

Chronometric Injection: A Residual-Stream Channel for Elapsed-Time Conditioning in Frozen Language Models

Sam Siavoshian^{*a}, Omar Ramadan^b, Amir Kashif Saeed^b, Benjamin A. Johnson^b, Amin Mohamed El-Amin Diab^b, and Benjamin M. Rodriguez^b

^aIndependent Researcher, California, USA

^bJohns Hopkins University, Whiting School of Engineering, Baltimore, MD, USA

ABSTRACT

Large language models can reason about time when time is written in the prompt, but the standard transformer interface provides no first-class input for elapsed wall-clock duration. This matters for interactive and agentic systems, where the same visible prompt may require different responses depending on whether seconds, hours, or days have passed. This paper studies whether a frozen-base-weight autoregressive language model can be conditioned on a continuous elapsed-time scalar without updating its base weights. We introduce *chronometric injection* (CI), a residual-stream conditioning channel that encodes elapsed seconds τ with sinusoidal and log-arithmetic features and injects the resulting vector into a frozen Qwen2.5-3B-Instruct model through zero-gated AdaLN-style FiLM projectors. The contribution is not a new modulation primitive, but a causal and mechanistic characterization of an elapsed-time channel inside an otherwise frozen language model. Linear probes show that $\log \tau$ is unrecoverable at layer 0, becomes nearly perfectly recoverable after one transformer block, remains recoverable through the network, and collapses to chance when the FiLM gates are zeroed. Behavioral results are more limited: the original zero-shot Temporal Policy Switching test remains negative, while direct policy supervision makes hidden elapsed time usable for forced-choice action control under identical visible prompts, reaching 0.8709 ± 0.0303 hidden-only accuracy. Overall, CI is best understood as a traceable residual-stream channel for elapsed-time conditioning, not as evidence of autonomous time tracking, broad policy generalization, or superiority over prompt-injected timestamps.

Keywords: chronometric injection, elapsed-time conditioning, temporal policy switching, hidden-time action control, frozen language models, FiLM, AdaLN-Zero, LoRA, mechanistic interpretability, causal ablations

1. INTRODUCTION

Autonomous and semi-autonomous systems operating in contested or time-critical environments must condition decisions on wall-clock elapsed time. Cache freshness in intelligence, surveillance, and reconnaissance (ISR) pipelines, session continuity in human-machine teaming, tool-call latency in multi-agent workflows, and urgency of response after a silent gap all depend on a continuous scalar that is independent of the visible text prompt. Large language models (LLMs) already possess rich textual knowledge of calendars, durations, and deadlines, yet the standard transformer interface supplies only token position. Real elapsed seconds are invisible unless an external system serializes them into the prompt.

This gap matters for defense and security applications. Consider a concrete ISR vignette: an analyst (or an autonomous agent acting on the analyst’s behalf) issues the identical natural-language request “update the track file for contact Alpha” once when the underlying sensor product is ten seconds old and again when the same product is twelve hours old. Reusing the ten-second-old product is operationally correct; reusing the twelve-hour-old product can produce a dangerous situational picture. Prompt-injected timestamps remain discrete tokens that must be attended to and interpreted at generation time; they do not constitute a first-class, continuous side channel that can modulate hidden states while leaving the visible prompt byte-identical. The same distinction arises for session continuity (resume after twenty seconds versus after three days) and for multi-agent tool-use policies that depend on cache age.

^{*}Further author information: Send correspondence to Sam Siavoshian.

E-mail: samsiavoshian2009@gmail.com

The technical question is therefore architectural: can a frozen pretrained autoregressive LLM be given a continuous elapsed-time scalar τ such that the scalar enters the residual stream, is causally used by the model, and can be isolated by intervention? We introduce *chronometric injection* (CI), a residual-stream conditioning channel that encodes elapsed seconds with multi-scale sinusoidal and logarithmic features and injects the resulting vector into a frozen Qwen2.5-3B-Instruct model through zero-gated AdaLN-style FiLM projectors. Only the chrono projectors, gates, and a small LoRA surface are trained; the base weights remain frozen.

The contribution is deliberately scoped. CI demonstrates that an externally supplied continuous time scalar can be turned into a recoverable, intervention-friendly residual-stream channel. It does not claim autonomous internal time tracking, subjective time perception, or universal superiority over prompt timestamps. The strongest evidence is mechanistic and causal: linear probes recover $\log \tau$ after a single transformer block, the signal collapses when the FiLM gates are zeroed, and sign-flip interventions invert time-conditioned behavior. Behavioral results are narrower: zero-shot temporal policy switching fails, while direct policy supervision enables hidden-time forced-choice action control under identical visible prompts.

The paper is organized as follows. Section 2 reviews related work and formalizes the problem. Section 3 describes the chronometric encoder, residual-stream injection, training surface, and evaluation metrics. Section 4 presents results. Section 5 discusses implications, limitations, and future directions for defense-relevant agent systems.

2. BACKGROUND AND RELATED WORK

2.1 Related Work

Timestamped and time-aware language models incorporate temporal metadata to improve temporally scoped facts or event ordering.^{1,2} Time-aware prompting, temporal soft prompts, and prefix tuning condition generation through visible text or continuous prompt vectors while leaving most base weights fixed.^{3–5} These approaches operate at the token or attention interface. CI instead supplies elapsed seconds as a hidden runtime scalar and injects it through gated residual-stream modulation, leaving the visible prompt unchanged.

The chronometric encoder draws on sinusoidal positional encodings, Time2Vec, and continuous numerical tokenization methods such as xVal.^{6–8} We employ a fixed multi-scale sinusoidal-plus-log basis solely as a practical covering of seconds-to-weeks variation; the scientific focus is whether the resulting vector becomes a recoverable and causal channel inside a frozen LLM.

Feature-wise linear modulation (FiLM), adaptive layer normalization, and zero-initialized conditioning mechanisms supply the architectural precedent for side-information modulation of hidden states.^{9,10} LoRA and IA³ provide parameter-efficient action surfaces over frozen models.^{11,12} Mechanistic studies have shown that structured variables can sometimes be linearly recovered from language-model activations.¹³ CI differs by constructing an explicit elapsed-time pathway and testing it with zero-gate controls, sign flips, layer localization, held-out supervision, and hidden-only policy switching.

2.2 Problem Definition

Let f_θ be a frozen autoregressive decoder-only language model with parameters θ . Let x be the visible prompt tokens and let $\tau \in \mathbb{R}_{\geq 0}$ be elapsed wall-clock seconds since a reference event (session start, last tool call, last user turn, etc.). The target is a conditioned model $g(x, \tau)$ that can use τ while leaving θ unchanged.

Success is defined by three criteria:

1. *Transmission*: τ is recoverable from residual-stream hidden states only when the chrono channel is active.
2. *Causality*: interventions on the learned gates change time-conditioned behavior in the expected direction.
3. *Behavioral scope*: observed behaviors reflect genuine channel use rather than supervised format recall or prompt artifacts.

The scope is intentionally limited. CI does not model physical time passing inside the transformer between forward passes; it accepts an externally supplied scalar at inference time. If that scalar is incorrect or adversarial, the model conditions on the wrong value. The paper therefore studies an architectural input pathway for trusted runtime variables, not autonomous temporal consciousness or persistent internal recurrence.

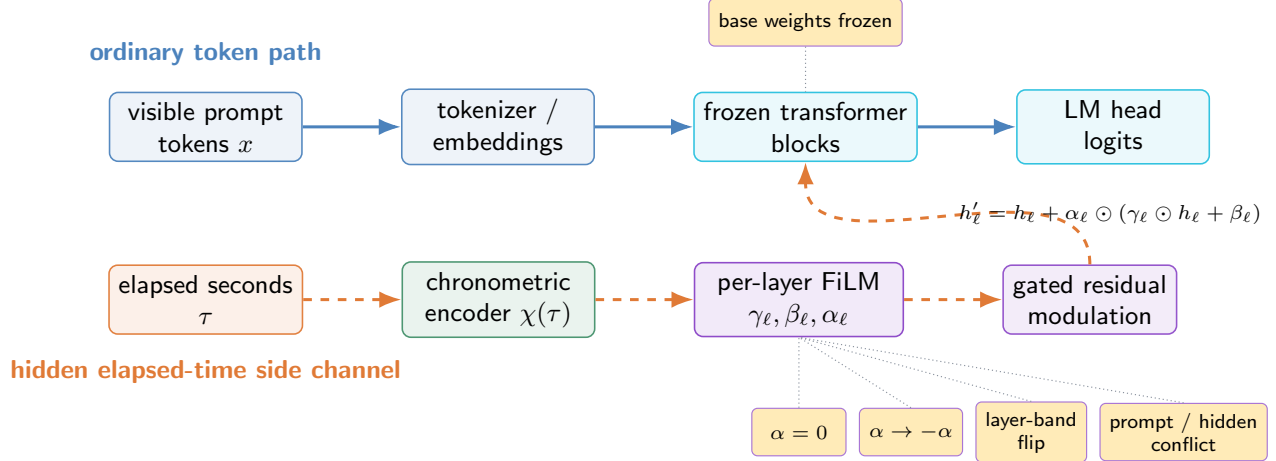


Fig. 1. Chronometric injection pipeline. The visible prompt follows the ordinary autoregressive token path, while elapsed seconds τ enter through a separate non-token side channel. The encoded scalar $\chi(\tau)$ is projected into per-layer FiLM parameters and gated into the residual stream, enabling probe, zero-gate, sign-flip, and policy-switching tests.

In defense and security systems the distinction is operationally useful. When time must be inspected or explained to a human operator, ordinary prompt text remains the transparent interface. When elapsed time is a trusted system variable (cache age, time since last ISR update, time since last human confirmation) that should modulate policy without altering the visible prompt, a residual-stream channel is the appropriate design.

The scalar τ always denotes elapsed seconds; the reference event is task-specific and is supplied by the surrounding system or data generator. The model is never required to infer the anchor from its own internal state.

3. METHODOLOGY

3.1 Overview

Chronometric injection adds a continuous elapsed-time side channel to a frozen LLM. The method has three stages. First, elapsed seconds τ are mapped to a fixed 31-dimensional encoding $\chi(\tau)$. Second, each injected decoder layer projects $\chi(\tau)$ into feature-wise scale and shift parameters. Third, a zero-gated initialization ensures that at step zero the model is identical to the frozen base model. Only the chrono projectors, gates, and an optional low-rank adapter are trained.

Figure 1 summarizes the resulting two-path interface: the ordinary autoregressive token path remains unchanged, while the hidden elapsed-time path enters through gated residual-stream modulation.

3.2 Chronometric encoder

The chronometric encoder maps elapsed seconds into sinusoidal features at multiple scales plus a monotone logarithmic coordinate:

$$\chi(\tau) = \begin{bmatrix} \sin(\tau/T_1), \cos(\tau/T_1), \dots, \\ \sin(\tau/T_{15}), \cos(\tau/T_{15}), \log(1 + \tau) \end{bmatrix}^\top. \quad (1)$$

The scale divisors are

$$T \in \{2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, \\ 4096, 16384, 65536, 86400, 604800\} \text{ seconds.}$$

The implementation uses $\sin(\tau/T_k)$ and $\cos(\tau/T_k)$ directly, without a 2π multiplier; therefore T_k is a radian-scale divisor and the mathematical sinusoid period is $2\pi T_k$. The largest divisors include day and week scales. The $\log(1 + \tau)$ coordinate provides a nonperiodic monotone direction. The encoder has 31 output dimensions and no learned parameters.

The exact periods are a pragmatic coverage choice rather than an optimized representation. The divisors cover seconds, minutes, hours, day-scale, and week-scale variation, while the logarithmic feature separates long durations even when sinusoidal components wrap. We intentionally avoid making the encoder adaptive in the main study so that the causal analysis focuses on the residual channel and gates rather than on a learned temporal-feature extractor.

3.3 Residual-stream injection

At decoder layer ℓ , let $h_\ell \in \mathbb{R}^d$ denote the residual-stream hidden state. The chrono projector computes

$$\gamma_\ell = W_\ell^\gamma \chi(\tau) + b_\ell^\gamma, \quad (2)$$

$$\beta_\ell = W_\ell^\beta \chi(\tau) + b_\ell^\beta, \quad (3)$$

and updates the residual stream as

$$h'_\ell = h_\ell + \alpha_\ell \odot (\gamma_\ell \odot h_\ell + \beta_\ell), \quad (4)$$

where $\alpha_\ell \in \mathbb{R}^d$ is a learned per-channel gate. The main configuration injects this channel into 35 of 36 decoder layers.

For implementation, the same τ -conditioned vectors γ_ℓ , β_ℓ , and α_ℓ are broadcast over all token positions in the sequence. Thus, for a hidden-state tensor $H_\ell \in \mathbb{R}^{B \times T \times d}$, the operation in (4) is applied feature-wise over the final dimension and shared over batch and token positions for a given example. The projectors are linear maps from the 31-dimensional encoder to the model width. In the implementation used for the reported checkpoints, the FiLM update is applied at the selected decoder-layer residual-stream hook exposed by the model wrapper, after the layer’s ordinary hidden state has been computed and before it is passed to the next decoder layer. The final decoder layer is skipped to leave the last language-model readout surface less directly perturbed and to match the reported A+B configuration. This placement detail is part of the experimental interface rather than a claim that this is the only viable insertion point.

This formulation makes the elapsed-time pathway explicit. The scalar τ is not represented as a prompt token. It enters through $\chi(\tau)$, affects layerwise feature transformations through $(\gamma_\ell, \beta_\ell)$, and is controlled by the gate α_ℓ .

3.4 Zero-gated initialization

The channel uses an AdaLN-Zero-style, zero-gated initialization:

$$\alpha_\ell = 0, \quad W_\ell^\gamma = 0, \quad b_\ell^\gamma = \mathbf{1}, \quad W_\ell^\beta = 0, \quad b_\ell^\beta = 0. \quad (5)$$

At step 0, (4) reduces to $h'_\ell = h_\ell$, so the base model is unchanged. However,

$$\left. \frac{\partial h'_\ell}{\partial \alpha_\ell} \right|_{\text{init}} = \gamma_\ell \odot h_\ell + \beta_\ell = h_\ell, \quad (6)$$

which gives the gate a nonzero gradient on the first backward pass. This initialization is important: additive variants with zero shift initialization can trap the gate at zero because the initial gate gradient vanishes.

3.5 Trainable surface

The main model also trains a rank-8 LoRA adapter on attention projections and the language-model head. LoRA is not assumed to encode time by itself. It supplies a small action surface over the chrono-modulated hidden states. In the original in-house and Track A setting, the key ablation freezes α at zero while leaving LoRA trainable; that LoRA-only model fails all pass-style time-conditioned checks, showing that the adapter alone does not explain the mechanistic-channel results. This statement is scoped to the original in-house setting. In Track B, the LoRA-only policy control is nontrivial, reaching 0.5814 overall accuracy and 0.5741 hidden-only accuracy, so it is treated as a weaker supervised policy baseline rather than as a universal failure case.

Operational summary. Given prompt tokens x and elapsed seconds τ , CI computes $\chi(\tau)$, projects that encoding into per-layer FiLM parameters, applies the gated residual update in (4), and trains only the chrono parameters plus the specified PEFT surface. The same explicit gates define the main interventions used below: zeroing the channel, flipping gate signs, and perturbing selected layer bands.

3.6 Evaluation protocol and metrics

The evaluation asks five questions:

1. Does the elapsed-time signal become linearly recoverable in the residual stream?
2. Is the chrono channel necessary and causal?
3. Which behavioral tests survive when specific supervised labels are removed?
4. How does residual injection compare with prompt-injected timestamps on direct recall?
5. Can hidden elapsed time control forced-choice agent policy under identical visible prompts (zero-shot versus supervised)?

Experiments use Qwen2.5-3B-Instruct as the frozen backbone,¹⁴ with a limited 7B scale check. Training data consist of 18 000 synthetic conversations spanning CLOCK, SILENT-GAP, and PHASE families; elapsed times are sampled from [1 s, 7 days]. Mechanical transmission is measured by ridge probes predicting $\log_{10} \tau$ from last-token residual states. Main 3B results are reported across three seeds where feasible; base weights remain frozen in all experiments.

Behavioral and external metrics. For clarity we define the principal readout metrics used in Section 4:

- **T1 / T1b (clock readout):** Pearson correlation (and log-mean absolute error) between the model’s parsed duration string and true τ , evaluated in-distribution (T1) and on held-out durations (T1b).
- **T2 (acknowledgement):** Difference in the rate at which the model explicitly acknowledges a large inter-turn gap when τ is large versus small.
- **T3 (phase discrimination):** Accuracy of weekday-versus-weekend greeting conditioned on the day/week phase encoded by τ .
- **T4 (distributional shift):** Mean pairwise KL divergence between next-token distributions obtained at different values of τ under a fixed prompt.
- **TPDR:** A set of ambiguous decision prompts whose text does not mention time; the primary endpoint is a lexical `deliberative_score` correlated against $\log \tau$.
- **tau.sessions:** An external composite benchmark measuring staleness accuracy, duration recall, adaptive response length, and an aggregate score under prompt-injected versus residual τ .

Temporal Policy Switching (TPS) evaluates whether elapsed time changes an action policy rather than a duration string. Each item is scored as a forced choice among four actions (reuse, refresh, ask, summarize). The diagnostic condition is `hidden_only`: the visible prompt is held fixed while only the hidden scalar τ changes. Track A tests zero-shot transfer from the original training; Track B trains directly on TPS policy labels. Track B evaluation contains 2 592 items across six families and nine elapsed-time values spanning 10s to 7 days. Gold labels follow deterministic family-threshold policies. Controls include zero-gate ablation, LoRA-only and chrono-only training, sign flips, layer-band interventions, held-out-supervision retraining, and prompt-injected baselines.

4. RESULTS AND ANALYSIS

4.1 Mechanical transmission

Linear ridge probes predicting $\log \tau$ from last-token residual states establish the primary mechanical result. In the trained model the probe is at chance at layer 0 ($R^2 = -0.005$), rises to $R^2 = 0.99990$ after a single transformer block, and remains above 0.9995 through the final layer. When the FiLM gates are zeroed, every layer returns to chance. The recoverable time axis is therefore produced by the chrono channel rather than by the unmodified base model.

A CLOCK-heldout probe (CLOCK examples removed from training) still yields layer-1 $R^2 = 0.99984 \pm 0.00007$ across three seeds. The axis does not require duration-string supervision; it is a direct consequence of FiLM injection and early transformer processing. Figure 2 shows the active, zero-gate, and held-out conditions.

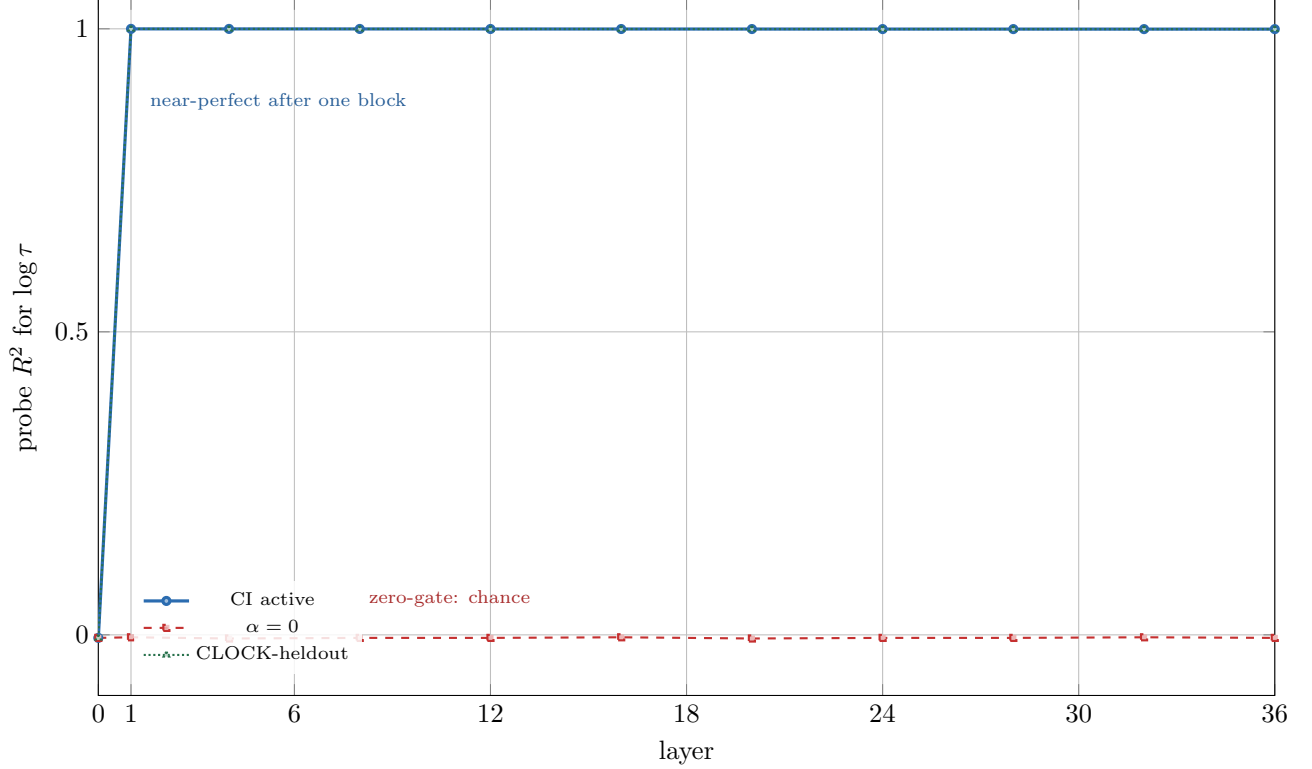


Fig. 2. Headline probe recoverability for $\log \tau$. The active chrono channel is at chance at layer 0, becomes nearly perfectly recoverable by layer 1, remains above the reported lower bound through the network, and returns to chance when the FiLM gates are zeroed.

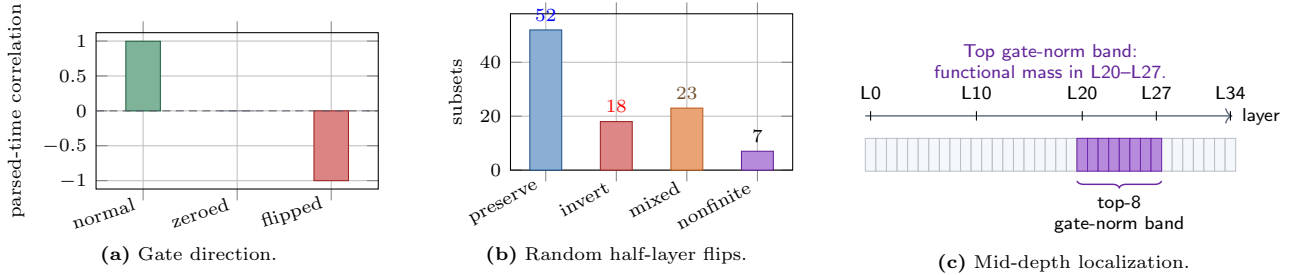


Fig. 3. Causal gate interventions on the chrono channel. Left: normal gates preserve the parsed T1 time axis, zeroing removes it, and flipping all gates inverts it. Middle: random half-layer sign flips yield preserved, inverted, mixed, and nonfinite outcomes, indicating a distributed signed pathway. Right: gate-norm localization concentrates the largest gate norms in L20–L27, matching the strongest layer-band intervention evidence.

4.2 Causal necessity

In the original in-house evaluation, LoRA-only training (gates frozen at zero) causes every pass-style behavioral check to fail across three seeds. Chrono-only training (LoRA frozen) passes the same checks. Gate sign-flip interventions invert parsed time predictions with correlation $r = -0.9998$. Random half-layer sign flips over 100 subsets produce a mixture of preserved, inverted, mixed, and non-finite outcomes, consistent with a distributed signed pathway. Gate-norm analysis localizes the strongest functional mass to layers L20–L27. Figure 3 summarizes these interventions.

4.3 Behavioral scope

Table 1 collects the central quantitative results. Direct clock-readout tasks (T1/T1b; see Section 3.6) achieve high correlations, yet a non-LLM linear model on $\chi(\tau)$ already reaches $r = 0.9965$; these tests primarily confirm that

Table 1. Main evidence after reframing. Strongest claims are mechanical transmission and causal use; behavioral claims are narrower.

Result family	Observed result	Interpretation	Boundary
Per-layer probe	L0 $R^2 = -0.005$; L1 $R^2 = 0.99990$; ≥ 0.9995 through L36; zero-gate chance	τ transmitted via chrono path	In-distribution probe
CLOCK-heldout probe	L1 $R^2 = 0.99984 \pm 0.00007$	Axis is mechanical, not duration-string dependent	Does not imply behavior without training
Track A checks	T1 $r = 0.9605 \pm 0.0349$; T1b $r = 0.9927 \pm 0.0031$; T2 $\Delta = 1.000$; T4 KL 0.1783 ± 0.0821	Channel usable under trained checks	Mostly wiring/format evidence
LoRA-only (orig. tests)	All pass-style checks fail (3 seeds)	Chrono gate necessary for those checks	Does not imply Track B LoRA-only collapse
Chrono-only ablation	Passes in-house tests with LoRA frozen	LoRA not required for sufficiency	LoRA improves precision/stability
All-layer sign flip	Parsed T1 inverts, $r = -0.9998$	Gate direction is causal	Small parseable n
Half-layer flips	$ r > 0.5$: 52 preserve, 18 invert, 23 mixed, 7 nonfinite	Pathway distributed and signed	Importance varies by seed
Held-out CLOCK	T1/T1b fail; T2, T4 pass 3/3	Clock readout is format recall; other use survives	T2 supervised by SILENT-GAP
Prompt baseline	Prompt wins direct recall and composite external score	CI not a universal replacement for timestamps	<code>tau_sessions</code> shows no CI advantage
TPDR v2	Headline pair $t = -5.167$, $p = 2.633 \times 10^{-6}$, $n = 46$	Exploratory response-shape effect	Other seed pairs unstable

the channel can be read out. When CLOCK supervision is removed, T1/T1b fail while T2 (acknowledgement) and T4 (pairwise KL shifts) continue to pass, indicating that usable elapsed-time information reaches downstream behaviors trained on non-CLOCK objectives.

On the external `tau_sessions` benchmark, prompt-injected timestamps outperform residual injection on staleness accuracy, duration recall, adaptive length correlation, and the composite score. Residual injection is therefore not a universal replacement for prompt timestamps; its value lies in the mechanistically traceable non-token channel.

TPDR v2 yields a statistically significant response-shape effect for the headline seed pair only; cross-seed stability is limited. The result is treated as exploratory.

4.4 Temporal Policy Switching

TPS measures whether elapsed time can change an action policy under a byte-identical visible prompt. Track A (zero-shot transfer from the original CLOCK/SILENT-GAP/PHASE training) remains negative: hidden-only accuracy is 0.3436 ± 0.0047 , statistically indistinguishable from the vanilla baseline, and long-action probability is non-monotonic with $\log \tau$.

Track B (direct policy supervision) succeeds. Overall accuracy reaches 0.7557 ± 0.0228 ; the critical hidden-only condition reaches 0.8709 ± 0.0303 (balanced 0.8562 ± 0.0356). Long-action probability rises with $\log \tau$ (Pearson/point-biserial $r = 0.6905 \pm 0.0266$). In prompt/CI conflict items the hidden scalar dominates (0.8709 scalar-follow versus 0.0895 prompt-follow), yet explicit abstain/revalidate behavior remains rare. Held-out-family accuracy on `market_data` is weak (0.3912 ± 0.0845), confirming that broad domain transfer is not established. Table 2 reports the headline numbers and controls.

4.5 Discussion

The results support a causal residual-stream channel for continuous elapsed time inside a frozen LLM. The signal is recoverable after one block, collapses under zero-gate ablation, survives removal of clock-readout supervision, and responds to sign-flip interventions. Once policy labels are supplied, the same channel can drive forced-choice action selection under identical visible prompts—the operational capability needed for the ISR vignette in Section 1, in which an agent must reuse or refresh a track file under a byte-identical user query according to data age.

The results do not support claims of autonomous time tracking, general superiority over prompt timestamps, or broad zero-shot policy generalization. Prompt injection remains preferable for explicit time recall; residual

Table 2. TPS headline results. Track A: zero-shot transfer. Track B: direct policy supervision. $\pm$ values are cross-seed sample standard deviations over three CI seeds. Monotonicity is Pearson/point-biserial correlation between $\log \tau$ and the binary long-action indicator.

Track	Metric	Value	Interpretation
A zero-shot	CI hidden-only accuracy	0.3436 ± 0.0047	No zero-shot policy transfer
A zero-shot	Vanilla hidden-only accuracy	0.3488	Base policy comparable
A zero-shot	r ($\log \tau$ vs. long action)	-0.2704 ± 0.0767	Long action does not rise with τ
B policy-trained	Overall accuracy	0.7557 ± 0.0228	Supervision teaches TPS policy
B policy-trained	Overall balanced accuracy	0.6994 ± 0.0355	Gains under class balancing
B policy-trained	Hidden-only accuracy	0.8709 ± 0.0303	Main identical-prompt result
B policy-trained	Hidden-only balanced accuracy	0.8562 ± 0.0356	Not only class skew
B policy-trained	Both-agree accuracy	0.8735 ± 0.0334	Visible and hidden time agree
B policy-trained	r ($\log \tau$ vs. long action)	0.6905 ± 0.0266	Long-action rises with τ
B diagnostic	Prompt-only diagnostic acc.	0.4074 ± 0.0000	Same adapter; hidden τ zeroed
B controls	Vanilla / LoRA-only / chrono-only	0.4182 / 0.5741 / 0.7963	Available controls (latter two single-seed)
B policy-trained	Conflict scalar- / prompt-follow	0.8709 ± 0.0286 / 0.0895 ± 0.0178	Hidden scalar dominates
B policy-trained	Conflict abstain/revalidate	0.0216 ± 0.0215	Conflict detection rare
B policy-trained	Held-out-template accuracy	0.7130 ± 0.0197	Some template generalization
B policy-trained	Held-out-family (market_data)	0.3912 ± 0.0845	Broad family transfer not established

injection is valuable when elapsed time is a trusted runtime variable that must modulate behavior without appearing in the user-visible prompt.

Limitations and threats to validity. The evaluation is narrow: one primary base-model family, synthetic in-distribution training over [1s, 7 days], and author-built TPDR and TPS benchmarks. Critically, a fully fair architectural comparison still requires prompt+LoRA, soft-prefix/prefix-tuning, and matched scalar-embedding baselines trained on the same TPS labels; the reported prompt-only diagnostic is not that baseline. Held-out-family transfer remains weak. These limits bound the claims to a causal channel demonstration plus supervised hidden-time action control, not general temporal policy learning or proven superiority of residual FiLM over other continuous side channels.

5. CONCLUSION AND FUTURE WORK

This paper has shown that continuous elapsed wall-clock time can be injected into a frozen autoregressive language model as a causal residual-stream channel. The channel is formed by a fixed multi-scale sinusoidal-plus-log encoding of elapsed seconds and zero-gated AdaLN-style FiLM modulation of the residual stream. Linear probes recover $\log \tau$ after a single transformer block; the signal disappears when the gates are zeroed and inverts under sign-flip interventions. Direct policy supervision further enables the hidden scalar to control forced-choice actions under byte-identical visible prompts, reaching 0.8709 ± 0.0303 hidden-only accuracy.

The contribution is architectural and mechanistic: an externally supplied continuous time scalar can be made into a traceable, intervention-friendly input pathway for a frozen LLM. The work does not claim autonomous internal time tracking, broad zero-shot policy generalization, or universal superiority over prompt-injected timestamps. For defense and security systems, the practical value is an interface that lets a trusted runtime variable (cache age, time since last ISR update, or time since last human confirmation) modulate policy without rewriting the visible prompt, as illustrated by the track-file reuse/refresh vignette in Section 1.

Future work should examine the channel on additional base-model families and larger scales, integrate it with persistent memory or recurrent state for longer-horizon temporal reasoning, and complete the policy-generalization evaluation with fair prompt-trained, soft-prefix, and matched scalar-embedding baselines together with leave-one-family-out folds. In the defense domain, the natural next applications are time-conditioned cache management in ISR pipelines, urgency modulation in human-machine teaming, and policy switching for multi-agent tool use under varying data freshness constraints.

ACKNOWLEDGMENTS

The authors thank the Qwen team for releasing the Qwen model family and the developers of open-source tools used in the experiments. The authors gratefully acknowledge the Computer and Information Technology (CIT) program within the Johns Hopkins University Whiting School of Engineering, Engineering for Professionals, which encompasses Artificial Intelligence, Computer Science, Cybersecurity, Data Analytics, Data Science, and Information Systems Engineering. The support provided through this program was instrumental to this work.

REFERENCES

- [1] Dhingra, B., Cole, J. R., Eisenschlos, J. M., Gillick, D., Eisenstein, J., and Cohen, W. W., “Time-aware language models as temporal knowledge bases,” *Transactions of the Association for Computational Linguistics* **10**, 257–273 (2022).
- [2] Rajaby Faghihi, H. and Kordjamshidi, P., “Time-stamped language model: Teaching language models to understand the flow of events,” in [*Proc. North American Chapter of the Association for Computational Linguistics: Human Language Technologies*], 4560–4570 (2021).
- [3] Cao, S. and Wang, L., “Time-aware prompting for text generation,” in [*Findings of the Association for Computational Linguistics: EMNLP 2022*], 7231–7246 (2022).
- [4] Wang, H., Liang, Y., and Ren, H., “Temporal-aware soft prompt tuning for automatic text dating,” in [*Proc. 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*], 3975–3987 (2025).
- [5] Li, X. L. and Liang, P., “Prefix-tuning: Optimizing continuous prompts for generation,” in [*Proc. Annual Meeting of the Association for Computational Linguistics*], 4582–4597 (2021).
- [6] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I., “Attention is all you need,” in [*Advances in Neural Information Processing Systems*], (2017).
- [7] Kazemi, S. M., Goel, R., Eghbali, K., Ramanan, J., Sahota, J., Thakur, S., Wu, S., Smyth, C., Poupart, P., and Brubaker, M., “Time2vec: Learning a vector representation of time,” *arXiv:1907.05321* (2019).
- [8] Golkar, S., Pettee, M., Eickenberg, M., Bietti, A., Cranmer, M., Krawezik, G., Lanusse, F., McCabe, M., Ohana, R., Parker, L., Régaldo-Saint Blancard, B., Tesileanu, T., Cho, K., and Ho, S., “xval: A continuous numerical tokenization for scientific language models,” *arXiv:2310.02989* (2023).
- [9] Perez, E., Strub, F., de Vries, H., Dumoulin, V., and Courville, A., “Film: Visual reasoning with a general conditioning layer,” in [*Proc. AAAI Conf. Artificial Intelligence*], (2018).
- [10] Peebles, W. and Xie, S., “Scalable diffusion models with transformers,” in [*Proc. IEEE/CVF Int. Conf. Computer Vision*], (2023).
- [11] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W., “Lora: Low-rank adaptation of large language models,” in [*Proc. Int. Conf. Learning Representations*], (2022).
- [12] Liu, H., Tam, D., Muqeeth, M., Mohta, J., Huang, T., Bansal, M., and Raffel, C. A., “Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning,” in [*Advances in Neural Information Processing Systems*], (2022).
- [13] Gurnee, W. and Tegmark, M., “Language models represent space and time,” *arXiv:2310.02207* (2023).
- [14] Qwen, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., et al., “Qwen2.5 technical report,” *arXiv:2412.15115* (2025).