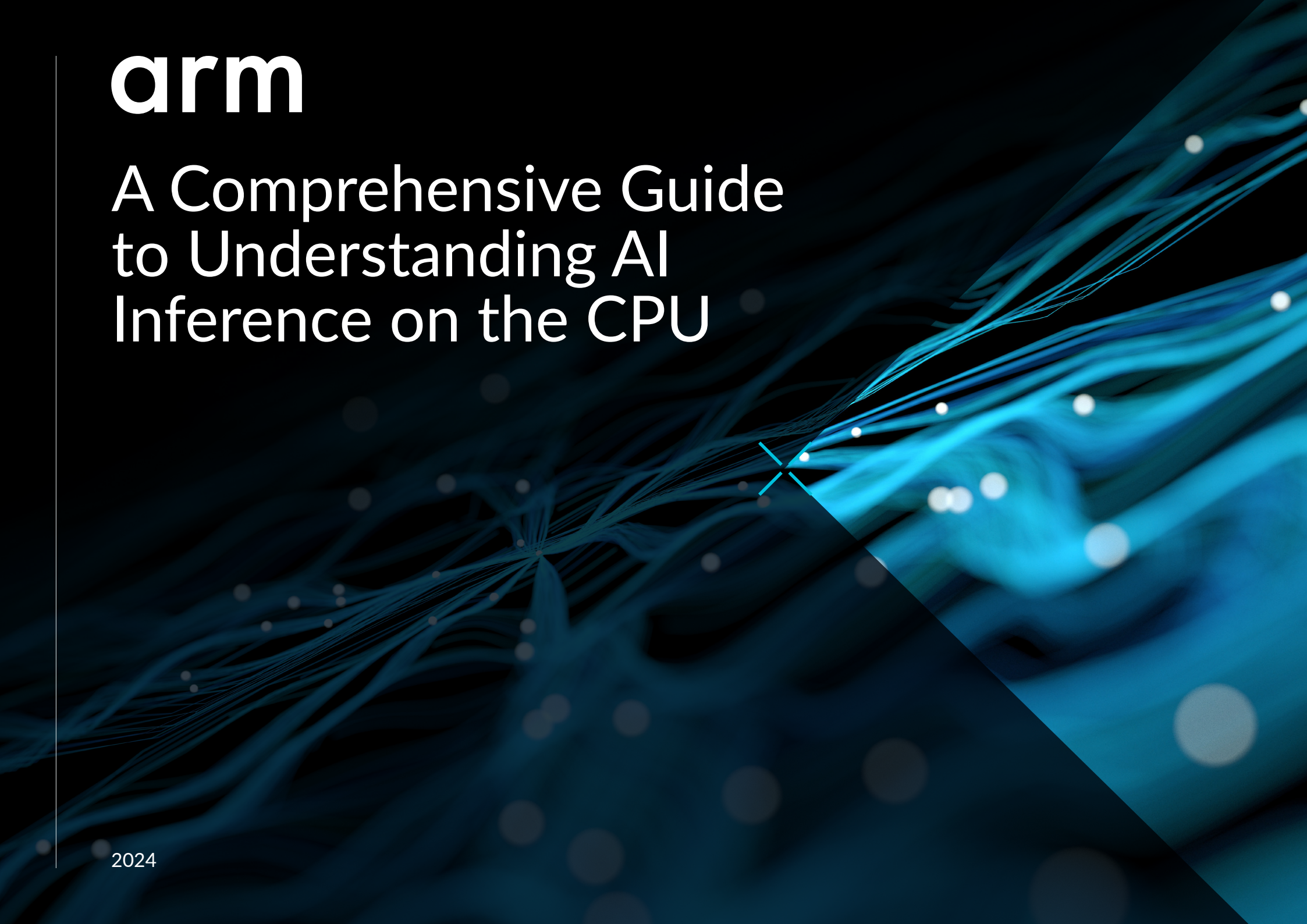
arm

A Comprehensive Guide to Understanding AI Inference on the CPU

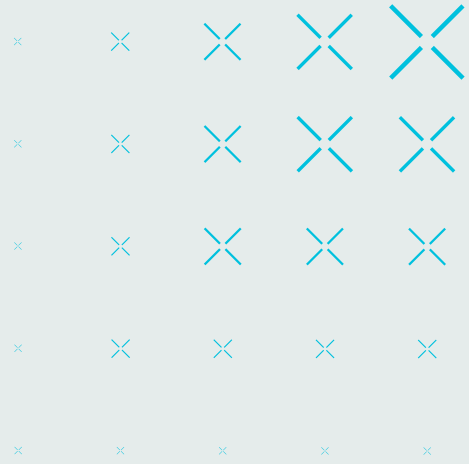


2024

TABLE OF CONTENTS

CHAPTER 01

Executive Summary



As AI continues to revolutionize industries, new workloads, like generative AI, inspire new use cases, the demand for efficient and scalable AI-based solutions has never been greater. While training often garners attention, inference—the process of applying trained models to new data—is essential for AI workloads, whether they are running in the cloud, or enabling real-world applications at the edge—on devices.

Inference covers the most widely used AI and machine learning (ML) workloads and use cases. On consumer devices, common AI workloads like object detection and facial recognition, as well as text generation and summarization, are all inference. In cars, AI inference workloads are used for autonomous and assisted-driving capabilities, while in IoT, inference is supporting the move to advanced automation. Inference is everywhere.

In general, AI is best managed through heterogeneous compute approaches that give technology companies the flexibility to use different compute components, including the CPU, GPU, and NPU, for different AI use cases and demands. However, there are many cases where the CPU is optimal for processing AI workloads, with these requiring the levels of performance, efficiency, and security that the CPU provides.

In fact, many AI inference workloads can run on the CPU, the easiest target for developers when creating their own AI-based applications. This is largely due to its ubiquity, ease of programmability, and general-purpose flexibility, as well as the latency and memory locality advantages when compared to other processors.

With this in mind, Arm has invested in ensuring that AI compute is available throughout the computing platform. And of course, Arm continues to invest in CPU designs to ensure that they are at the heart of energy-efficient AI inference at scale, as they power billions of devices globally, from consumer devices to the cloud.

The Arm architecture is unique in its flexibility and scale, enabling Arm CPUs to integrate easily with accelerator technologies in differentiated silicon solutions and seamlessly meet ever-evolving workload demands. The architecture has enabled significant impacts across various sectors:

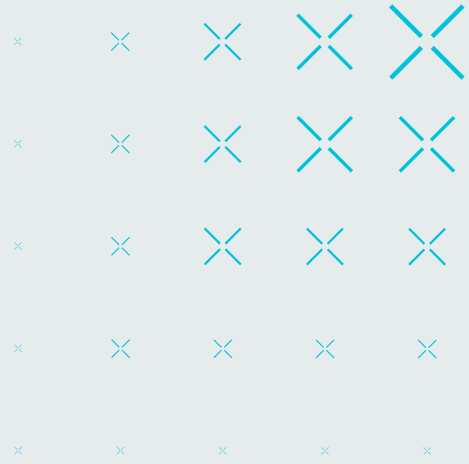
- **Datacenters:** Top hyperscalers are using the Arm Neoverse CPU platform.
- **Mobile:** Advanced AI inference workloads, including generative AI, can be processed and run on mobile devices on the Arm CPU.
- **Automotive:** Autonomous systems for autonomous driving features. and applications are built on Arm Automotive Enhanced (AE) CPUs.
- **IoT:** Advanced edge-AI workloads are running on IoT devices via the Arm CPU.

In this report, we outline why AI inference is happening on the CPU, the different technological approaches for AI inference, and examples of AI inference use cases, from the cloud to the edge, taking place on the Arm CPU.

CHAPTER 02

Introduction

The AI Revolution and the Role of Inference



The explosion of AI is changing the way we work, live, and play. However, running complex AI workloads demands substantial compute and datacenter power, which the current global power grid cannot sustain.

Driving this demand for high compute is AI training, which is the process of building models using vast datasets. AI training has always been compute intensive, but is growing exponentially with the introduction of new generative AI workloads. This is time-consuming and uses significant power and energy. For example, according to the 2024 AI Index Report from Stanford University, it costs tens of millions of dollars to train just one large language model (LLM).

However, there is another far more energy-efficient aspect to AI. It is called inference, where trained models are applied to new data to make predictions. AI inference is less resource intensive, cheaper, and faster than training, and takes place across all technology markets. Inference is where AI becomes "real" for consumers, enabling real-world applications at their fingertips. There is a misconception that AI is mainly used for training workloads, but in fact, 85 percent of all datacenter AI workloads today are AI inference, while just 15 percent are training, according to Omdia's Data Center Compute Intelligence Service.

AI inference covers the most common workloads on today's devices. Object detection, facial recognition, content recommendations and generation, language translation, text generation and summarization, audio processing, and virtual assistants on today's consumer devices are all AI inference workloads. Even autonomous systems for cars use AI inference for real-time driving decisions to improve road and passenger safety.

The Importance of CPU-Based Inference

Arm's view is that to cover all the different AI requirements for the future, AI compute capabilities will need to be distributed throughout the whole platform. Rather than AI workloads running on an island, AI will continue to utilize both CPUs and accelerators, through a heterogeneous compute approach that ensures AI workloads are matched to the appropriate computing component. There are needs and benefits to both types of compute: dedicated AI accelerators optimize the handling of an established set of algorithms that require a great deal of computation, whereas CPUs work well on more fine-grained pieces of work, or where the computational load is less. Currently, the vast majority of AI inference workloads run on the CPU, from data management and labeling to model profiling and experimentation, and it remains the easiest path for developers when targeting their own AI workloads.

As the CPU is already pervasive and at the heart of various computing platforms, in many instances, it is the common foundation in computing systems, with AI workloads starting here. This is due to its ubiquity, ease of programmability, and general-purpose flexibility, as well as the latency and memory locality advantages when compared to other processors. For many, the CPU is the practical choice for applications, particularly for those where AI-based tasks make up a smaller portion of the workload and are not evenly distributed.

Leveraging advances in processor design and AI innovation, the CPU ensures that AI inference workloads are processed as efficiently as possible in the cloud and at the edge—on the devices—to save energy and costs. Due to the power-efficient designs, the CPU is useful for AI processing on devices that are power or memory constrained. The CPU is also highly versatile and can handle a wide variety of AI-based tasks and data types, from LLMs in the cloud to running AI at the edge, whether it is on the smartphone, in the home, in the office, or in the car.

Efficient inference on CPUs plays a crucial role in AI, as it directly impacts the response time and the computational resources required. The ubiquity of CPUs means that enhanced AI experiences based on inference workloads are reaching a larger number of users than ever before.

With AI inference powering so many devices and markets, the Arm CPU is central to the AI-based experience that consumers currently enjoy every day and will continue to enjoy in the future.

CHAPTER 03

A Comprehensive Overview of AI Inference on Arm CPUs

Arm CPUs are known for their power efficiency, which is a critical requirement for AI workloads, especially inference where power consumption is a significant concern. Through a series of demos and partner case studies, Arm is showcasing how AI inference runs on the Arm CPU across mobile, cloud, automotive, and IoT markets, with these resulting in leading-edge AI experiences.

The Datacenter and the Cloud

Unparalleled Performance and Cost-Effective AI Inference

Arm is working closely with all key parts of the ML stack, including cloud service providers (CSPs) and the rapidly growing ML independent software vendor (ISV) community. Arm CPU technologies have been critical in creating powerful and energy-efficient chips for the datacenter and cloud, with all major CSPs offering Neoverse-based platforms. The unique flexibility of the Arm CPU paired with the strength of our partnerships across the technology ecosystem offers the best route to differentiated silicon solutions.

For example, NVIDIA's Grace Blackwell and Grace Hopper superchips for AI infrastructure both incorporate Arm CPUs alongside NVIDIA's AI accelerator technologies to deliver significant uplifts in AI performance.

The Grace Blackwell superchip combines NVIDIA's Blackwell GPU architecture with the Arm-based Grace CPU. This Arm-based computing solution enables system-level design optimizations that reduce energy consumption by 25 times and provide a 30 times increase in performance per GPU, compared to NVIDIA H100 GPUs using competitive architectures for LLMs. These optimizations, which deliver game-changing performance and power savings, are possible thanks to the unprecedented flexibility for silicon customization provided by Arm Neoverse CPUs.

Meanwhile, the Grace Hopper superchip is another breakthrough using the NVIDIA Grace CPU, which is based on 72-core Arm Neoverse CPUs. This combination has resulted in a 10-times performance surge for the most challenging AI tasks, while improving performance-per-watt efficiency.

In April 2024, Google Cloud introduced Google Axion Processors, based on Arm Neoverse V2, for general-purpose compute and AI inference workloads. These processors can deliver up to 60 percent increased energy efficiency and up to 50 percent more performance than comparable x86-based instances. Google services, including Google Earth Engine and YouTube Ads, are already running on Arm-based servers with additional plans to scale deployments on Axion.

Meanwhile, the recently introduced AWS Graviton4 chip offers four times the performance of the original Graviton chips and uses significantly less energy than competing CPUs. Companies like SAP, Epic Games, and SmugMug are leveraging these advancements to enhance their services.

The Arm Neoverse CSS platform has delivered unparalleled performance, efficiency, and flexibility, and enabled partners to innovate and differentiate their solutions. For instance, the Microsoft Azure Cobalt 100, the first in its series, uses the Arm Neoverse CSS to address complex challenges associated with AI and sustainability. The Cobalt 100 offers 40 percent performance improvements over the competition, powering Microsoft Teams and combining with Maia accelerators to drive the Azure end-to-end AI architecture.

The Arm v8.6-A architecture includes the SMMLA instruction, significantly accelerating these operations by multiplying and accumulating 8-bit integer matrices. Software teams can leverage this instruction to enhance the performance of int4 and int8 GEMM kernels in llama.cpp. Experiments on the AliCloud Yitian710 cloud instance, featuring 64 vCPUs and 256 GB of memory, demonstrated significant performance improvements, particularly in prompt processing and token generation.

When compared to other server CPUs on AliCloud, the Yitian710 outperforms Intel Icelake and Sapphire-Rapids CPUs, delivering up to 3.2 times increased performance in prompt processing and 2.2 times in token generation. The cost-effective benefit of the Yitian710 platform also offers up to three times higher tokens per dollar, making it attractive for developers seeking to deploy smaller, focused LLMs.

CPUs can handle a wide range of tasks and offer more flexible deployment, making them attractive for developers looking to integrate LLMs into their products and services. Advancements in our Neoverse-based server processors are reducing the barriers to entry for LLM adoption, broadening access to powerful AI capabilities.

Figure 1. Prompt Processing Comparison (Prompt = 128 tokens)

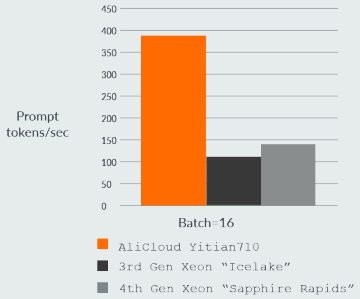
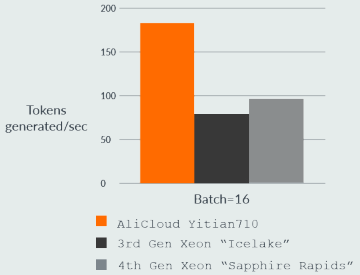


Figure 2. Token Generation Comparison (Output = 128 tokens)



To demonstrate the capabilities of Arm-based server CPUs for LLM inferencing, Arm software teams and partners optimized int4 and int8 kernels in llama.cpp to leverage newer instructions.

Experiments on AWS Graviton3, specifically on an AWS r7g.16xlarge instance with 64 vCPUs and 512 GB of memory, showed significant performance improvements. These optimizations increased prompt processing speed by up to 2.5 times and token generation throughput by up to 2 times, helping to ensure AWS Graviton3 met the 100ms latency requirement for both single and batched scenarios.

Comparisons with other server CPUs also highlight the superior performance of AWS Graviton3, delivering up to three-times increased performance for both prompt processing and token generation. Additionally, since AWS Graviton3 is more cost-effective, it can offer up to three times higher tokens per dollar, which in turn makes it ideal for LLM adoption.

It is evident that Arm CPUs provide a variety of performance, energy savings, and cost benefits for running AI workloads in the cloud, with this continuing to grow in response to evolving LLMs, high batch counts, training models and other compute-intensive tasks.

These benefits are extending to multiple AI use cases at the edge across mobile, automotive, and IoT, where running on-device AI provides better privacy, latency and scalability, alongside reduced cost and energy benefits. This is because on-device AI does not need to reply on cloud infrastructure, making it quicker to deploy workloads and easier to scale applications, while ensuring user data remains more private. On-device AI also enables offline use cases or consistent performance in situations with poor network access.

Figure 3. Optimization Uplift for Prompt Processing (Prompt = 128 tokens)

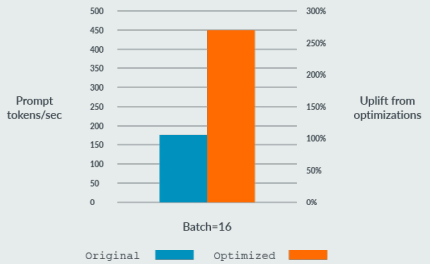
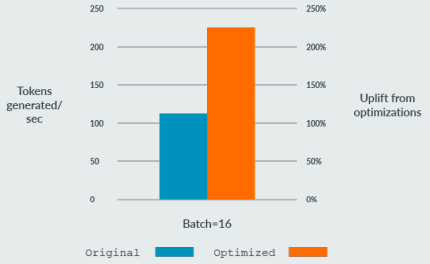


Figure 4. Optimization Uplift for Token Generation (Output = 128 tokens)



Mobile

For a long time, consumers have grown accustomed to classical AI and computer vision (CV) workloads on mobile through capabilities such as text prediction and face recognition. Advances in AI, and particularly generative AI are pushing this a step further delivering advanced AI workloads in the palm of user's hands. Today, we are seeing the progression from text prediction to text generation, and the introduction of new experiences, such as virtual assistants and chatbots, as generative AI continues to evolve and advance.

Revolutionizing Mobile AI on Arm CPUs

When the latest Llama3 model from Meta and the Phi-3 3.8B model from Microsoft came out, Arm worked quickly to run them on Arm CPUs on mobile. These new AI models are far more capable and can respond to a wider range of questions. One of our demos utilizes the Microsoft Phi-3 3.8B model on mobile through 'Ada,' a chatbot specifically trained to be a virtual teaching assistant for science and coding.

The mobile device generative AI workloads take place entirely at the edge on the Arm CPUs, with no speed up from accelerators. The impressive performance—a time-to-first token response performance and text-generation rate of just under 15 tokens per second that is faster than the average human reading speed—is enabled through a combination of existing CPU instructions for AI, alongside dedicated software optimizations for LLMs through the ubiquitous Arm compute platform. This is just one workload that Arm's teams have been exploring – and there is still more to come.

The wider market is witnessing the emergence of high-performance AI-enabled flagship smartphones. These devices leverage Arm v9 CPU technologies and include the MediaTek Dimensity 9300-powered vivo x100 and x100 Pro, Samsung Galaxy S24 series, and Google Pixel 8 smartphones. The blend of performance and efficiency that these flagship devices offer is paving the way for unprecedented opportunities in AI innovation.

Automotive

Vehicles have incorporated AI within their compute platforms for many years. Today, the use of AI in cars is growing exponentially. This is due to an ever-increasing number of driver assistance features being developed to improve advanced driver assistance systems (ADAS), which use CV and sensor fusion, as well as in-vehicle infotainment (IVI) functionalities that cover improvements in voice and speech recognition, and autonomous driving.

AI in automotive use cases covers a wide range of workloads, from classic AI workloads to modern generative AI, which can now be transferred over-the-air (OTA) in today's AI-enabled software-defined vehicles. There are already examples of these autonomous systems that utilize AI inference workloads being built on Arm Automotive Enhanced (AE) CPUs.

Pioneering Next-Gen Autonomous Driving Systems with Nuro

Arm and Nuro have partnered to develop the next-generation of self-driving vehicles using Arm AE CPU technologies, which combine AI-first software with custom-built hardware. Nuro's technology is already integrated into seven vehicle platforms for local goods delivery, with Arm collaboration creating scalable, safety-certified autonomous systems for commercial applications.

IoT

More advanced edge AI is now arriving on IoT devices, with these inference workloads running on the Arm CPU.

Demonstrating Small Language Models on Edge IoT Devices

Aizip and Renesas demonstrated ultra-efficient small language models (SLMs) working on Arm-based micro-processor units (MPUs) for a wide range of edge IoT applications, including automation. Built on the Arm Cortex-A55 CPU, the Renesas MPUs can deploy Aizip's series of ultra-efficient SLMs and AI agents, named Gizmo, that range in size from 300 million to 2 billion parameters.

Despite their compactness, these SLMs and AI agents achieve robustness and accuracy comparable with cloud-based LLMs for domain-specific applications. This is significant because ensuring accurate tool calling for automation on low-cost edge IoT applications is a challenge for SLMs.

IoT applications require higher performance and intelligence at the edge, with the combination of Aizip's SLMs and AI agents running on Arm-based technologies to fulfill this need.

Empowering Developers to Accelerate AI Everywhere on Arm

Developers are the fundamental bridge between Arm-based hardware and the AI-powered experiences used by consumers today and in the future. As Arm CPU designs are ubiquitous and readily available everywhere, from cloud to edge, it is important for these developers to have a seamless experience when building their applications on Arm, for Arm-based devices. This enables developers to achieve impressive acceleration capabilities and performance optimizations for their applications, while delivering lower development costs and a quicker time to market.

Accelerating AI Developer Innovation with Arm Kleidi

As the CPU continues to be the easiest path for developers when targeting their AI workloads, Arm launched Arm Kleidi, which is focused on accelerating AI inference on Arm Cortex and Arm Neoverse CPU architectures everywhere from cloud to edge. Arm Kleidi has three key benefits for the developer ecosystem:

Open Arm technology integrated into frameworks, seamlessly accelerating models on Arm CPUs without extra work for developers.

Developer empowerment through a wide range of resources, including usage guidance, learning paths, and demonstrations.

A vibrant ecosystem of ML software providers, frameworks and open-source projects with access to all the latest AI features.

Together, these three elements help ensure that Kleidi provides developers with the most straightforward path to target AI workloads on the Arm CPU, signifying our unwavering commitment to supporting the global developer community. The goal is a smoother, more seamless development process that optimizes the performance of AI workloads.

> The mission of Arm Kleidi is to collaborate with leading AI frameworks, CSPs, and the ML ISV community to provide out-of-the-box inference performance improvements for billions of workloads without the need for extra developer work or expertise.

We have introduced Arm Kleidi libraries for popular AI frameworks, which feature Arm KleidiAI for unleashing CPU performance across AI workloads and Arm KleidiCV for accelerating computer vision (CV) workloads. By using the Kleidi libraries, developers can accelerate AI without requiring extra effort, while enabling the faster and smoother execution of AI and CV models on Arm-based devices.

KleidiAI, our response to the surge in device types, neural networks, and inference engines, is a suite of highly optimized, lightweight AI kernels. These kernels excel in various use cases, including generative AI, by integrating seamlessly with popular AI frameworks such as MediaPipe, Llama.cpp, PyTorch, and TensorFlow Lite.

KleidiAI benchmarks show significant performance improvements for generative AI workloads, accelerating the time-to-first token for Llama 3 and Phi-3 LLMs by 190 percent. Arm teams have also managed to enable int4 quantization in Unity Sentis to reduce the model memory utilization by around 73 percent when running the Phi-2 LLM, with KleidiAI simplifying optimization. Meanwhile, with KleidiCV, Arm has identified a typical performance uplift of 75 percent in OpenCV for a variety of image processing tasks.

Arm is already working closely with leading technology partners to integrate Kleidi technologies into popular AI frameworks. Through working with Google AI Edge, Arm successfully integrated KleidiAI into the MediaPipe framework. This integration, which is accelerated on the Arm CPU via XNNPACK, an open-source library of highly optimized neural network operators, supports a wide range of LLMs, including the Gemma 2B LLM.

These efforts have led to a significant 30 percent increase in performance in time-to-first token, which relates to how many tokens are being processed per second. In this instance, KleidiAI integration enabled 250 tokens to be processed in one second for a more responsive experience, with this being observed via Arm's chatbot summarization demo on the Gemma 2B LLM on Samsung's Galaxy S24 smartphone (Exynos 2400), which is powered by Arm CPU technologies.

These promising results have far-reaching implications for AI developers when running their own AI workloads. Moreover, the outstanding performance shows what is possible for LLMs on the CPU, and how this can enable many real-world AI inference use cases, including chatbot, smart reply, and message summarization.

This work is just the beginning. In the future, Arm is already extending Kleidi beyond mobile to other markets like the cloud and automotive. This will accelerate billions of workloads on the Arm CPU and transform AI experiences with very little overhead required for application developers.

Streamline SLM and Hugging Face Model Deployment with Neoverse

As AI shifts from large language models (LLMs) to exploring smaller language models (SLMs), it is also equally important to encourage developers to create generative AI solutions with multimodal capabilities. Unlike LLMs, SLMs are more efficient, require fewer resources, and are easier to customize. Models like Llama, Mistral, Qwen, Gemma, and Phi3 are optimized for simpler tasks such as conversation, translation, summarization, and categorization, consuming significantly less energy for training.

Hugging Face pipelines simplify model deployment by abstracting away the complexity, allowing developers to use any model from their hub for inference without AI expertise or underlying code. Specifically used for sentiment analysis, an essential AI technique for interpreting emotions and opinions in text, Arm Neoverse CPUs significantly speed up the process, supporting quicker, more impactful AI insights.

By using Hugging Face Transformers with the pipeline tool, developers can efficiently run sentiment analysis models like BERT, DistilBERT, and RoBERTa on Arm-powered instances. Benchmarks on Arm-powered AWS Graviton processors demonstrated up to three-times performance improvements with AWS Graviton3 compared to the previous generation, achieving real-time latency targets even with four vCPUs. This efficiency, coupled with lower costs, makes Arm's Neoverse CPUs a compelling choice for deploying SLMs and AI applications broadly.

Delivering Open-Source Optimization from Cloud to Edge

Arm is working in open-source frameworks, such as llama.cpp and PyTorch, which are AI frameworks for inferencing open-source LLMs on CPU with highly optimized code. As part of this work with llama.cpp, Arm software teams have added specific tuned assembly routines for 4-bit quantized versions of these LLMs and then upstreamed this code.

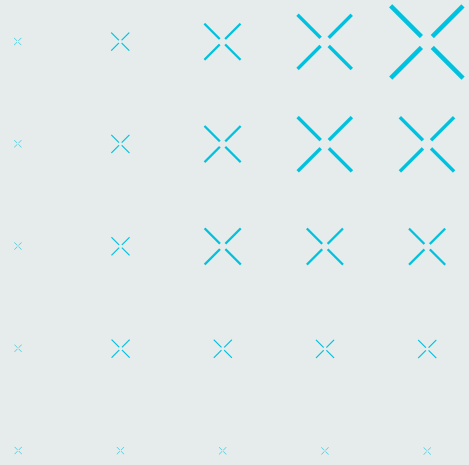
If developers download the latest version of these frameworks they will automatically benefit from these optimizations. This brings significant performance improvements, as developers can now take a range of Llama models,run them on AWS Graviton4 processors in the cloud, and achieve over 30 tokens per second throughput on the generative phase of the models.

These AI optimizations are also available beyond the cloud for developers. They can take the same software and run it on a mobile platform, such as premium smartphones. For example, running this software on the 3B parameter model enables great performance for a range of generative AI use cases.

These examples are a testament to the flexibility and programmability of the Arm CPU. Within 24 hours of the Llama 3 model being released, Arm software teams were able to deliver these optimizations in open source and generate impressive performance numbers.

CHAPTER 04

Enabling CPU-Based Inference



With a heritage grounded in pushing the boundaries of power efficiency, Arm CPU designs are powering billions of devices worldwide, from the most widely used consumer devices to the cloud, making them essential to AI inference.

Central to these CPU designs is the industry leading Arm architecture. Today's AI inference workloads leverage the instructions and features that Arm has been building into the CPU architecture for the past decade. The world's software developers are now seamlessly accessing these capabilities to accelerate and innovate their own AI-based workloads and applications. Arm's ongoing architectural innovation directly corresponds with the evolution of AI-based applications that are becoming faster, more interactive, and more immersive.

Decades of Architectural Evolution and Innovation

The AI features and instructions that are enabling AI workloads on the CPU have been evolving for over two decades. This started with the Armv7 architecture, which introduced advanced Single Instruction Multiple Data (SIMD) extensions, such as NEON technology, as Arm's initial venture into ML workloads. It has been enhanced over the past few years, with additions focused on vector dot product and matrix multiplication as part of Armv8, before the introduction of Arm Scalable Vector Extensions 2 (SVE2) and Arm Scalable Matrix Extension (SME) as key elements of Armv9.

The Arm architecture offers a flexible platform that can be closely integrated with AI accelerator technologies. This flexibility extends to the support for various vector lengths, from 120 bit to 2048 bit, allowing multiple neural networks to be executed easily across many different data points. Continuous investments in the Arm architecture have led to the development of critical features for accelerating AI computation. This continuous evolution and wide reach are a testament to the versatility and adaptability of the Arm architecture, as well as the commitment to stay at the forefront of AI developments.

Generative AI workloads are likely to change and evolve substantially over the next few years, with the ability to enable generative AI workloads efficiently at scale being one of the biggest challenges facing our industry today. This is because generative AI models scaling to billions of parameters creates enormous questions as to how these workloads can be computed efficiently.

From the perspective of the Arm CPU architecture, these generative AI workloads comprise two major elements:

- Many highly parallelizable multiplications and additions, interspersed with non-linear operations to form the network.
- The ability to feed these calculations with data, maximizing the flows from memory and optimally reusing the data to avoid wasting the memory traffic.

The flexibility and continuous evolution of the Arm CPU architecture mean it is well placed to react to these ongoing computing changes.

Security in the Age of AI

Alongside rising computing requirements, AI and the increasing integration and digitalizing of the world's data is making the "size the of the prize" greater for attackers, with attacks getting ever more sophisticated. However, even in today's age of AI, the fundamental building blocks of security remain the same, with AI requiring the same secure foundations that have evolved over the past 30 years. Arm has spent decades investing in security features as part of the Arm architecture that drive privacy-preserving compute to protect data and valuable AI models.

AI inference workloads running on the CPU benefit from Arm's security technologies and features, including those outlined below that are built into the Armv8 and Armv9 architecture sets:

- **Arm Memory Tagging Extension (MTE):** A security feature in Arm's v9 CPUs that helps detect and fix memory safety bugs, improving software security and reducing development time.
- **Realm Management Extension (RME):** A hardware feature that offers a secure execution environment, forming the basis for the Arm Confidential Compute Architecture, by enabling isolated execution of sensitive workloads.
- **Pointer Authentication (PAC) and Branch Target Identification (BTI):** Security technologies in the Arm architecture designed to mitigate memory corruption attacks by verifying addresses and restricting branches to intended targets.

Arm is committed to continuing to invest in improving security features in the architecture and making it easier to deploy across technologies and computing solutions that are built on the Arm CPU architecture.

Developer Partnerships for AI Workloads

There is no AI without software and the software developer community is actively choosing the CPU as the preferred path for targeting AI workloads. This is putting AI in the hands of billions.

By building on Arm, developers have access to tens of thousands of cloud customers who have already integrated Arm CPU technologies into their AI-optimized silicon solutions. AI developers can then leverage any trained models in the cloud on the CPU using a fraction of the energy and in one-third of the time when compared to previous systems based on legacy architectures. Developers can also accelerate AI inference performance significantly, which contributes to reduced operating costs and more efficient use of computing resources. This is crucial in the fast-evolving field of AI.

The Arm CPU provides the AI developer community with opportunities to experiment with their own techniques to deliver software optimizations that make LLMs smaller, more efficient, and faster. The flexibility and programmability of Arm CPUs enable developers to adopt AI into their applications at pace as models continue to evolve. Moreover, Arm's extensive software libraries and tools, alongside integration with all major operating systems and AI frameworks through Kleidi, help ensure developers can optimize without wasting valuable resources.

CHAPTER 05

Driving the Future of AI Inference on Arm

Arm stands at the forefront of the AI revolution, continuously pushing the boundaries of power efficiency without compromising performance. Our pervasive CPU architecture and leading platform empower AI inference capabilities everywhere, across all technological interfaces and performance levels. These range from large-scale datacenters employing numerous processors for LLMs, to AI at the edge across consumer and IoT devices and automotive applications. Moreover, due to the flexibility of the CPU, it is the practical choice for processing AI workloads where the computational load is less, and where low cost and high levels of power efficiency are priorities.

While the Arm CPU continues to offer a home for more efficient AI inference, Arm helps ensure that AI compute can scale across the whole computing platform. Arm's architecture innovations mean technology partners can be seamlessly augmented and integrated with AI accelerator technologies, such as GPUs and NPUs, as part of a flexible heterogeneous computing approach to AI workloads.

Arm remains committed to the continuous evolution of our processor technologies, offering increased performance and efficiency over time. This has led to a doubling of AI processing capabilities every two years over the past decade.

Moreover, Arm's dedication to continuous architectural innovation is unwavering. The latest Armv9 architecture features SVE2 and SME, enabling our ecosystem to achieve superior performance with reduced power consumption across their AI-based solutions and applications. The Arm architecture also builds in comprehensive security features and technologies that are needed as attacks grow more sophisticated in the age of AI.

This relentless focus on technology innovation underscores our commitment to driving the future of AI inference on Arm CPUs. From cloud to edge, the Arm CPU is the computing foundation for AI inference, and a vital component in the growing adoption of AI.

CHAPTER 06

Appendices

Glossary of Terms

Arm Architecture
A family of processor and system architectures developed by Arm. It is known for enabling products to be built in an efficient, affordable, and secure way, with the flexibility to enable unique levels of customization.

AI Inference
The process of applying a trained AI model to new data to make predictions or decisions. Unlike training, which is computationally intensive, inference is less resource-demanding and more efficient.

Arm Neoverse
A suite of products and technologies developed by Arm designed for high-performance computing and AI workloads, particularly in datacenters and cloud environments.

AI Training
The process of building an AI model by feeding it large amounts of data and allowing it to learn patterns and features. This phase is computationally intensive and requires significant resources.

Central Processing Unit (CPU)
The primary component of a computer that performs most of the processing inside a computer. It is responsible for executing instructions from programs and managing system hardware components.

Heterogeneous Compute
An approach that uses a mix of different types of processors (e.g., CPUs, GPUs, NPUs) to optimize the performance of various computing tasks, including AI workloads.

Inference Engine
Software that executes AI inference on a trained model, typically optimized for performance and efficiency on specific hardware.

Arm Kleidi
Arm technologies and developer enablement resources for unlocking AI performance on Arm CPUs to advance AI model capability, accuracy, and speed.

Neural Processing Unit (NPU)
A specialized hardware component designed to accelerate the computation of ML algorithms specifically neural networks.

Single Instruction Multiple Data (SIMD)
A parallel computing architecture that allows a single instruction to be executed on multiple data points simultaneously. This is useful for applications like multimedia processing and AI.

Small Language Model (SLM)
A more efficient and less resource-intensive version of a language model, designed to perform specific tasks, such as conversation, translation, and summarization.

Scalable Matrix Extension (SME)
An instruction set architecture (ISA) introduced in the Armv9-A to accelerate AI and ML workloads and improve performance, power efficiency, and flexibility for AI and ML-based applications running on the Arm CPU.

Scalable Vector Extension 2 (SVE2)
An extension of the Arm architecture that enhances vector processing capabilities, allowing for more efficient execution of ML workloads.

Additional Resources and References

- arm.com/architecture
- arm.com/architecture/security-features/arm-confidential-compute-architecture
- arm.com/products/silicon-ip-cpu/neoverse
- arm.com/technologies/neon
- cloud.google.com/blog/products/compute/introducing-googles-new-arm-based-cpu
- community.arm.com/arm-community-blogs/b/ai-and-ml-blog/posts/generative-ai-on-mobile-on-arm-cpu
- community.arm.com/arm-community-blogs/b/ai-and-ml-blog/posts/kleidiai
- community.arm.com/arm-community-blogs/b/ai-and-ml-blog/posts/kleidicv
- community.arm.com/arm-community-blogs/b/infrastructure-solutions-blog/posts/accelerated-llm-inference-on-arm-neoverse-n2
- community.arm.com/arm-community-blogs/b/infrastructure-solutions-blog/posts/accelerating-sentiment-analysis-on-arm-neoverse-cpus
- community.arm.com/arm-community-blogs/b/infrastructure-solutions-blog/posts/best-in-class-llm-performance
- developer.arm.com/documentation/102340/0100/Introducing-SVE2
- newsroom.arm.com/blog/arm-kleidi
- newsroom.arm.com/blog/aws-graviton4
- newsroom.arm.com/blog/generative-ai-on-mobile
- newsroom.arm.com/blog/kleidiai-integration-mediapipe
- newsroom.arm.com/blog/memory-safety-arm-memory-tagging-extension
- newsroom.arm.com/blog/neoverse-nvidia-grace
- newsroom.arm.com/blog/pac-bti
- newsroom.arm.com/blog/scalable-matrix-extension
- newsroom.arm.com/news/arm-neoverse-subsystems
- newsroom.arm.com/news/arm-nuro-autonomous-partnership
- newsroom.arm.com/news/google-cloud-custom-silicon-on-arm
- newsroom.arm.com/news/microsoft-custom-silicon-on-arm
- newsroom.arm.com/news/new-automotive-technologies-2024
- newsroom.arm.com/news/tcs23-mediatek-vivo