Final Report: Power Side-Channel Identification of Expert Activations in Mixture-of-Experts Language Models

Shenbo Xu
xushenbo857@gmail.com
Gabriel Kulp
gabriel@intseclab.org
Amir Nuriyev
amir.ravnur@gmail.com
Justin Kang
justin_kang@berkeley.edu

Supervised Program for Alignment Research — Spring 2026

Abstract

GPUs are central to modern AI infrastructure, but their physical side-channel leakage has received little scrutiny in comparison with CPUs or dedicated secure hardware. This report studies whether board-level power measurements from an NVIDIA H100 can reveal which experts activate during Mixture-of-Experts (MoE) language-model inference. Using a ChipWhisperer ADC trace from GPT-OSS-20B layer-2 decode-time expert windows, the strongest defensible strict waveform-only classifier reaches 24.7% top-1 and 63.2% top-5 accuracy over 32 experts, compared with a 3.1% random baseline. A controlled forced-expert harness reaches much higher accuracy, confirming that the sensor can observe expert-dependent work when the signal is repeated and isolated. However, natural decode traces remain noisy, imbalanced, and context-dependent. The main conclusion is that real expert-dependent power leakage exists, but a clean high-confidence per-expert fingerprint has not yet been achieved.

Keywords power side-channel · mixture-of-experts · GPU security · AI control · AI safety

1 Introduction

Modern AI systems increasingly run on large shared GPU clusters. Those clusters are usually analyzed as software and network security systems, but the hardware itself also emits physical signals: current draw, voltage changes, electromagnetic radiation, acoustic noise, and thermal behavior. If these signals correlate with model internals, then model weights, architecture choices, prompts, or outputs may leak even when the software stack appears locked down.

Mixture-of-Experts (MoE) language models are a particularly useful target for studying this risk. For each token, a router selects a small subset of expert feedforward networks. The selected experts are input-dependent, and prior work argues that expert activation sequences reveal substantial information about the underlying text [6]. If expert identity can be inferred from power measurements, then power traces become a path toward prompt or token reconstruction.

This project asks whether GPU power side channels can identify MoE expert activations during inference. The work proceeded in three stages. First, I tested a Hamming-weight matrix-reordering hypothesis in a controlled GEMM setting, asking whether sorting matrix rows by bit population

count made power traces more separable. Second, I built a forced-expert harness that activated one expert repeatedly inside a capture window, to test whether the measurement setup could detect expert-dependent computation under favorable conditions. Third, I moved to the harder natural-decode setting, where GPT-OSS-20B selects four experts per token and each expert appears only once in a short, noisy power window.

The final result is mixed but informative. Strict ADC-only natural-decode classification is far above chance, with the best run in the 24–25% top-1 band and top-5 around 63%. At the same time, macro accuracy remains only around 10%, and many rare experts collapse into common predictions. The signal is real, but current measurements do not yet support a balanced high-confidence 32-way waveform fingerprint.

2 Related Work

Physical side-channel attacks have a long history in cryptography, where power and electromagnetic measurements can reveal secrets even when the algorithm is mathematically sound. Similar ideas have been applied to neural networks: prior work has extracted DNN parameters from embedded devices and GPUs, and has reverse-engineered model architecture from electromagnetic or magnetic leakage [2, 3, 4, 5].

GPU power leakage is especially relevant because neural workloads are dominated by large matrix operations. Input-dependent power draw in GPU matrix multiplication has already been observed [1], suggesting that operand values and operation structure can affect physical measurements. However, showing generic input-dependent power is weaker than showing that a deployed LLM inference trace leaks semantically useful model state.

MoE routing makes that question concrete. Hayase et al. show that expert selections in MoE models can reveal almost as much about text as the text itself [6]. Their result turns expert identity into an information-bearing target. This report contributes a hardware-facing experiment: rather than observing expert IDs through software hooks, it tests how much of that expert identity can be recovered from board-level power traces during GPU execution.

3 Methods

3.1 Hardware and Model

Experiments used an NVIDIA H100 SXM4 GPU instrumented with a ChipWhisperer Husky-class ADC connected through a board-level current-sensing setup. The target model was `openai/gpt-oss-20b`, a 32-expert MoE language model with top-4 routing. The main real-decode experiments focused on layer 2, where each token causes four selected experts to execute. In the Hugging Face execution path used here, selected experts run sequentially in increasing expert-id order. That order is a real implementation side channel, but the strict results below exclude rank, duration, timestamps, router score, hook timing, and file order from the model input.

3.2 Tasks

I evaluated three related tasks. The Hamming-weight experiment created random float16 matrices, sorted rows by bit population count in ascending or descending order, and captured repeated GEMMs for the original, ascending, and descending conditions. This tested whether a simple bit-level proxy for switching activity created a visible GPU power difference.

Table 1: Strict and contextual real-decode result ladder. The 24–25% top-1 band is the most defensible headline for ADC-only natural decode.

Run	Top-1	Top-2	Top-3	Top-5	Macro	Interpretation
5MSPS masked pad8000, time/FFT, len2048, Inception FFT/SE	24.04%	38.70%	48.29%	62.61%	9.69%	Best clean older strict baseline.
18.75MSPS English80, contextual pad30000, masked len8192	24.71%	38.27%	48.47%	63.23%	10.73%	Best newer strict ADC-only run.
10MSPS smaller contextual run, mask pad8000, len8192	24.88%	43.81%	52.44%	66.63%	9.07%	Good top-k, but low macro and smaller scale.
27.44% tuned stress run	27.44%	43.25%	51.38%	63.00%	9.83%	Higher micro top-1, but not a clean balanced win.

The controlled forced-expert harness activated one expert at a time and repeated the expert workload many times inside each capture window. This is not the natural inference task, but it provides a sensor sanity check: if expert-dependent power cannot be recovered in this favorable setting, it is unlikely to be recovered during real decode.

The main task was strict ADC-only real-decode expert classification. Each example was a power crop around one layer-2 expert window during GPT-OSS-20B decode. The label was the expert ID for that window. The classifier received only deterministic features derived from the ADC waveform.

3.3 Features and Classifiers

For the best strict 5MSPS family, each example used a broad local crop: roughly 8000 raw samples of left context, the target expert window, and roughly 8000 samples of right context. Neighboring same-token expert windows that overlapped the crop were masked to local baseline when possible. The feature tensor combined time-domain samples, first differences, envelope, partial sums, and FFT-derived channels. The strongest model family was an InceptionTime-style 1D convolutional classifier with FFT and squeeze-excite style feature branches. I also tested plain CNNs, ResNet-style 1D models, TCN and BiLSTM variants, 2D STFT CNNs, and transformer variants.

For evaluation, I report top-k accuracy over 32 experts, macro accuracy, and qualitative confusion-pattern evidence. Random top-1 chance is 3.1%. The majority-class baseline is higher, around 9%, because natural prompts do not select experts uniformly.

4 Results

4.1 Natural Decode

The main result is that strict ADC-only real-decode traces contain expert-dependent information, but the waveform signal is weak and imbalanced. The best defensible high-rate run reaches 24.71% top-1 and 63.23% top-5 accuracy; an older 5MSPS run reaches 24.04% top-1 and 62.61% top-5. Both are far above the 3.1% random baseline and also beat the majority prior, but macro accuracy remains around 10%, showing that the model recovers some experts much better than others.

The top-k pattern matters. A top-5 result around 63% means the waveform often narrows the candidate expert set, even when it cannot reliably choose the exact expert. For a downstream structured attack, a probability vector that eliminates many experts may still be useful. For a clean

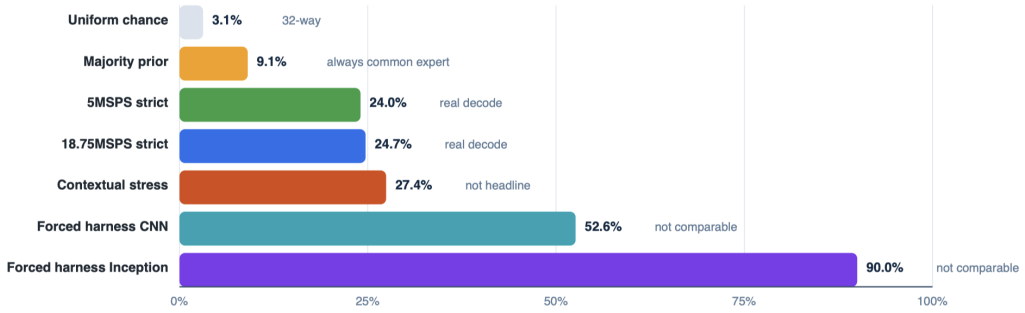

Figure 1: Accuracy summary across representative real-decode classifier runs. The useful headline is not only top-1: top-5 reaches roughly 63%, while macro accuracy remains low.

Table 2: Controlled forced-expert harness performance. This is a favorable sensor proof, not the strict natural-decode task.

Model	Top-1	Top-2	Top-3	ECE
CNN	53.2%	69.8%	79.0%	0.098
CNN+aug	52.6%	69.6%	78.6%	0.136
Mamba	52.9%	71.0%	80.1%	**0.034**
BiLSTM+aug	41.6%	59.8%	70.0%	0.048

claim that the waveform uniquely fingerprints all 32 experts, however, the low macro accuracy is a serious limitation.

4.2 What the Classifier Sees

The strict model does not receive a clean isolated matrix-multiplication island. It sees a short target expert region embedded in broader GPU power context, then transformed into a multi-channel time/frequency tensor. In one representative 5MSPS example, the crop is 18,490 raw samples (3.70 ms), while the target compute window is only 2,490 samples (0.50 ms), or 13.5% of the crop.

Forensic trace audits found that hook markers are usable for labeling, but the analog waveform does not expose a crisp onset or peak that always lands inside the compute window. Target-window RMS and neighboring context are often close, and local envelope peaks can fall outside the exact hook window. This limits how much tighter cropping or simple alignment can recover.

4.3 Controlled Harness and Hamming Test

The controlled forced-expert harness achieved much higher classification accuracy. A 1D CNN reached 53.2% top-1 and 79.0% top-3 accuracy on a balanced 32-way task; a Mamba SSM reached 52.9% top-1 and 80.1% top-3 with better calibration. These results prove that the measurement setup can observe expert-dependent computation when one expert is isolated and repeated many times. They should not be quoted as the natural-decode result, because real decode gives only one short expert execution per routed token.

The Hamming-weight row-reordering experiment produced a null result. Across 500 traces per condition and $T = 200{,}000$ time points, no pair survived Bonferroni or Benjamini-Hochberg correction. The proportion of samples with $|t| > 2$ was approximately 1.3–1.9%, consistent with chance.

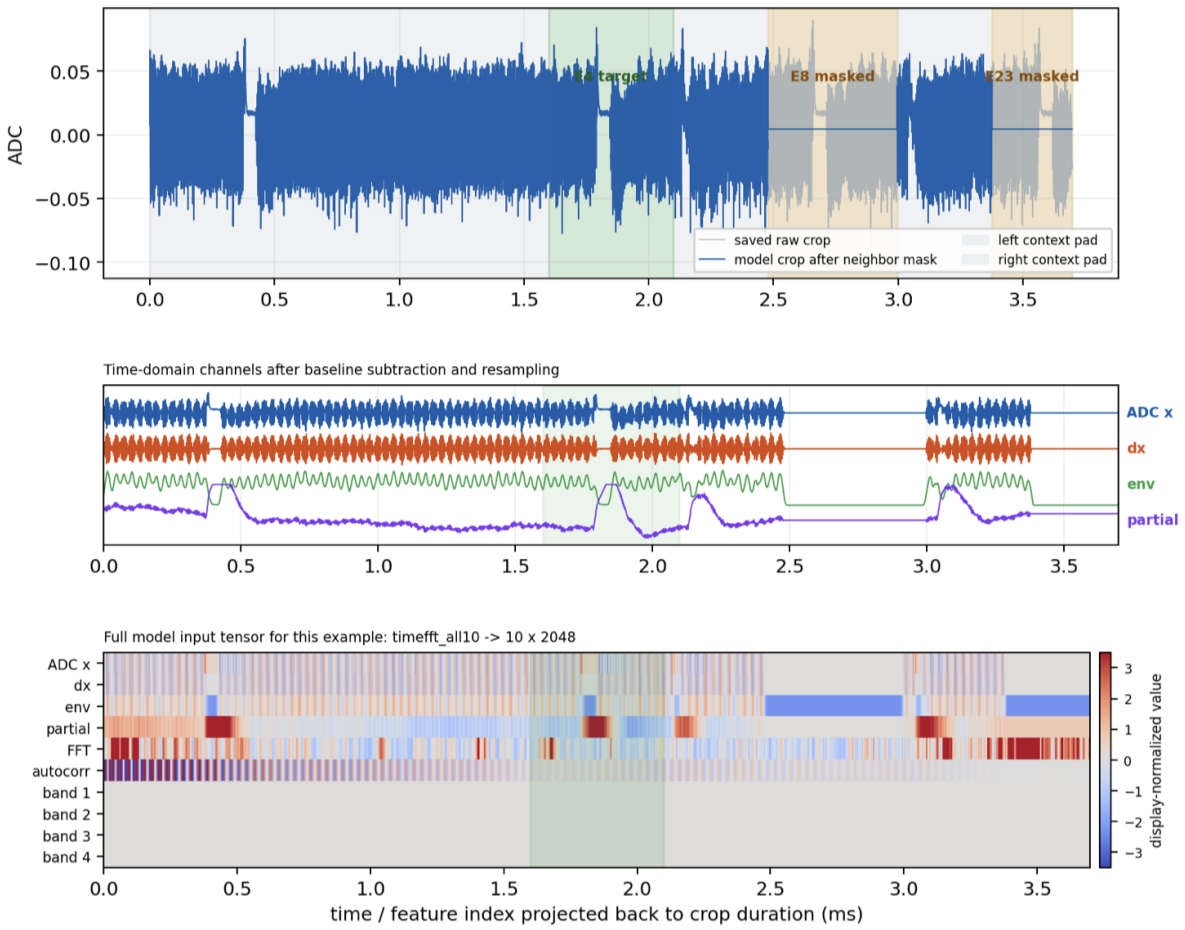

Figure 2: Actual model-input visualization for one layer-2 example from the 24.04% run. The target expert window is a small part of a broader local ADC crop, which explains why broad context helps but also makes the result less like a pure target-only fingerprint.

Table 3: Pairwise separability of power traces under Hamming-weight row permutation. No samples survive multiple-comparisons correction.

Pair	max SNR	mean SNR	max $\lvert t\rvert$	pct $\lvert t\rvert > 2$	pct Bonf.	pct BH
orig vs hw_asc	0.0416	1.41×10^{-3}	4.56	1.62%	0.0%	0.0%
orig vs hw_desc	0.0247	1.79×10^{-3}	3.51	1.92%	0.0%	0.0%
hw_asc vs hw_desc	0.0361	1.31×10^{-3}	4.24	1.30%	0.0%	0.0%

5 Discussion

5.1 Why Accuracy Stalls

The gap between the forced harness and natural decode is the central finding. The harness repeats a clean expert workload many times, increasing signal-to-noise ratio inside a single trace. Natural decode gives one short expert occurrence per token, with the same-shaped MLP kernels for all experts, global board-level power mixing, and non-uniform expert selection. The classifier therefore learns a combination of expert-dependent waveform structure, local context, and prompt-dependent routing priors.

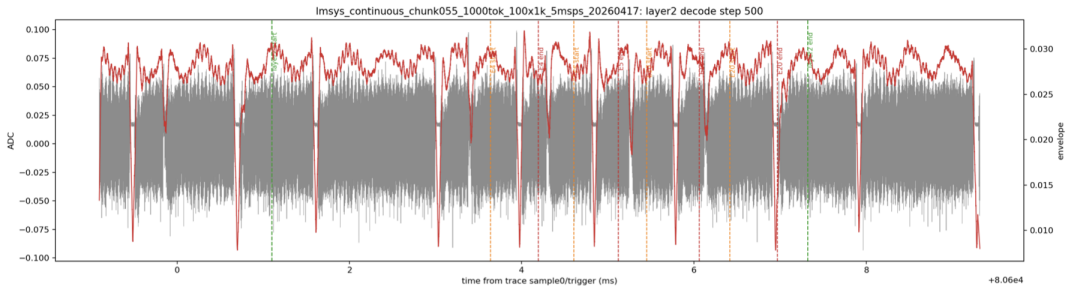

Figure 3: Representative decode trace with expert-window overlays. The markers are in bounds and visually plausible, but the power waveform is a global mixed measurement rather than a perfectly isolated expert pulse.

Figure 4: Expert selection is non-uniform in natural decode. This makes micro top-1 a useful attack metric but a weak proxy for balanced expert separability.

Class imbalance is especially important. Natural prompts do not select all experts equally, and many confusion matrices show collapse toward common or easy experts. Micro top-1 can rise while rare experts remain poorly separated. That is why the 27.44% stress result is not the headline: it improves micro accuracy but leaves macro accuracy below 10%, which is consistent with stronger use of contextual or prior structure rather than a balanced 32-expert waveform fingerprint.

5.2 Sample Rate and Model Capacity

Increasing sample rate did not produce a step change. The 10MSPS and 18.75MSPS experiments landed in the same 24–25% top-1 band as the best 5MSPS family when evaluated under strict ADC-only constraints. This does not prove higher-rate capture is useless, but it suggests that modestly increasing sample rate is not the missing ingredient. The bottleneck is more likely measurement isolation, crop quality, analog bandwidth, probe placement, and repetition.

Architecture sweeps showed a similar pattern. Inception-style 1D convolutions with time, derivative, envelope, partial-sum, and FFT features were consistently strongest. Transformers and more exotic spectral features did not break the 30% barrier. This points away from model capacity as the main limitation: the classifier can exploit visible structure, but the current crop does not isolate enough repeatable expert-specific signal.

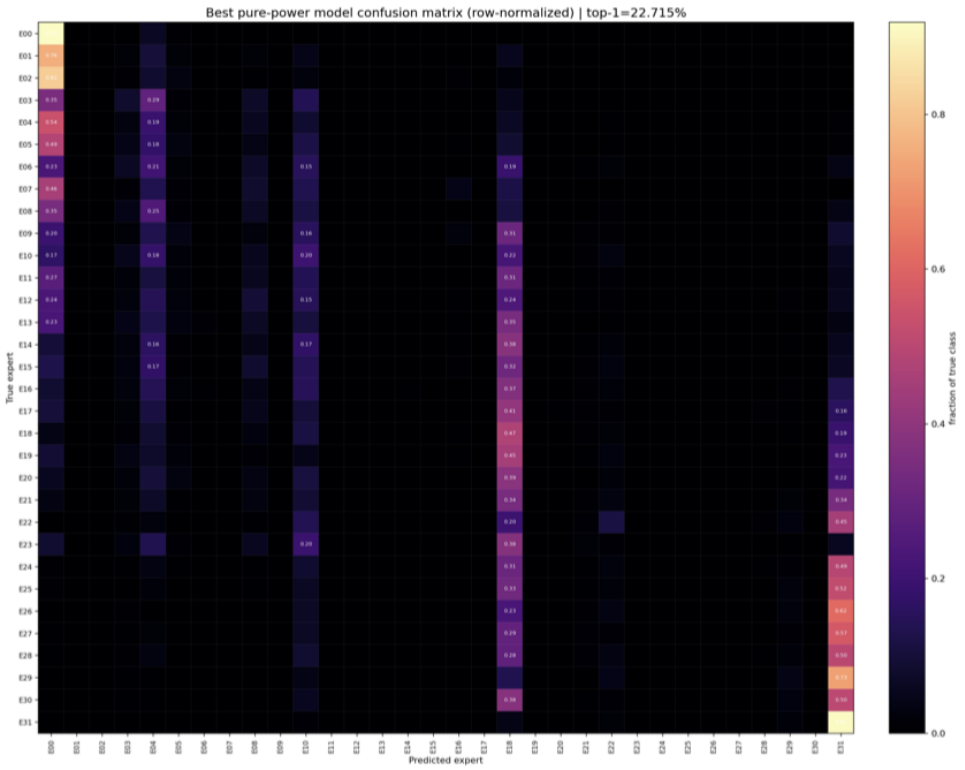

Figure 5: Representative strict power-only confusion matrix. The recurring pattern is class collapse: a few experts are recovered much more often than many rare experts.

5.3 Implications and Next Steps

The practical attack path is likely structured rather than purely single-window classification. Power logits could be combined with known implementation facts, such as monotonic expert-id execution order, train-only expert priors, and aggregation over multiple occurrences of an expert. Those side channels should be reported separately from strict waveform evidence, but they are realistic in an end-to-end attack.

Future measurement work should be data-first. Before collecting another large dataset, the capture pipeline should include pre-arm triggering, FIFO warning checks, in-bounds marker validation, range and standard-deviation quality gates, and visual audits of early chunks. A useful recapture should also test repetition or aggregation in a real-decode setting. The old harness succeeded because repetition increased SNR; a realistic attack may need the same principle across tokens rather than inside one artificial capture.

6 Conclusion

This project demonstrates a real but limited GPU power side channel for MoE expert identity. On natural GPT-OSS-20B decode traces, strict ADC-only classification reaches 24–25% top-1 and about 63% top-5 accuracy over 32 experts, far above the 3.1% random baseline. However, macro accuracy around 10% and visible class collapse mean the result is not yet a clean balanced expert fingerprint. The controlled forced-expert harness shows that expert-dependent power is measurable when the workload is repeated and isolated, while the Hamming-weight reordering experiment

rejects a simple bit-popcount explanation. The best next step is not another blind model sweep, but better measurement isolation, quality-gated recapture, and structured decoding that combines waveform evidence with explicitly separated implementation side channels.

References

[1] Jiang et al., "Input-Dependent Power Usage in GPUs," 2024.

[2] Mathieu et al., "A Practical Introduction to Side-Channel Extraction of Deep Neural Network Parameters," IACR, 2023.

[3] Horvath et al., "BarraCUDA: Edge GPUs Do Leak DNN Weights," 2023.

[4] Batina et al., "CSI NN: Reverse Engineering of Neural Network Architectures Through Electromagnetic Side Channel," USENIX Security, 2019.

[5] Maia et al., "Can One Hear the Shape of a Neural Network? Snooping the GPU via Magnetic Side Channel," USENIX Security, 2022.

[6] Hayase et al., "Expert Selections in MoE Models Reveal (Almost) As Much As Text," 2024.