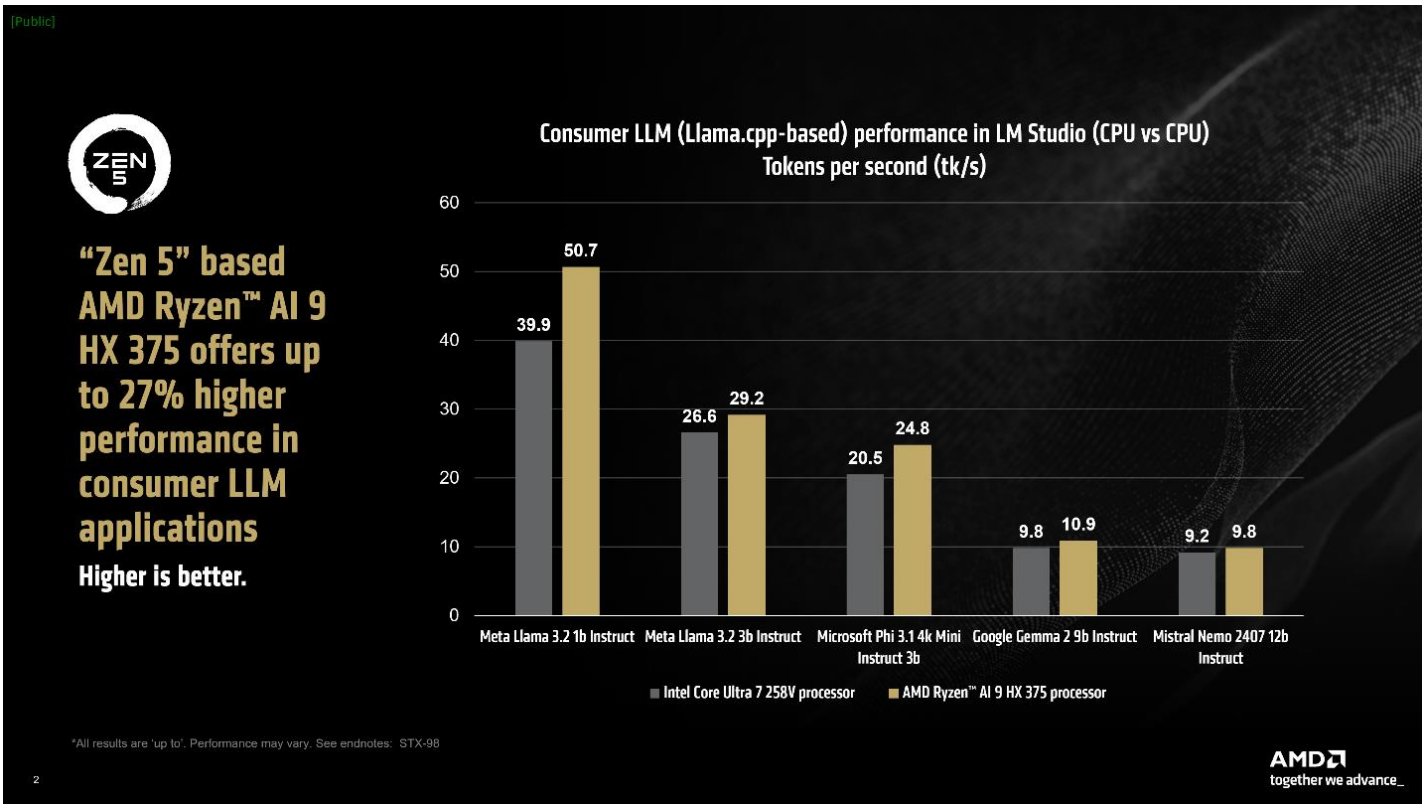
Language models have come a long way since GPT-2 and users can now quickly and easily deploy highly sophisticated LLMs with consumer-friendly applications such as LM Studio. Together with AMD, tools like these make AI accessible for everyone with no coding or technical knowledge required.

Overview of llama.cpp and LM Studio

LM Studio is based on the llama.cpp project;- which is a very popular framework to rapidly deploy language models. It has no dependencies and can be accelerated using only the CPU – although it has GPU acceleration available. LM Studio uses AVX2 instructions to accelerate modern LLMs for x86-based CPUs.

Performance comparisons: throughput and latency



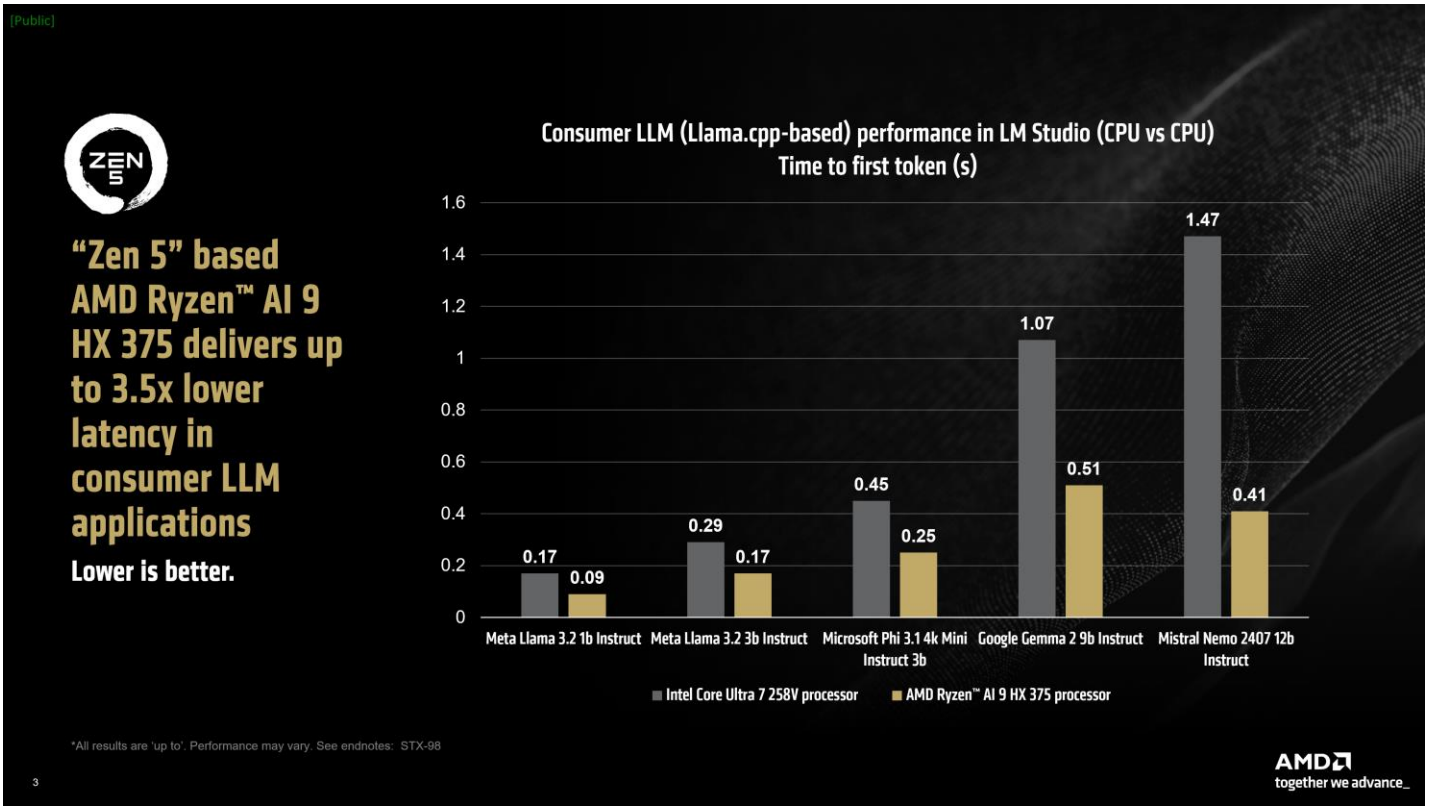
AMD Ryzen™ AI accelerates these state-of-the-art workloads and offers leadership performance in llama.cpp based applications like LM Studio for x86 laptops[1]. It is worth noting that LLMs in general are very sensitive to memory speeds.

In our comparison, the Intel laptop actually had faster RAM at 8533 MT/s while the AMD laptop has 7500 MT/s RAM. In spite of this, the AMD Ryzen™ AI 9 HX 375 processor achieves up to 27% faster performance than its competition when looking at tokens per second. For reference, tokens per second or tk/s is the metric which

denotes how quickly an LLM is able to output tokens (which roughly corresponds to the number of words printed on-screen per second).

The AMD Ryzen™ AI 9 HX 375 processor can achieve up to 50.7 tokens per second in Meta Llama 3.2 1b Instruct (4-bit quantization).

Another metric for benchmarking large language models is “time to first token” which measures the latency between the moment you submit a prompt and the time it takes for the model to start generating tokens. Here we see that in larger models, the AMD “Zen 5” based Ryzen™ AI HX 375 processor is up to 3.5x faster than a comparable competitor processor[1].

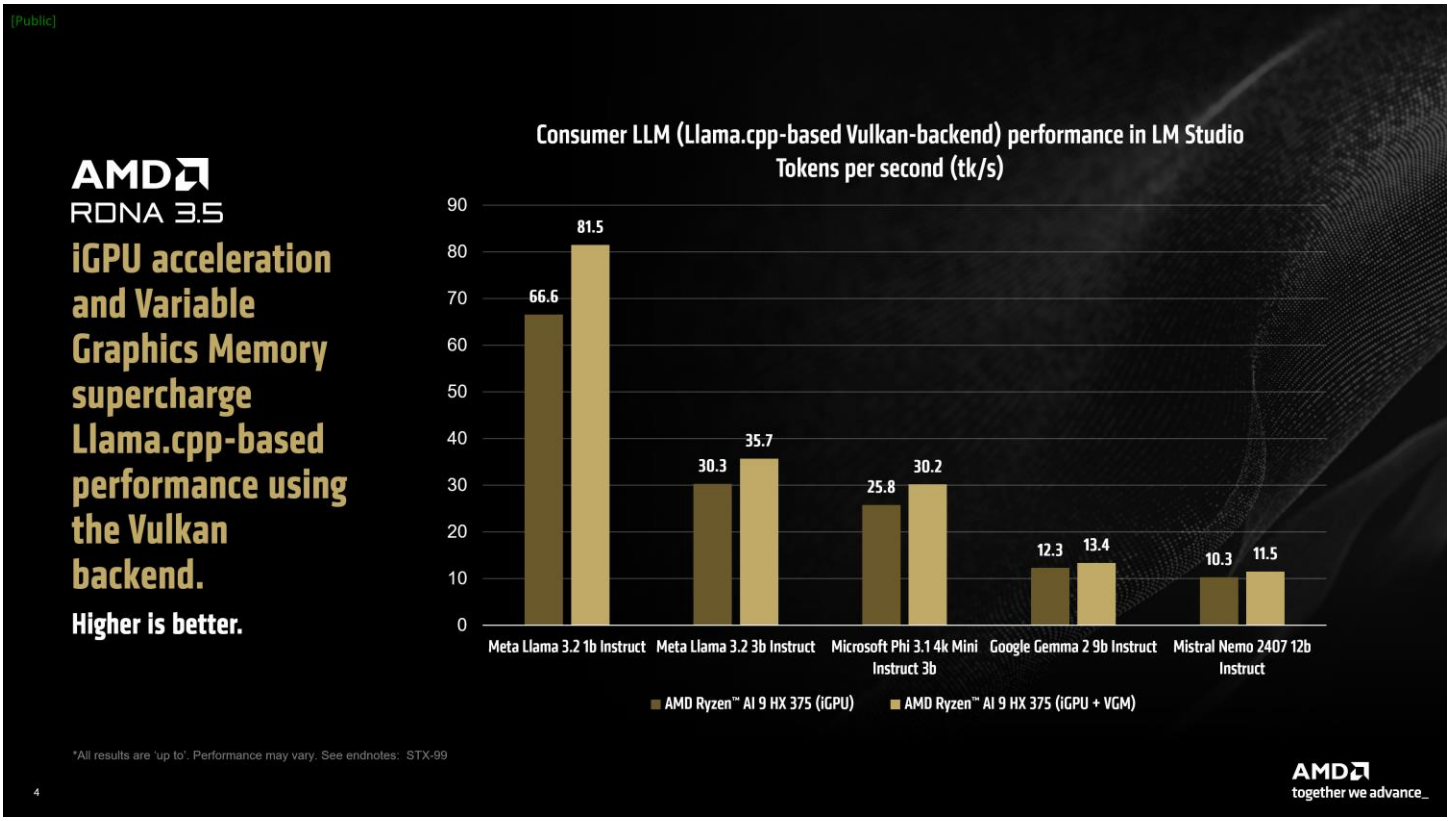


Using Variable Graphics Memory (VGM) to speed up model throughput in Windows

Each of the three accelerators present in an AMD Ryzen™ AI CPU have their own workload specialization and scenarios where they excel. Where AMD XDNA™ 2 architecture-based NPUs provide incredible power efficiency for persistent AI while running Copilot+ workloads, and CPUs provide broad coverage and compatibility for tools and frameworks – it is the iGPU which often handles on-demand AI tasks.

LM Studio features a port of llama.cpp which can accelerate the framework using the vendor-agnostic Vulkan API. Acceleration here is usually dependent on a mix of hardware capabilities and driver optimizations for the Vulkan API. Turning on GPU offload in LM Studio resulted in a 31% average performance increase in Meta Llama 3.2 1b Instruct performance compared to CPU-only mode. Larger models like Mistral Nemo 2407 12b Instruct which are bandwidth bound in the token generation phase saw an uplift of 5.1% on average.

We observed that when using the Vulkan-based version of llama.cpp in LM Studio and turning on GPU-offload, the competition’s processor saw ***significantly lower*** average performance in all but one of the models tested when compared to their CPU-only mode. Because of this reason and in effort to keep the comparison fair, we have not included the GPU-offload performance of the Intel Core Ultra 7 258v in LM Studio’s Llama.cpp based Vulkan back-end.

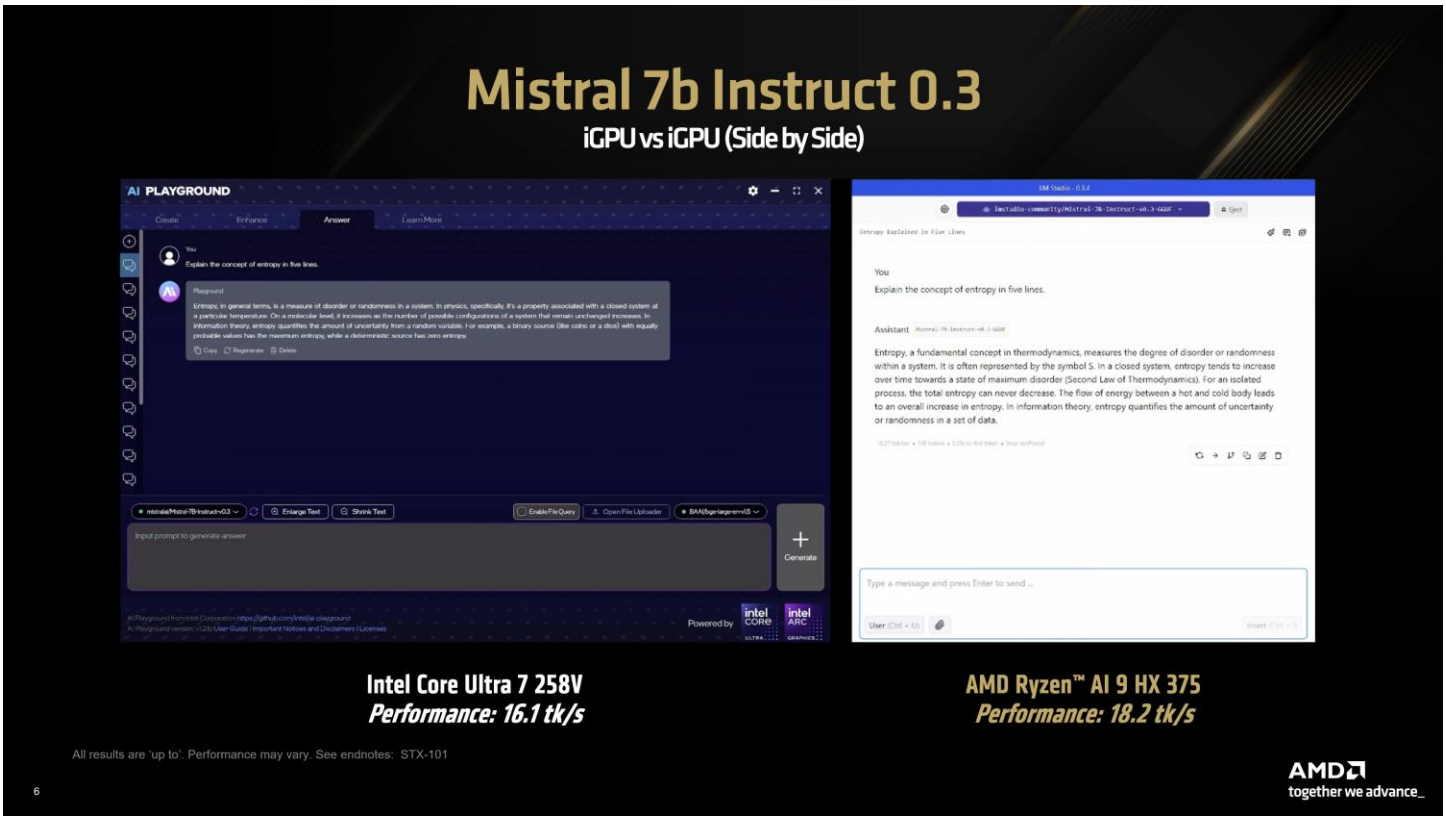


AMD Ryzen™ AI 300 Series processors also include a feature called Variable Graphics Memory (VGM). Typically, programs will utilize the 512 MB block of dedicated allocation for an iGPU plus the second block of memory that is housed in the “shared” portion of system RAM. VGM allows the user to extend the 512 “dedicated” allocation to up-to-75% of available system RAM. The presence of this contiguous memory significantly increases throughput in memory-sensitive applications.

After turning on VGM (16GB), we saw a further 22% average uplift in performance in Meta Llama 3.2 1b Instruct for a net total of 60% average faster speeds, compared to the CPU, using iGPU acceleration when combined with VGM. Even larger models like Mistral Nemo 2407 12b Instruct saw a performance uplift of up to 17% when compared to CPU-only mode.

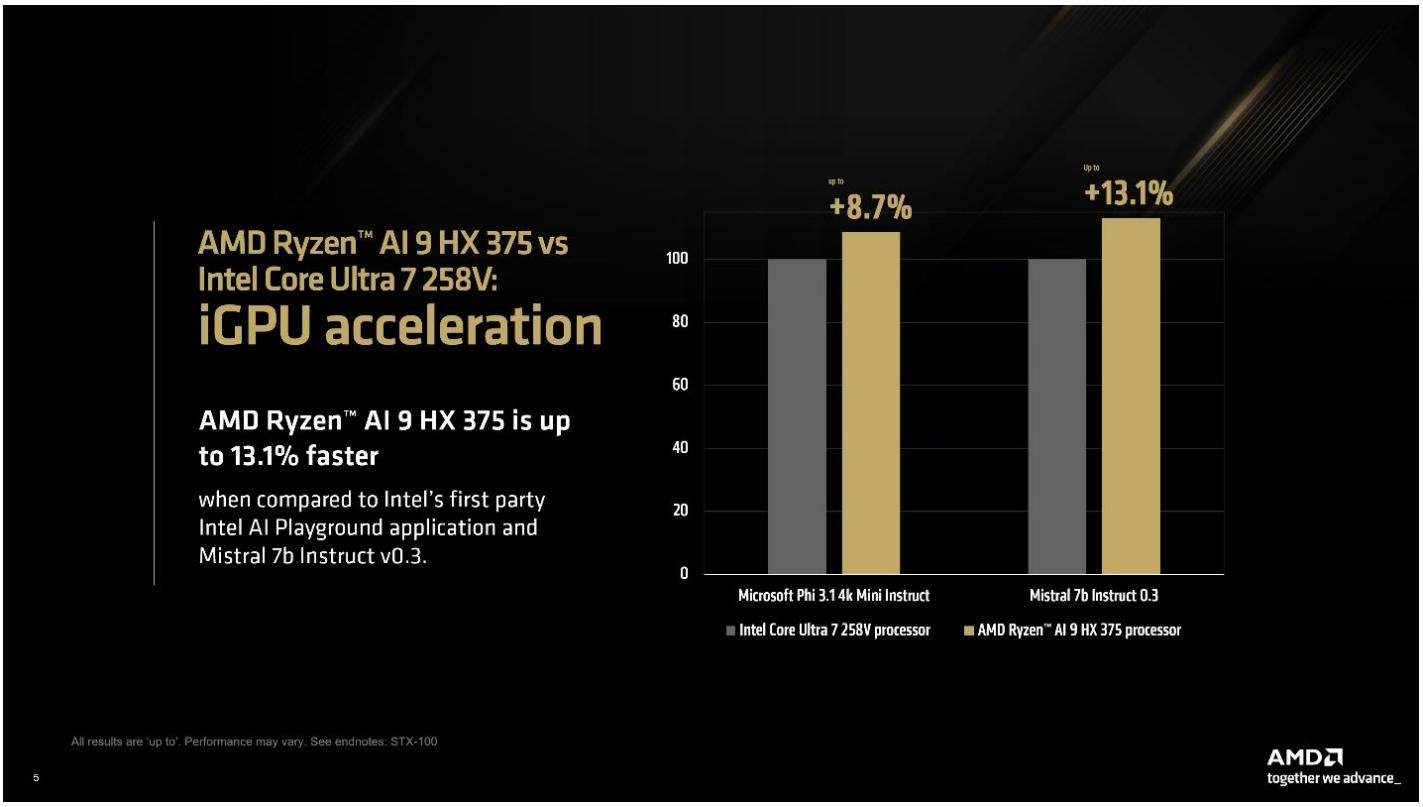
Side by side comparison: Mistral 7b Instruct 0.3

While the competition’s laptop did not offer a speedup using the Vulkan-based version of Llama.cpp in LM Studio, we compared iGPU performance using the first-party Intel AI Playground application (which is based on IPEX-LLM and LangChain) – with the aim to make a fair comparison between the best available consumer-friendly LLM experience.



Video demo.

We used the models provided with Intel AI Playground – which are Mistral 7b Instruct v0.3 and Microsoft Phi 3.1 Mini Instruct. Using a comparable quantization in LM Studio, we saw that that the AMD Ryzen™ AI 9 HX 375 is 8.7% faster in Phi 3.1 and 13% faster in Mistral 7b Instruct 0.3.



AMD believes in advancing the AI frontier and making AI accessible for everyone. This cannot happen if the latest AI advances are gated behind a very high barrier of technical or coding skill – which is why applications like LM Studio are so important. Apart from being a quick and painless way to deploy LLMs locally, these applications allow users to experience state-of-the-art models pretty much as soon as they launch (assuming the llama.cpp project supports the architecture).

AMD Ryzen™ AI accelerators offer incredible performance and turning on features like Variable Graphics Memory can offer even better performance for AI use cases. All of this combines to deliver an incredible user experience for language models on an x86 laptop.

You can try out LM Studio yourself here.

Endnotes:

1. For this comparison we selected best-available 14-inch laptops available in the North American market at the time of procurement.
2. STX-98: Testing as of Oct 2024 by AMD. Average performance of three runs for specimen prompt "Explain the concept of entropy in five lines". All tests conducted on LM Studio 0.3.4. Models tested: Meta Llama 3.2 1b Instruct, Meta Llama 3.2 3b Instruct, Microsoft Phi 3.1 4k Mini Instruct, Google Gemma 2 9b Instruct, Mistral Nemo 2407 13b Instruct. (All models are Q4 K M quantization). Intel specific configuration: CPU threads = 8. AMD specific configuration, threads = 12. (Llama.cpp recommends setting threads equal to the number of physical cores). HP OmniBook Ultra Laptop 14-inch with AMD Ryzen AI 9 HX 375. 32GB 7500 MT/s RAM. VBS = ON. Windows 11 Pro 24H2. ASUS Zenbook S14 UX5406SA 14-inch with Intel Core Ultra 7 258V. 32GB 8533 MT/s RAM. VBS = ON. Windows 11 Pro 24H2. Performance may vary. STX-98.
3. STX-99: Testing as of Oct 2024 by AMD. Average performance of three runs for specimen prompt "Explain the concept of entropy in five lines". All tests conducted on LM Studio 0.3.4. Models tested: Meta Llama 3.2 1b Instruct, Meta Llama 3.2 3b Instruct, Microsoft Phi 3.1 4k Mini Instruct, Google Gemma 2 9b Instruct, Mistral Nemo 2407 13b Instruct. (All models are Q4 K M quantization). CPU threads = 12. (Llama.cpp recommends setting threads equal to the number of physical cores). GPU offload = MAX. VGM set to 16GB during VGM runs. HP OmniBook Ultra Laptop 14-inch with AMD Ryzen AI 9 HX 375. 32GB 7500 MT/s RAM. VBS = ON. Windows 11 Pro 24H2. Performance may vary. STX-99.
4. STX-100: Testing as of Oct 2024 by AMD. Average performance of three runs for specimen prompt: ”How long would it take for ball dropped from 10 meter height to hit the ground?”, "Explain the concept of entropy in five lines". All tests conducted on LM Studio 0.3.4 for AMD laptops. All tests conducted using Intel AI Playground 1.21b for Intel laptops. Models tested: Mistral 7b Instruct v0.3 Q4 K M, Mistral 7b Instruct v0.3 sym_int4, Microsoft Phi 3.1 4k Mini Instruct Q4 K M, Microsoft Phi 3.1 4k Mini Instruct sym_int4. AMD specific configuration, threads = 12. (Llama.cpp recommends setting threads equal to the number of physical cores). GPU offload = MAX. VGM set to 16GB during VGM runs. HP OmniBook Ultra Laptop 14-inch with AMD Ryzen AI 9 HX 375. 32GB 7500 MT/s RAM. VBS = ON. Windows 11 Pro 24H2. ASUS Zenbook S14 UX5406SA 14-inch with Intel Core Ultra 7 258V. 32GB 8533 MT/s RAM. VBS = ON. Windows 11 Pro 24H2. Performance may vary. STX-100.
5. STX-101: Testing as of Oct 2024 by AMD. Average performance of three runs for specimen prompt: "Explain the concept of entropy in five lines". All tests conducted on LM Studio 0.3.4 for AMD laptops. All tests conducted using Intel AI Playground 1.21b for Intel laptops. Models tested: Mistral 7b Instruct v0.3 Q4 K M, Mistral 7b Instruct v0.3 sym_int4. AMD specific configuration, threads = 12. (Llama.cpp recommends setting threads equal to the number of physical cores). GPU offload = MAX. VGM set to 16GB during VGM runs. HP OmniBook Ultra Laptop 14-inch with AMD Ryzen AI 9 HX 375. 32GB 7500 MT/s RAM. VBS = ON. Windows 11 Pro 24H2. ASUS Zenbook S14 UX5406SA 14-inch with Intel Core Ultra 7 258V. 32GB 8533 MT/s RAM. VBS = ON. Windows 11 Pro 24H2. Performance may vary. STX-101.
6. GD-220c: Ryzen™ AI is defined as the combination of a dedicated AI engine, AMD Radeon™ graphics engine, and Ryzen processor cores that enable AI capabilities. OEM and ISV enablement is required,

and certain AI features may not yet be optimized for Ryzen AI processors. Ryzen AI is compatible with: (a) AMD Ryzen 7040 and 8040 Series processors except Ryzen 5 7540U, Ryzen 5 8540U, Ryzen 3 7440U, and Ryzen 3 8440U processors; (b) AMD Ryzen AI 300 Series processors, and (c) all AMD Ryzen 8000G Series desktop processors except the Ryzen 5 8500G/GE and Ryzen 3 8300G/GE. Please check with your system manufacturer for feature availability prior to purchase. GD-220c.