THE CEILING OF EDGE COMPUTING

UNBOXING AND TESTING ADVANTECH MIC-743-AT

Product: MIC-743-AT - NVIDIA® Jetson ThorTM

Date: 17.09.2025

Author: You Junzhe

When I opened the box and saw a heavy industrial computer, the first thing I saw was the front of the MIC-743-AT-ES.

【Overall】The appearance continues the consistent calm design of industrial computers, with a matte black metal case and Advantech logo, which is simple and professional. The upper cover is also covered with a protective film to prevent scratches during transportation and maintain a brand new texture when unboxing.

On the front is the low-key Advantech logo, with no other devices, as expected from international manufacturers.

The opening at the bottom shows the NVMe slot, which is also a product of Advantech, this time the test is provided with a capacity of 1T SSD, the right side is the M2 interface, which can be connected to an external WIFI transmission module to improve the transmission capacity.

There is also a SIM card slot and a series of control buttons, including REC, OTG, RST, and Console interfaces, allowing users to quickly debug and verify functions. This is not a mobile phone, of course, it can provide 4/5G transmission function, and you can also access the network or remote control in areas without WiFi.

Moving to **the back of the fuselage,** the I/O interfaces of this host are quite complete:

- Dual antenna holes to meet wireless communication needs
- Audio interface (Line out / Mic) for easy connection to external audio devices
- 4 USB 3.0 ports to support peripheral expansion
- HDMI output for direct connection to the monitor
- The 5G Base-T LAN port and QSFP28 high-speed module slot demonstrate data center-grade transmission capabilities
- DC-IN (19~36V) industrial power input, suitable for various edge environment applications

Specifications

Process		NVIDIA® Jetson T5000™
	CPU	14 core Arm® Neoverse V3AE (64-bit) SMP CPU architecture L1 Cache (I, D) per core: 64KB + 64KB, L2 1MB, L3 16MB
	GPU	2560 NVIDIA® CUDA® cores 96 5th Gen Tensor cores MAXN: 1.57 GHz
	Memory	128GB LPDDR5X
Ethernet	RJ45	1 x 10/100/1000/2500/5000 Mbps Ethernet
	QSFP28	4 x 25GbE
I/O	Display	HDMI (Max. resolution 3840x2160 @ 60Hz)
	USB	4 x USB 3.2 Gen 2
	OTG USB	1 x Micro USB
	Console	1 x microUSB console for debug (UART to USB)
	Audio	1 x audio speaker out, MIC in on board
	SIM card	1 x Nano SIM on side of board
Expansion	M.2 Ekey	1 x 2230 E-key (Signal: PCIex1 + USB2.0)
	M.2 Bkey	1 x 3052/3042 B-key (Signal: USB3.0 + USB2.0)
	TPM	1 x TPM (onboard)
Storage	M.2 M.key	1 x 2280 M-key NVMe (PCIex4); 1TB SSD by defualt
Power	Power Supply	Power input: 19-36V
	Power Type	Terminal Block 2 Pin
	Internal pin header	3 x SMART FAN
Environment	Operational Temperature	-10 ~ 35 °C with 0.7 m/s air flow
	Operating Humidity	95% @ 40 °C (non-condensing)
	Vibration	3 Grms @ 5 ~ 500 Hz, random, 1 hr/axis
Mechanical	Dimensions (W x D x H)	195 x 200 x 71.5 mm with rubber foot
	Weight	2kg
	Installation	Desktop
Operating System		NVIDIA JetPack 7.0
Certifications		CB/UL/CE/FCC/BSMI/CCC (No RED certification)

After completing the body appreciation, we plugged in and started to test its performance, Advantech MIC-743-AT-ES, equipped with the latest NVIDIA® Jetson Thor™ module, we are curious: Can an edge AI system really "bear" a model with tens of billions of parameters on the local side?

Before testing, take a look at the NVIDIA Jetson AGX Thor (Dev Kit) specifications. Although we are testing the **Advantech MIC-743-AT-ES** this time, using **the NVIDIA Jetson Thor module: the Advantech system built by the NVIDIA Jetson T5000**, not the NVIDIA Jetson AGX Thor Dev Kit, it belongs to the Thor generation, so we use the official NVIDIA Jetson AGX Thor data as a reference. It can help everyone understand the performance benchmarks of this generation.

Looking at his specification data on the Internet, I found that it was really Thor level.

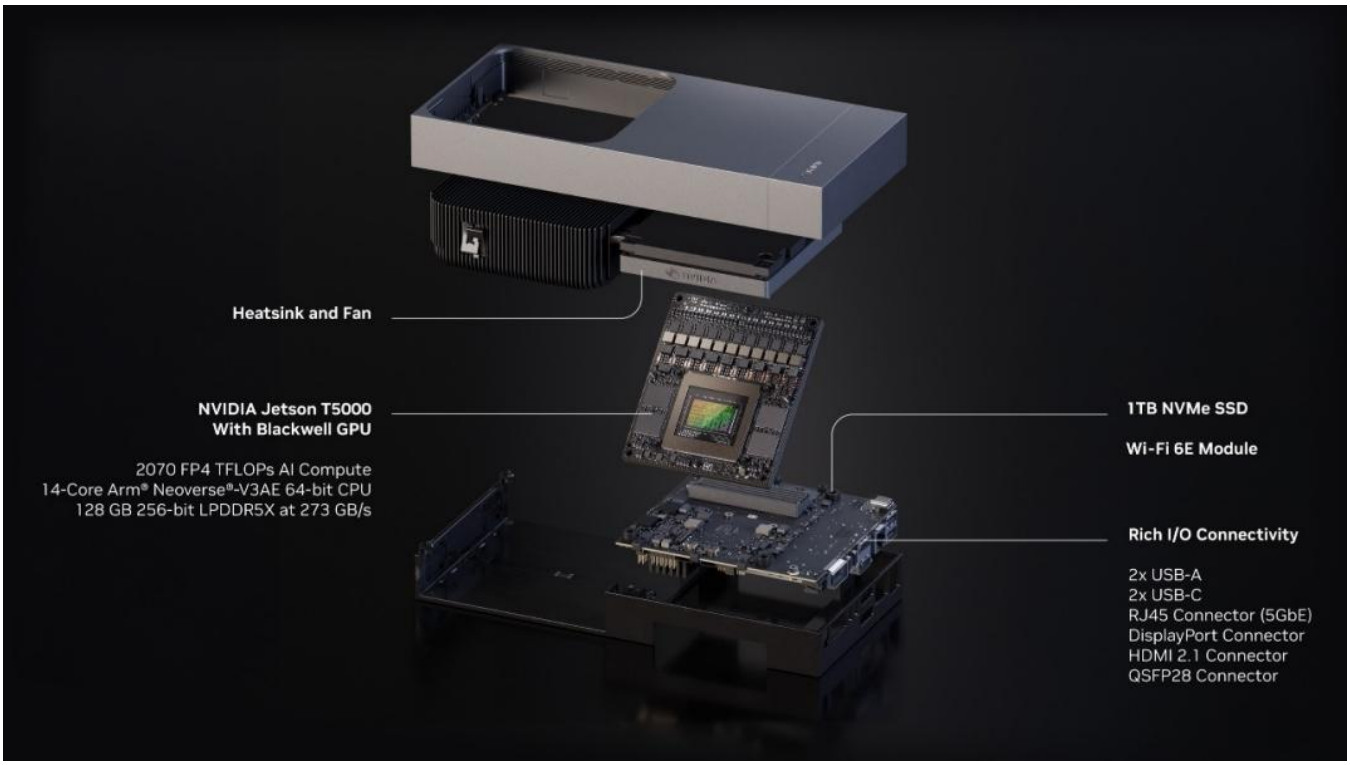


Image credit: NVIDIA

NVIDIA Jetson AGX Thor™ Development Kit Specifications

- **GPU:** 2560-core NVIDIA Blackwell-based GPU with 96 5th Gen Tensor cores supporting 10 TPCs Multi-Instance GPU (MIG).
- **CPU:** 14-core Arm® Neoverse-V3AE® 64-bit processor with 1 MB L2 cache per core and shared 16 MB system-level L3 cache
- **Vision Accelerator:** 1× PVA v3
- **Memory:** 128 GB LPDDR5X,256-bit bus, up to 273 GB/s bandwidth
- **Storage:** 1 TB NVMe solid-state drive, M.2 Key M interface
- **Power:** 40 W to 130 W

The **14-core Arm Neoverse-V3AE CPU** is paired with 128 GB **LPDDR5X** and 273 GB/s bandwidth to ensure that large amounts of visual and sensor data enter and exit in real time; Coupled with the **PVA v3 visual accelerator**, it can offload preprocessing/optical flow, etc. 1 TB **NVMe** enables smooth data caching and model switching; The 40–130 W power consumption range is ideal for edge devices to maintain high throughput inference and low latency control under limited power.

The difference from the previous generation Orin can be said to be very large

項目	Jetson Orin Nano Super 8GB	Jetson AGX Thor (Dev Kit)
GPU 架構	Ampere；1024 CUDA / 32 Tensor	Blackwell；2560 CUDA / 96 第五代 Tensor，支援 MIG（10 TPC）
標稱 AI 性能	**67 TOPS**（INT8；Super 模式，稀疏）/ **33 TOPS**（INT8；密集）/ **~17 TFLOPs**（FP16）	**2070 TFLOPS**（FP4；Sparse）
記憶體與頻寬	8 GB LPDDR5，128-bit，**102 GB/s**	128 GB LPDDR5X，256-bit，**273 GB/s**
CPU	6 核 Arm Cortex-A78AE	14 核 Arm Neoverse-V3AE
功耗範圍	7–25 W	40–130 W

Spec comparison NVIDIA Jetson Thor delivers data center-class edge computing power (2070 FP4 TFLOPS) with Blackwell architecture and MIG for large-scale multi-model and generative workloads. But what exactly is it really like? This time, we directly threw GPT-OSS 120B onto the MIC-743-AT-ES, an AI inference system equipped with the NVIDIA Jetson Thor module, to test its real-world performance.

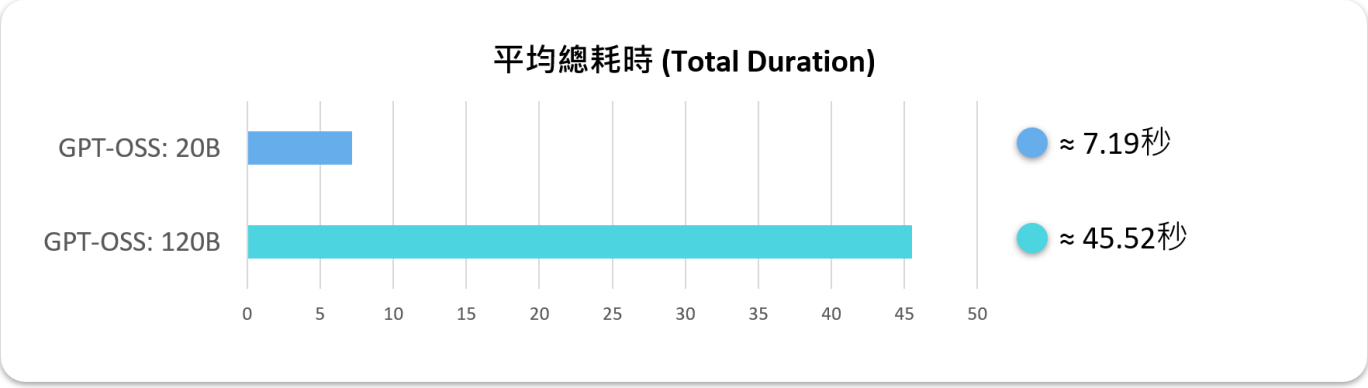
1. Measured model performance: GPT-OSS 20B vs GPT-OSS 120B

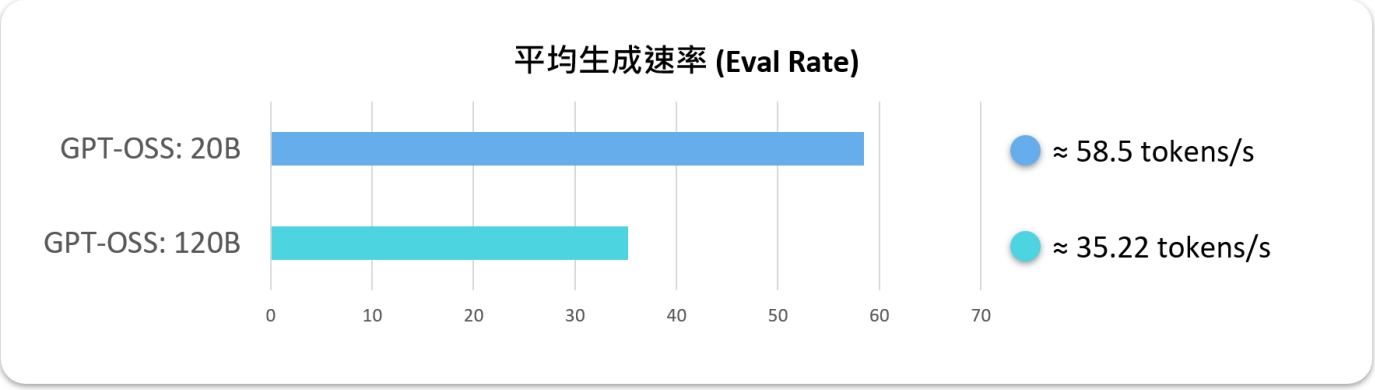
I loaded two GPT-OSS models (20B and 120B) and tested their performance with Ollama local inference:

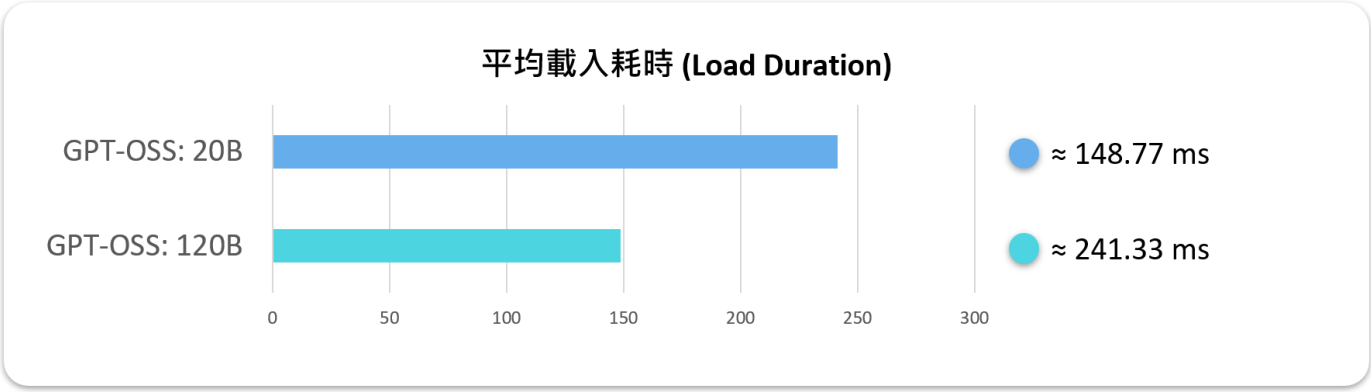
【Test Environment】

- **Hardware platform:**Advantech MIC-743-AT-ES
- **AI Module:**NVIDIA Jetson T5000
- **Memory:** 128GB LPDDR5X
- **OS / SDK**：Ubuntu + JetPack 7.0
- **Testing Tools:**Ollama, local reasoning
- **Model:**GPT-OSS:20B vs. GPT-OSS:120B

【Measured Data】



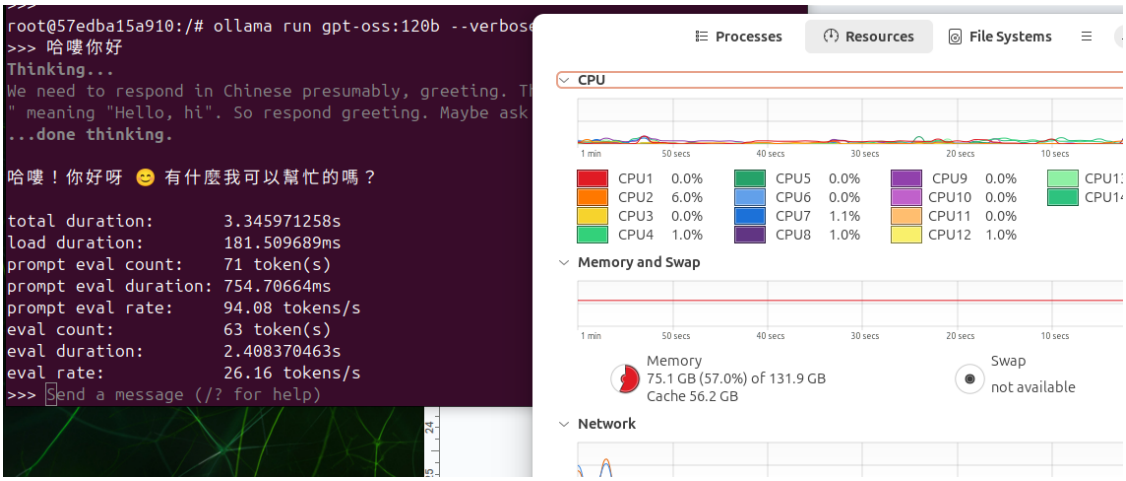




When actually testing [**GPT-OSS 20B**], the overall experience can be said to be quite smooth.

The average total time is about 7 seconds, and the loading time is less than 150 milliseconds, which is almost negligible. The prompt partially processed nearly 500 tokens, but the evaluation took only 0.04 seconds, which is equivalent to swallowing almost 40,000 tokens in one second, which is surprisingly efficient. The generation stage outputs an average of 410 tokens, which takes about 7 seconds, and the generation speed stabilizes at 58.5 tokens/s, which is already very ideal for interactive applications.

In contrast, **GPT-OSS 120B's** performance is a bit slower, but it is still impressive. The average total time is about 45 seconds, of which the loading time is about 0.24 seconds, which is slightly longer than 20B, but in the scale of tens of billions of models, such a starting speed is actually quite fast. The prompt part processes more than 1,600 tokens at a time, and it takes 1.65 seconds to evaluate alone, so the efficiency is naturally not comparable to that of 20B. However, the generation process outputs more than 1,500 tokens, and the average speed remains at 35 tokens/s.



In simple terms, **20B is suitable for real-time interaction, with fast speed and low** latency; **120B is for scenarios that require higher language comprehension and deeper reasoning.** The performance of the two is just the opposite of "speed vs. intelligence density", depending on which one is prioritized in different applications.

Detailed data are below

項目	GPT-OSS:20B	GPT-OSS:120B
平均總耗時 (Total Duration)	≈ 7.19 秒	≈ 45.52 秒
平均載入耗時 (Load Duration)	≈ 148.77 ms	≈ 241.33 ms
平均 Prompt Token 數 (Prompt Eval Count)	≈ 489 tokens	≈ 1604 tokens
平均 Prompt 評估耗時 (Prompt Eval Duration)	≈ 42.98 ms	≈ 1649.69 ms (≈1.65 秒)
平均 Prompt 評估速率 (Prompt Eval Rate)	≈ 37,756 tokens/s	≈ 689.76 tokens/s
平均生成 Token 數 (Eval Count)	≈ 410 tokens	≈ 1519 tokens
平均生成耗時 (Eval Duration)	≈ 6,990.80 ms (≈6.99秒)	≈ 43,509.22 ms (≈43.5秒)
平均生成速率 (Eval Rate)	≈ 58.5 tokens/s	≈ 35.22 tokens/s

2. GPT-OSS 120B model operation results

When I typed in "Hello, hello," the model not only correctly understood the context, but also thoughtfully added, "Is there anything I can do to help? 😊」。

This natural interaction gives people a sense of "really running a super large model on the local end".

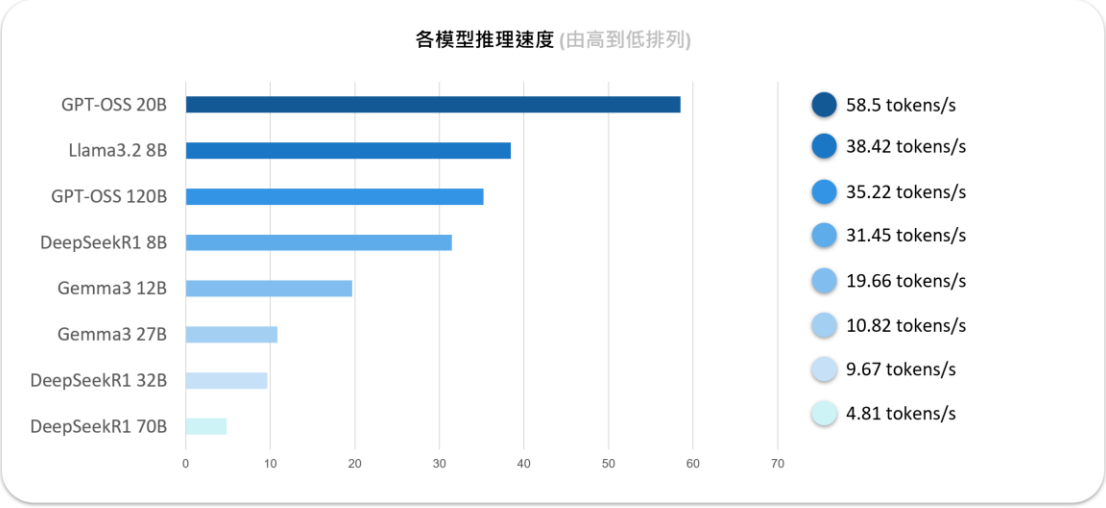
Although the current performance is not the best, being able to evoke such a language model offline and at any time is a milestone in itself.

Interestingly, the CPU is almost not fully eaten up during the model response process, indicating that the computing core may mainly fall on the GPU, and the CPU is only responsible for peripheral scheduling. On the other hand, the memory eats up 75GB, which is exactly what is expected with the huge amount of parameters like GPT-120B.

This also means that if you want to play the 120B model locally, you don't have more than 128GB of memory, so you don't have to think about it.

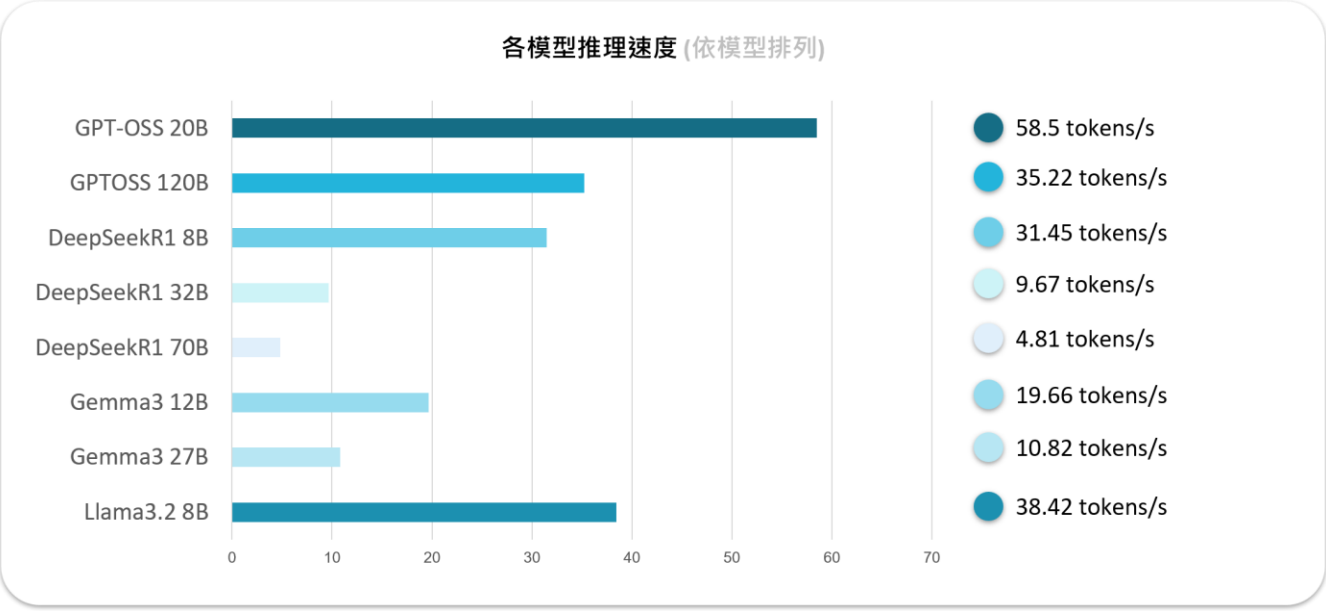
3. LLM performance control experiment

In order to more comprehensively observe the performance of NVIDIA Jetson Thor under different models, we have made inference speeds for various LLMs, which are summarized as follows:



On the other hand, the advantages of small models are more direct, such as GPT-OSS 20B and Llama3.2 8B, which can reach speeds of 38 to 58 tokens/s, making them the first choice for scenarios that require real-time interaction or low latency.

As for the DeepSeekR1 series, the 8B is acceptable, but the 70B is a bit too difficult, with the speed dropping to only 4.81 tokens/s, making it almost impossible to use in real-time applications.



4. Measurement conclusion

Overall, I think NVIDIA Jetson Thor's performance is really surprising when inferring exabyte-level models locally. A behemoth like GPT-OSS 120B can still run a stable output speed of about 26 tokens/s, which is considered a medium to high level for generation tasks, completely exceeding my original expectations. Looking at resource usage, the memory eats up about 57%, which is not small, but at least there is still room to spare, which means that if you want to stretch the context for a longer period of time, or even add additional parallel tasks, the system still has room to handle it.

What's even more interesting is that the CPU is almost stressed, and most of the work is carried by the GPU/Tensor Cores, which shows the value of Jetson Thor as a dedicated AI accelerator: it doesn't rely on burning the CPU to support performance, but efficiently hands over inference tasks to the right hardware.

Finally, when it comes to application value, I think the MIC-743-AT-ES has successfully proven one thing: running **hyperscale models at the edge is no longer an impossible task. It can not only support LLMs with 120B parameters, but also maintain practical reasoning speed**, which opens up new possibilities for smart manufacturing, smart transportation, and even scenarios t

hat require local generative AI.