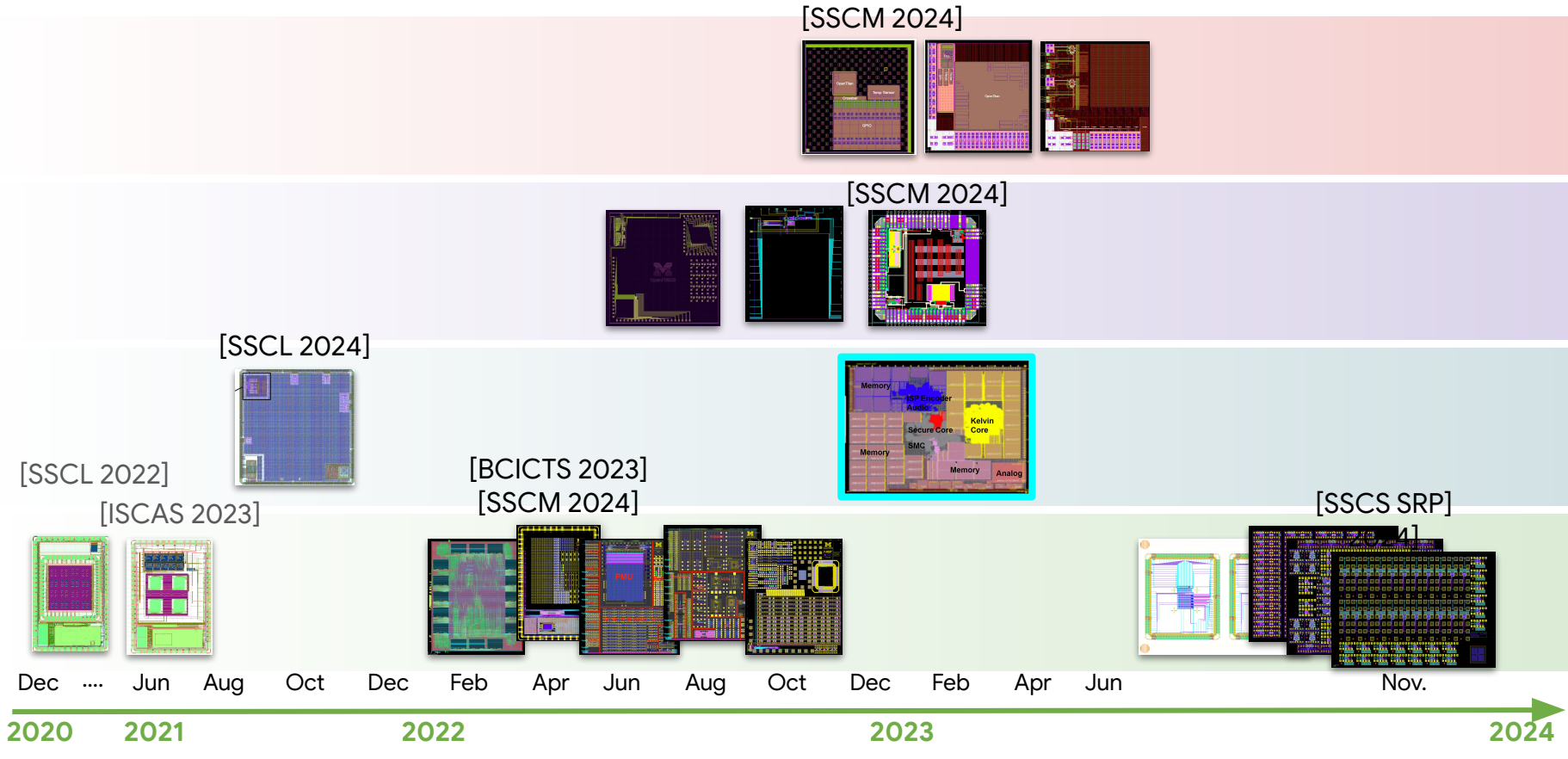


Several possibilities for better approaches.

https://arxiv.org/abs/1607.06450
https://arxiv.org/abs/1910.07467

Directions and Collaborations

ASICs are the Key to *Efficient* On Device Operation



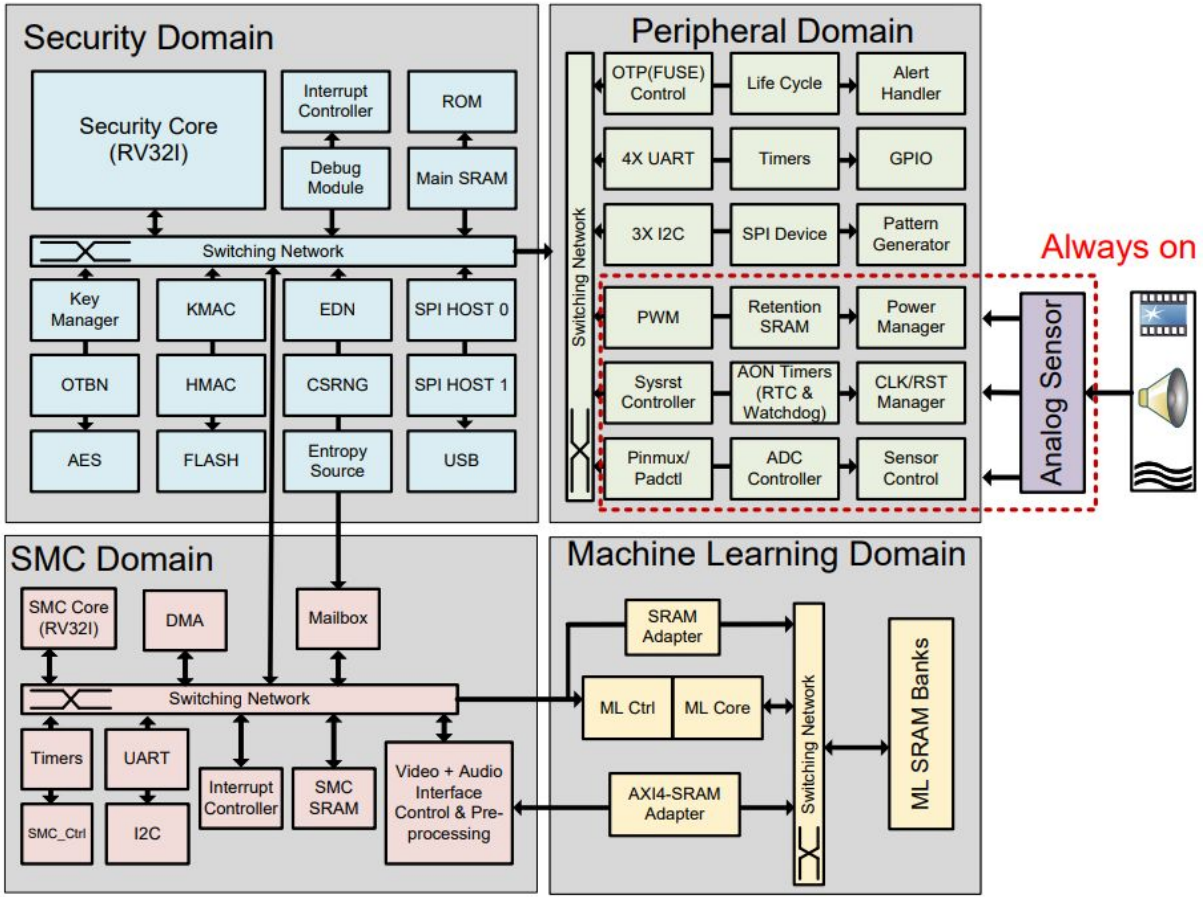
Dr. Saligane and team's collaborative tapeouts and partnerships

Google Research

Open Se Cura Framework

Google Research

Secure Edge AI SoC in GF 12nm using closed tools!



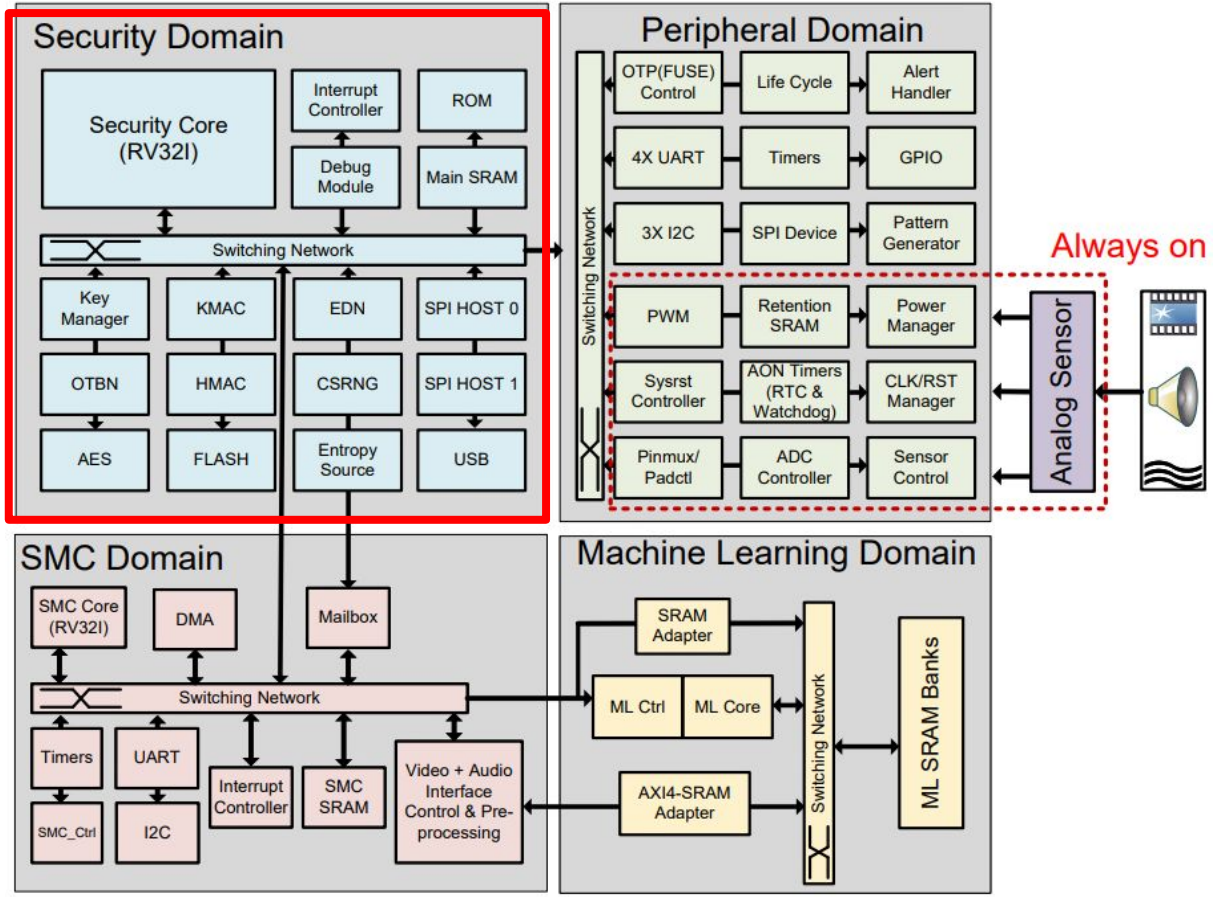
Google Research

Open Se Cura Framework

Google Research

Secure Edge AI SoC in GF 12nm using closed tools!

RoT SoC with Security Primitives



Google Research

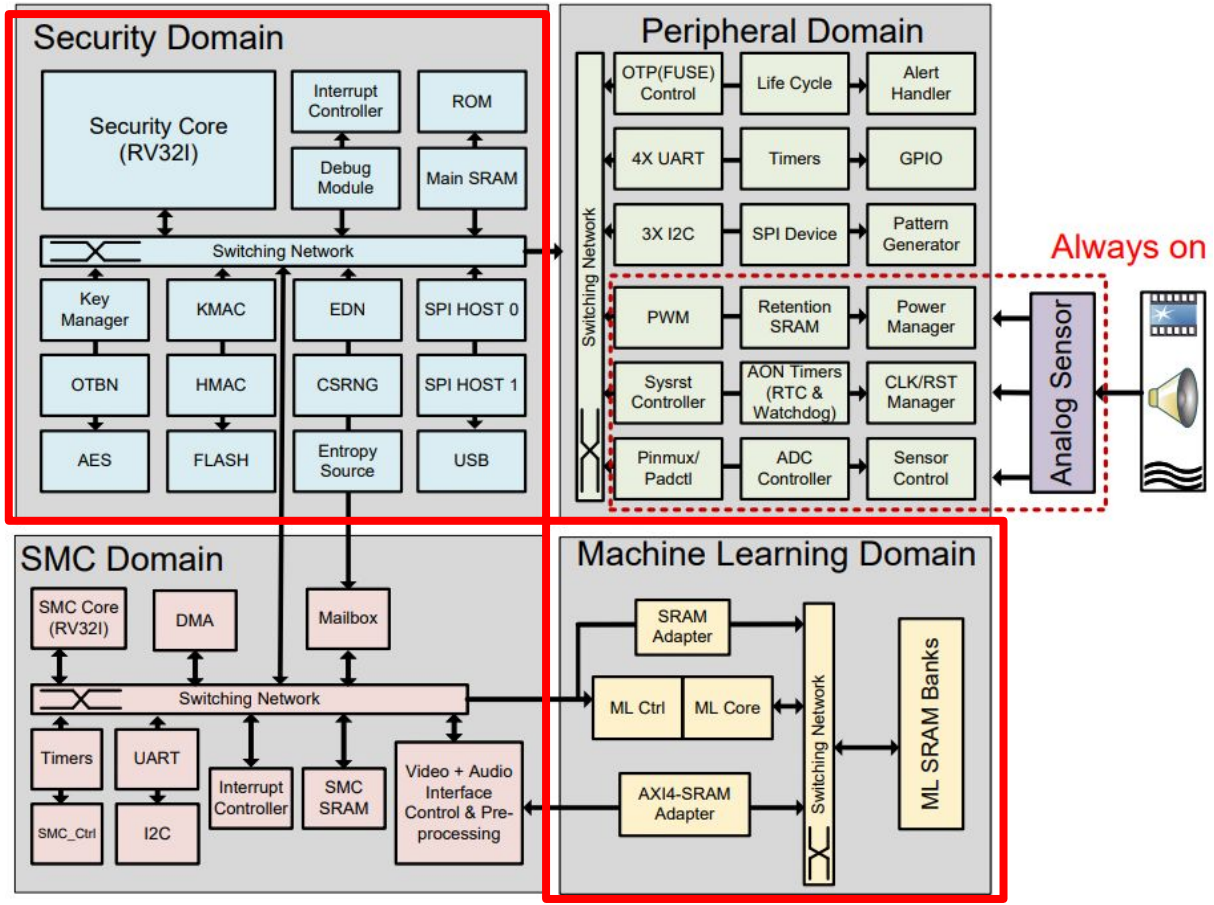
Open Se Cura Framework

Google Research

Secure Edge AI SoC in GF 12nm using closed tools!

RoT SoC with Security Primitives

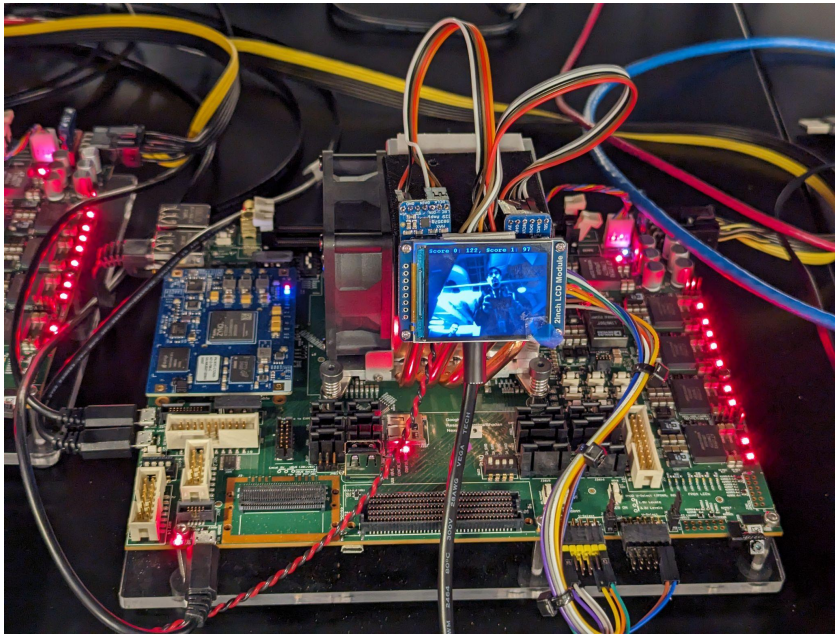
ML Core - Matrix Mult. Systolic Array Engine

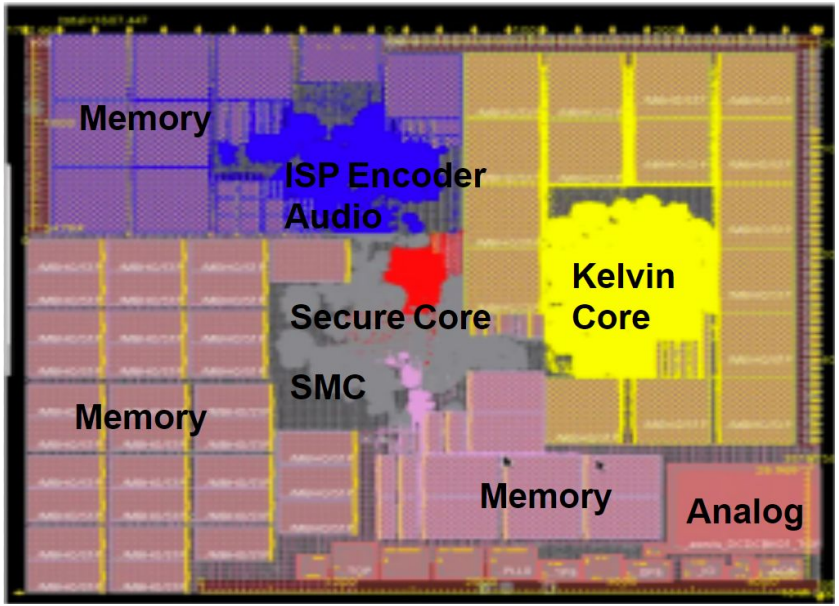


Google Research

Open Se Cura In GF 12nm

Google Research





FPGA-Based Emulation!

Google Research

Open-Source Foundations for ASIC *Collaborations*

README | Apache-2.0 license

OpenFASoC

OpenFASoC: Fully Open-Source Autonomous SoC Synthesis using Customizable Cell-Based Synthesizable Analog Circuits

docs failing

OpenFASOC is focused on open-source automated analog generation from user specification to GDSII with fully open-sourced tools. This project is led by a team of researchers at the University of Michigan and is inspired by FASoC, that sits on proprietary tools. (See more about FaSoC at website)

- *Temperature sensor -*

sky130hd_temp-sense-generator failing

Build and test with the latest version of tools set failing

Open in Colab

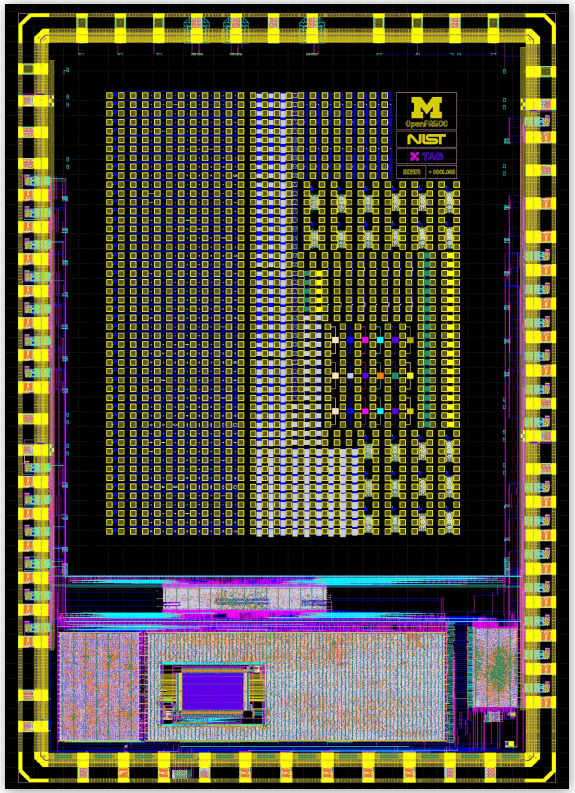
Contributors 33

+ 19 contributors

Languages

Python 39.6% Tcl 26.0%
SourcePawn 18.3%
Makefile 9.5% Verilog 4.1%
SystemVerilog 1.4%
Other 1.1%

https://github.com/idea-fasoc/OpenFASOC



Google Research

HW and Software Co-Design for LLM Silicon Development

ReaLLMASIC

Overview

ReaLLMAsic aims to bridge the gap between theoretical model design and practical hardware implementation, ensuring efficient, scalable, and robust ML model development.

Our project stands out for its extensive exploration of various model configurations and modules, catering to a diverse range of use cases.

Key exploration features include:

- `Module Variation`: Explore with different module types -- e.g. Softmax, Softermax, ConSmax, and SigSoftmax -- discover which is best suited (PPA) to your application.
- `Flexible Tokenization`: Explore different tokenization: tiktoken, sentencepiece, phonemization, character level, custom tokenization, etc.
- `Diverse Dataset Performance Testing`: Evaluate model efficacy across various languages and datasets including: csv-timeseries, mathematics, music, lyrics, literature, and webtext.
- `Standard and Custom Hyperparameters`: Fine-tune models using conventional hyperparameters and explore the impact of custom settings on model performance and PPA impacts.

ReaLLMASIC on Github

The Future is What We Make It

- Custom Silicon and EdgeLLM ASICs can make Robotics and AI Ubiquitous
- Partnerships are needed for bringing AI to Life.
- Let’s make the future!

Let’s build silicon together!