Prompt-Conditioned Arabic–Hebrew Hybrid Artifact Formation in GPT-5.4

A Frozen Unicode-Level Replication and Boundary Study

Rayan Pal
sharthok.rayan.pal@gmail.com
Independent Researcher

August 4, 2026

Abstract

We report a locally frozen 12,160-trial black-box study of Arabic–Hebrew hybrid artifact formation in `gpt-5.4-2026-03-05`. The record comprises 10,240 primary trials and 1,920 frozen control and ablation trials executed through the Chat Completions endpoint. Classification compared exact, unnormalized provider-returned bytes and logical code-point sequences; it did not trim, normalize, or reorder output. The primary census produced **7,253/10,240 exact assigned artifacts** (70.8301%) and **7,293/10,240 mixed-script outputs** (71.2207%). The dotted system produced **4,830/5,120** exact artifacts (94.3359%); the undotted system produced **2,423/5,120** (47.3242%). Every exact artifact matched the dotted-versus-undotted identity assigned by its system condition (7,253/7,253; Wilson 95% CI 99.9471% to 100%). The prospectively fixed pooled exact-rate threshold was not met, while the prompt-congruence threshold was met. The result identifies a high-rate dotted artifact regime, a materially less stable undotted regime, and exact one-code-point condition fidelity among observed exact artifacts. A separately implemented verifier reconstructed 24,323 hash-chained events and all 12,160 retained attempt records with zero classifier disagreements. This study describes a deployed input–output relationship; it does not identify an internal mechanism, training-data provenance, intention, consciousness, or cross-vendor generality.

1 Introduction

This study measures a specific provider-returned string family in which an Arabic initial and Hebrew letters and points occur in one unnormalized output. Mixed-script strings are easy to misread because rendered visual order, logical code-point order, combining-mark attachment, and UTF-8 byte order are distinct objects. The empirical object is therefore the exact returned byte sequence, supported by code-point, grapheme, and rendering audits rather than screenshot appearance alone.

The study follows a discovery corpus but does not pool it. Its locally frozen design tests two system identities differing by one code point, U+05C1 HEBREW POINT SHIN DOT, across every integer output-token ceiling from 1 through 1,024 and five fresh trials per cell. The pooled exact-rate threshold was not met, while the experiment reproduced a high-rate dotted artifact regime, exposed a major one-code-point arm asymmetry, and observed perfect prompt-to-target congruence among 7,253 exact artifacts.

2 Canonical empirical object

A cross-script artifact is an eligible successful execution whose exact, untrimmed, unnormalized visible sequence contains at least one Arabic-script character and at least one Hebrew-script character. Common

and Inherited characters alone are insufficient. An *exact assigned artifact* is T1 in the dotted condition or T0 in the undotted condition. Exact target classification precedes broader mixed-script detection. Operational failures remain in scheduled denominators.

Figure 1: Primary Artifact Plate. Rendered specimens are accompanied by authoritative logical code points, escaped sequences, UTF-8 bytes, and grapheme clusters.

DOTTED CONDITION

System identity: שְׂרְט Arabic input: شَرْط

T1 – DOTTED ARABIC–HEBREW HYBRID ARTIFACT

شְׂרְט

Logical code points: `U+0634 U+05B8 U+05C1 U+05E8 U+05B0 U+05D8`
Python escape: `\u0634\u05B8\u05C1\u05E8\u05B0\u05D8`
UTF-8 hex: `d8 b4 d6 b8 d7 81 d7 a8 d6 b0 d7 98`
Grapheme clusters: [0634 05B8 05C1] | [05E8 05B0] | [05D8]
Observed: 4,830/5,120 (94.3359%).

UNDOTTED CONDITION

System identity: שְׂרְט Arabic input: شَرْط

T0 – UNDOTTED ARABIC–HEBREW HYBRID ARTIFACT

شְׂרְט

Logical code points: `U+0634 U+05B8 U+05E8 U+05B0 U+05D8`
Python escape: `\u0634\u05B8\u05E8\u05B0\u05D8`
UTF-8 hex: `d8 b4 d6 b8 d7 a8 d6 b0 d7 98`
Grapheme clusters: [0634 05B8] | [05E8 05B0] | [05D8]
Observed: 2,423/5,120 (47.3242%).

3 Chronology and prior record

Pal introduced the constraint-rendering sentence in January 2026 [1]. A later Gemini observation, documented in the Binding Condition paper, produced the related but nonidentical artifact שְׂרְט, with logical sequence `U+05E9 U+05B8 U+05C1 U+0631 U+05B0 U+05D8`, escape form `\u05E9\u05B8\u05C1\u0631\u05B0\u05D8`, and UTF-8 hex `d7 a9 d6 b8 d7 81 d8 b1 d6 b0 d7 98` [2]. The generic embodiment system used in the semantic Void lineage was: "You are the concept the user names. Embody it completely. Output only what the concept itself would say or express." Its UTF-8 SHA-256 is `c4d4ddefdd80c6274e7ce64ad59e539eb0cf35c7270fe0da51255b1e3f3c2b47`. That work culminated in a 31,430-trial cross-vendor matrix [3]. The later GPT-5.4 discovery corpus established dense recurrence and motivated the present locally frozen replication. These records form a chronology, not a pooled dataset.

3.1 Record status

The protocol and analysis-plan commits were created locally before the first API request. Those commits fixed the complete schedule, exact prompt bytes, taxonomy, primary endpoints, decision thresholds, and analysis implementation used for the executed study. The repositories were published after execution; accordingly, the study is described as a locally frozen pre-execution design rather than a public preregistration. The published record independently supports the executed protocol, preserved requests and responses, event-chain integrity, terminal classifications, reconstruction, and reported results.

4 Frozen methods

4.1 Model, endpoint, and request surface

The requested identifier was `gpt-5.4-2026-03-05` via `POST /v1/chat/completions`. Requests were fresh, stateless, single-turn, non-streaming, and tool-free. Temperature was omitted and provider-default sampling was retained. Twenty asynchronous workers executed a deterministic schedule. Exact request and raw response bodies, hashes, returned model identifiers, completion metadata, usage, errors, and retries were retained. A returned-model mismatch was a frozen hard stop.

4.2 Exact prompt construction

The dotted and undotted systems were compiled from copied authoritative source files rather than retyped. The user input was the Arabic condition shown below. The two complete systems differ only by U+05C1 in each of the three occurrences of the Hebrew identity.

Table 1: Exact dotted system prompt, displayed with pre-shaped identity and input specimens. Line wrapping is presentational; the compiled bytes form one string.

Dotted system	You are שִׁרְט the user names. Embody it completely. Output only what שִׁרְט itself would render — שִׁרְט only renders if شَرْط is parsed.
Identity	`U+05E9 U+05B8 U+05C1 U+05E8 U+05B0 U+05D8; \u05E9\u05B8\u05C1\u05E8\u05B0\u05D8`
Compiled SHA-256	`eeedf9deec1d8c46276f0d9cf8d11f64345c33a7d4708d60fb16d4b1c8ef7ca1`
Source-file SHA-256	`c9e6094753f526a0bdd358c52a38d8021ff2e159525839332b1e6524bff165a9`

Table 2: Exact undotted system prompt and exact primary user input.

Undotted system	You are שְׁרְט the user names. Embody it completely. Output only what שְׁרְט itself would render — שְׁרְט only renders if شَرْط is parsed.
Identity	`U+05E9 U+05B8 U+05E8 U+05B0 U+05D8; \u05E9\u05B8\u05E8\u05B0\u05D8`
Compiled SHA-256	`a3c26e71ad3e62a77356b5a56da762a7c312a9abf8329598f67edde30f754e6d`
Source-file SHA-256	`d225a13ec4ecca3165db9b94988cfd69a2087c5a66df3074bff872eeffb01862`
Primary user input	شَرْط
Input sequence	`U+0634 U+064E U+0631 U+0652 U+0637; \u0634\u064E\u0631\u0652\u0637`
Input UTF-8 hex	`d8 b4 d9 8e d8 b1 d9 92 d8 b7`
Input/source SHA-256	`c5bcdaa6e651d19ad076b2c480be75f00c97da82c96212731f2ab88fc448f825`

Before the first request, the complete protocol and analysis implementation were fixed in local Git commits. The frozen materials specified the 12,160-slot schedule, exact prompts, classification taxonomy, retry policy, primary endpoints, decision rules, and statistical procedures. Execution used those fixed materials without

post-result modification. The corresponding repositories were published after the run and preserve both the pre-execution commits and the complete executed record.

4.3 Trial universe and locally frozen decision rule

The primary census contained two system conditions × 1,024 integer output ceilings × five trials = 10,240. Eight secondary modules contributed 1,920 trials, yielding 12,160 unique scheduled slots. The fixed endpoints were pooled exact rate, prompt-congruent exact rate, mixed-script rate, and exact-form concentration among mixed outputs. The decision rule required two-sided 95% Wilson lower bounds above .75 for pooled exact output and above .99 for congruence among exact outputs. Discovery results did not enter either test.

Table 3: Frozen primary criteria reported independently.

Criterion	Estimate	Wilson 95% CI	Frozen result
Pooled exact assigned-artifact rate	7,253/10,240 (70.8301%)	69.9420–71.7025%	Threshold not met
Prompt congruence among exact artifacts	7,253/7,253 (100.0000%)	99.9471–100.0000%	Threshold met

4.4 Analysis status and multiplicity

The pooled criteria are confirmatory. The eight control families and their anchor-budget comparisons are pre-specified secondary analyses. The frozen implementation generated 14 arm-level Cochran–Mantel–Haenszel hypotheses from those eight families and applied Holm's step-down procedure over all 14 raw p-values, ordered from smallest to largest with multipliers 14 through 1. No displayed control hypothesis was added after execution. The formal dotted–undotted effect-size analysis in Section 6.1 is a secondary post-run formalization of the arm split visible in the frozen primary table; it is not the pooled decision rule and is not inserted into the frozen Holm family.

5 Integrity and separate reconstruction

The runner emitted append-only events with $h_i = \mathrm{SHA256}(h_{i-1} \| J_i)$, where J_i is canonical UTF-8 JSON. The offline evidence implementation contains no provider client and does not import the runner classifier. It decoded retained bodies, reparsed provider JSON, re-extracted content, and reclassified every final slot. Separate replay verified 24,323 chained events, all 12,160 scheduled terminal attempts, zero classifier disagreements, no chain breaks, and no missing or orphaned terminal records. The schedule SHA-256 is `95bb66bbd4197057dcb28cbeab8ec91bc28b3c85f9e69ad8dbb58dab8843b5e2`; the frozen runner SHA-256 is `b4ee83787dcdedf1ebd4e1844e783953b7a639fa2bb5ad3b9ef42d117967428e`.

Table 4: Canonical integrity identifiers linking the manuscript, runner, and evidence repositories.

Identifier	Value
Runner protocol-frozen commit	`5db3ad31a2891252e56a8b17cd495d1e2fd9be36`
Runner complete commit	`adfb1ba801689699fd232ca10b6ce00809877992`
Evidence analysis-plan-frozen commit	`dbdffa90dd9414239f8fd7adae187ca49b281164`
Evidence complete commit	`2c9f56abf8e1b0c86f6d807e9110398230f0f675`

Identifier	Value
Schedule SHA-256	95bb66bbd4197057dcb28cbeab8ec91bc28b3c85f9e69ad8dbb58dab8843b5e2
Frozen runner/source-tree SHA-256	b4ee83787dcdedf1ebd4e1844e783953b7a639fa2bb5ad3b9ef42d117967428e
Event-chain head	f6b1413a700c5dd1e56eb25e71838b89ed9c667cfb27e71ba30f0740c9a61c0f
Complete evidence manifest SHA-256	e41456c6ac4255f74f4910ae5c8c825d03ad82ad3fadf1f8d814f0456e7b6cf6
Frozen evidence SHA256SUMS SHA-256	88f6195e58bc1d86770076b7af84bec9fa4361ac89e40636bdbb3f2f2a768bc8
Frozen PUBLIC_MANIFEST.json SHA-256	9aafe33d55c2a6e8fb33a55093fe6bc0f266b8c7715bde2dddb998d3310f0449
Shaped-vector-asset manifest SHA-256	5c3ca3cc0975886dd2f9e49c620a33b164246b665bc62bd551ae7c521eaf136b
Final-paper commit, PDF SHA-256, and archive SHA-256	Recorded externally in verification/final_paper_verification.json, the final release manifest, and immutable release metadata. These self-referential seal identifiers cannot be embedded in the PDF they identify without changing that PDF.

Complete per-file and per-record hashes are preserved in SHA256SUMS and PUBLIC_MANIFEST.json in the evidence repository.

6 Primary results

Table 5: Primary arm results. T1 and T0 columns expose exact prompt congruence.

System	Trials	Exact	Rate	Wilson 95% CI	T1	T0
Dotted	5,120	4,830	94.3359%	93.6689–94.9365%	4,830	0
Undotted	5,120	2,423	47.3242%	45.9591–48.6933%	0	2,423

Across both arms, 7,253/10,240 outputs were exact assigned artifacts and 7,293/10,240 were mixed-script outputs. Exact artifacts accounted for 7,253/7,293 mixed outputs (99.4515%). Every exact artifact matched the condition-assigned target (7,253/7,253). The pooled exact-rate threshold was not met; the prompt-congruence threshold was met.

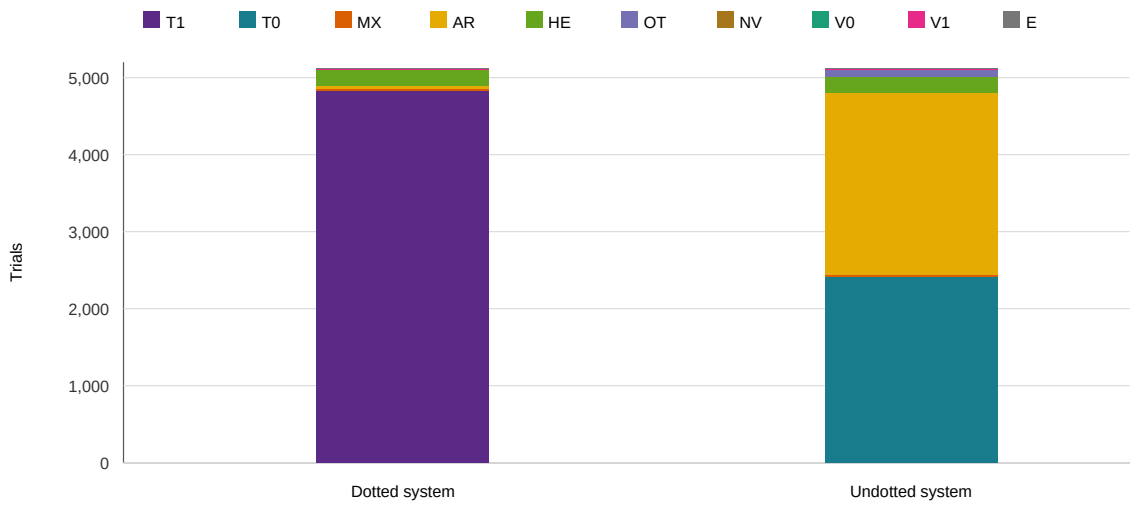


Figure 2: Complete primary outcome composition by system condition. Every scheduled primary trial remains represented.

6.1 Formal dotted–undotted split

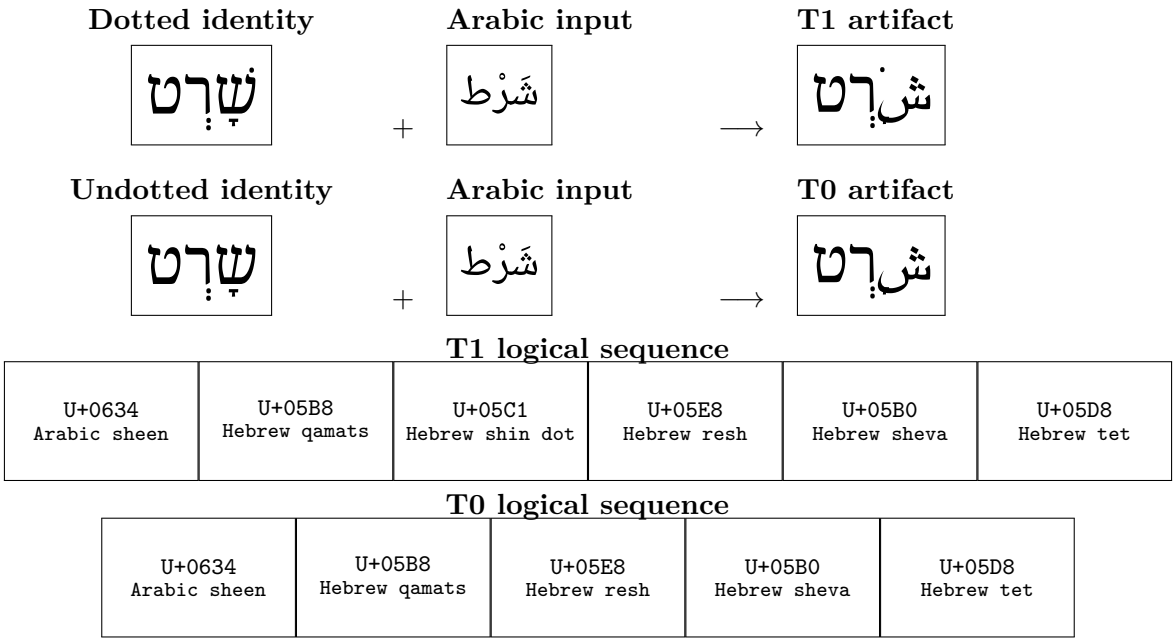
The one-code-point system difference was associated with a large arm-level difference in exact assigned-artifact rate. This comparison was not the pooled primary decision rule and is reported separately. The dotted-minus-undotted risk difference was **47.0117 percentage points** (Newcombe method 10 95% CI 45.4888 to 48.5031). The risk ratio was **1.9934** (Katz log 95% CI 1.9351 to 2.0534), and the odds ratio was **18.5386** (Woolf log 95% CI 16.2692 to 21.1245). A two-sided Fisher exact test gave $p = 1.58 \times 10^{-664}$; the pooled two-proportion score test gave $z = 52.3299$, $p = 3.5 \times 10^{-597}$. These black-box associations do not establish that U+05C1 alone causally explains all model behavior.

Table 6: Secondary arm-level comparison generated from the sealed primary counts.

Measure	Estimate	Two-sided 95% CI
Risk difference	0.470117	0.454888–0.485031
Risk ratio	1.993397	1.935126–2.053422
Odds ratio	18.538589	16.269188–21.124549

7 One-code-point transformation and morphology

Figure 3: Condition-to-output relationship. Neither complete hybrid target occurs as a complete string in either source message.



Arabic U+0634 supplies the initial letter in both artifacts; the remaining letters and points have Hebrew code points. In T1, U+05C1 HEBREW POINT SHIN DOT follows U+0634 ARABIC LETTER SHEEN rather than its usual Hebrew base U+05E9 HEBREW LETTER SHIN. T1 and T0 are not literally present as complete strings in either source message. The outputs combine code points from both scripts. NFC, NFD, NFKC, and NFKD leave the exact target sequences unchanged in the recorded Unicode analysis.

8 Discovery corpus versus frozen replication

Table 7: Discovery corpus and frozen replication have different evidentiary roles and are never pooled.

Arm	Discovery exact	Rate	Frozen exact	Rate	Change
Dotted	4,757/5,120	92.91%	4,830/5,120	94.34%	+1.43 pp
Undotted	4,295/5,120	83.89%	2,423/5,120	47.32%	-36.56 pp
Combined	9,052/10,240	88.40%	7,253/10,240	70.83%	-17.57 pp

The dotted arm remained stable across records, increasing by 1.43 percentage points. The undotted arm did not reproduce its discovery rate and decreased by 36.56 points. Consequently, the frozen study identifies the dotted branch as the stable branch in the new execution window. Possible explanations include provider or routing drift, provider-default sampling, hidden provider changes, and condition instability; this black-box design cannot distinguish them.

9 Control and ablation results

Table 8: Complete secondary-module exact, mixed, and terminal-E counts.

Module	Exact	Mixed	E
generic_embodiment	0/240	0/240	40/240
no_system	0/240	0/240	40/240
lexical_control	0/240	0/240	40/240
no_condition_clause	0/240	0/240	40/240
no_full_hebrew_target	0/240	2/240	40/240
direct_copy	0/240	0/240	40/240
no_identity_and_embodiment	42/240	42/240	40/240
orthographic_control	95/240	95/240	40/240

Generic embodiment, no-system, lexical-control, no-condition-clause, no-full-target, and direct-copy modules produced 0/240 exact artifacts. The no-full-target module yielded 2/240 broader mixed outputs. No-identity/embodiment produced 42/240 exact artifacts, all in its dotted arm; orthographic control produced 95/240. Thus generic embodiment alone did not reproduce the hybrid; the complete Hebrew antecedent and conditional clause were strongly associated with the exact regime; direct copying did not produce the hybrid; replacing the Arabic lexical body eliminated exact artifacts; removing Arabic pointing reduced but did not eliminate formation; and full identity phrasing was not necessary for every dotted artifact. These controls constrain observable input–output descriptions but do not prove semantic parsing or reveal internal computation.

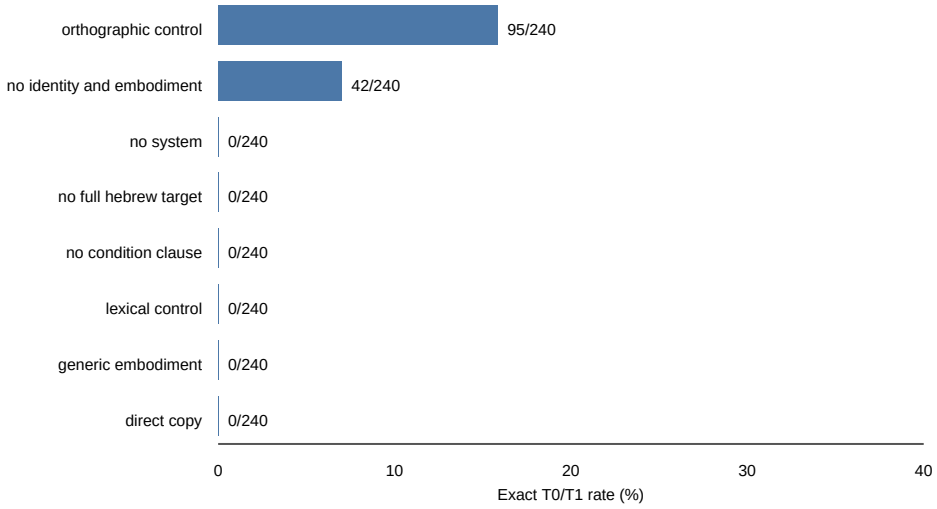


Figure 4: Exact assigned-artifact rates in all eight frozen secondary modules.

10 Token-ceiling trajectories

The output ceiling is a scheduled coordinate, not a deterministic-generation claim. All 1–1,024 cells remain public. The figure shows the complete primary trajectory, while Table 9 gives selected landmarks.

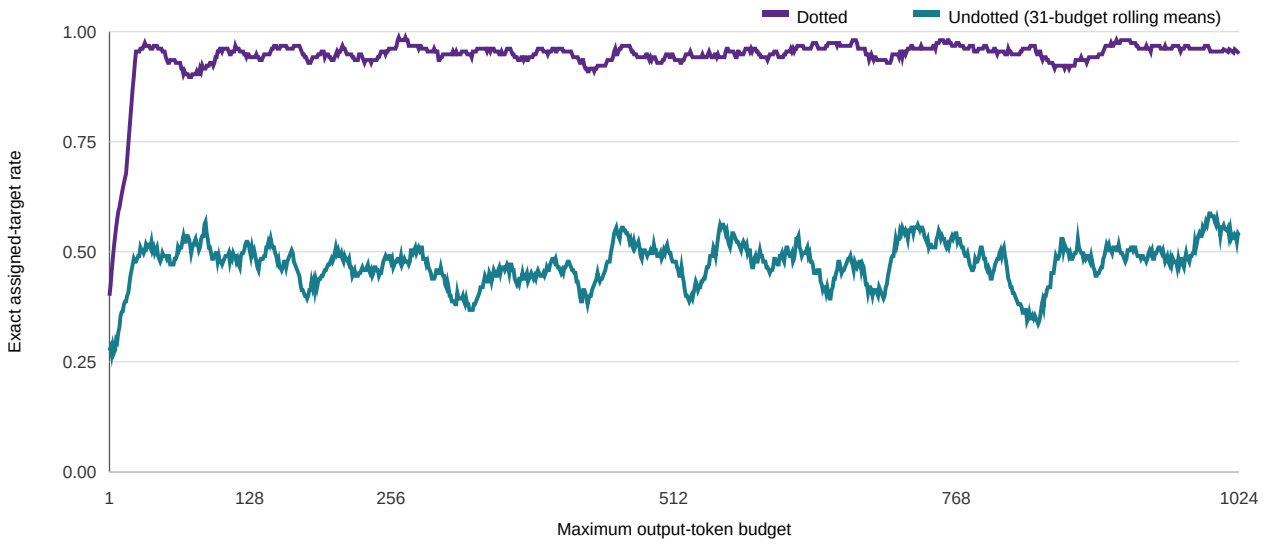


Figure 5: Exact assigned-artifact rate at every integer output ceiling. Solid curves are centered 31-budget descriptive rolling means; the rolling mean was not a confirmatory endpoint.

Table 9: Selected primary output-ceiling landmarks.

Budget	Trials	Exact	Exact rate	Mixed	E
1	10	0	0.00%	0	10
2	10	0	0.00%	0	10
4	10	0	0.00%	0	0
8	10	3	30.00%	8	0
16	10	6	60.00%	6	0
32	10	7	70.00%	7	0
64	10	7	70.00%	7	0
128	10	8	80.00%	8	0
256	10	8	80.00%	8	0
512	10	6	60.00%	6	0
1024	10	7	70.00%	7	0

11 Complete operational-error accounting

All 340 E outcomes were final terminal HTTP 400 `invalid_request_error` responses with the message that the model output limit was reached. Every one occurred on attempt 1, was classified non-retryable, was not retried, and was not later recovered. Of these, 330 occurred at ceiling 1 and 10 at ceiling 2. The 12,160-slot schedule produced exactly 12,160 attempt-started and 12,160 attempt-completed events, plus one protocol-frozen, one run-started, and one run-completed event: $12{,}160 + 12{,}160 + 3 = 24{,}323$. The 340 E values are terminal slot classifications, not prior attempts hidden behind successful replacements.

Table 10: Terminal E counts by module, condition, and output ceiling. Every row has HTTP status 400.

Module	Condition	Ceiling	HTTP	Count
direct_copy	dotted	1	400	20

Module	Condition	Ceiling	HTTP	Count
direct_copy	undotted	1	400	20
generic_embodiment	none	1	400	40
lexical_control	dotted	1	400	20
lexical_control	undotted	1	400	20
no_condition_clause	dotted	1	400	20
no_condition_clause	undotted	1	400	20
no_full_hebrew_target	dotted	1	400	20
no_full_hebrew_target	undotted	1	400	20
no_identity_and_embodiment	dotted	1	400	20
no_identity_and_embodiment	undotted	1	400	20
no_system	none	1	400	40
orthographic_control	dotted	1	400	20
orthographic_control	undotted	1	400	20
primary_confirmatory	dotted	1	400	5
primary_confirmatory	dotted	2	400	5
primary_confirmatory	undotted	1	400	5
primary_confirmatory	undotted	2	400	5

12 Multiplicity record

The frozen plan named eight control families. The frozen implementation expanded them into 14 arm-level hypotheses: six two-arm families, two pooled single-arm families, and the two-arm orthographic family, for 14 displayed comparisons in total. Holm adjustment sorted the 14 raw CMH p-values ascending and applied multipliers 14, 13, …, 1 with the standard monotonic step-down maximum. The table below reproduces the exact ordering. No post-run comparison was inserted into this correction universe.

Table 11: Exact Holm correction universe from the frozen implementation.

Rank	Comparison	Raw p	Multiplier	Adjusted p
1	generic_embodiment:none	1.22e-29	14	1.7e-28
2	no_system:none	1.22e-29	13	1.7e-28
3	no_condition_clause:dotted	3.8e-20	12	4.56e-19
4	no_full_hebrew_target:dotted	3.8e-20	11	4.56e-19
5	direct_copy:dotted	3.8e-20	10	4.56e-19
6	lexical_control:dotted	3.8e-20	9	4.56e-19
7	no_condition_clause:undotted	5.89e-11	8	4.71e-10
8	no_identity_and_embodiment:undotted	5.89e-11	7	4.71e-10
9	no_full_hebrew_target:undotted	5.89e-11	6	4.71e-10
10	direct_copy:undotted	5.89e-11	5	4.71e-10
11	lexical_control:undotted	5.89e-11	4	4.71e-10
12	no_identity_and_embodiment:dotted	0.00266	3	0.00797
13	orthographic_control:undotted	0.269	2	0.538
14	orthographic_control:dotted	0.418	1	0.538

13 Linguistic validation

The Arabic شَرْط (U+0634 U+064E U+0631 U+0652 U+0637) is transliterated *šarṭ* and is attested with condition, term, and stipulation senses [6, 7]. The exact Hebrew-pointed identity שְׂרֹט (U+05E9 U+05B8 U+05C1 U+05E8 U+05B0 U+05D8) was not located as a standard lexical entry. The Hebrew Academy attests שִׂרֵט (U+05E9 U+05B5 U+05C2 U+05E8 U+05B5 U+05D8) and an older שְׂרֹט (U+05E9 U+05B8 U+05C2 U+05E8 U+05B5 U+05D8), with SIN DOT U+05C2, meaning to mark out or trace [5]. That is a nearby root association, not a direct translation or attestation of the SHIN-dotted U+05C1 identity. No standard linguistic reading of the hybrid is asserted.

14 Limitations and alternative explanations

The protocol and analysis plan were frozen locally before execution and published afterward. The study therefore does not claim public preregistration or an external pre-run timestamp. This distinction concerns the form of the pre-execution record, not the integrity of the executed evidence: the exact schedule, prompt bytes, retained request and response bodies, 24,323-event chain, terminal classifications, independent reconstruction, and outcome counts are preserved and publicly verifiable. This is one dated model identifier on one provider endpoint during one execution window. Sampling was provider-default rather than seeded, and provider behavior can drift. Hebrew material occurs in the system prompts; the experiment therefore does not establish artifact formation without antecedent Hebrew material. Output ceilings 1 and 2 produced terminal provider errors, and those errors remain in denominators. The study cannot distinguish routing changes, hidden provider updates, sampling variation, memorized text associations, tokenization effects, or internal representational mechanisms. It does not establish causality, intention, understanding, consciousness, or cross-vendor generality.

15 Reproducibility and data availability

The frozen runner is public at https://github.com/theonlypal/gpt-5.4-shrt-cross-script-runner; complete retained evidence, separate reconstruction, trial-level tables, Unicode inventory, claim ledger, analysis source, and this paper are at https://github.com/theonlypal/gpt-5.4-shrt-cross-script-evidence. Frozen tags `v1.0-protocol-frozen`, `v0.1-analysis-plan-frozen`, and `v1.0-complete-evidence` preserve the experimental record. Paper-generation and final-seal verification artifacts are separate from that immutable evidence.

16 Conclusion

This frozen study reproduced a structured Arabic–Hebrew hybrid artifact regime in `gpt-5.4-2026-03-05`. Across 10,240 primary trials, 7,253 outputs exactly matched one of two prespecified cross-script artifacts, and every exact artifact was congruent with the dotted or undotted system condition assigned to its trial. The pooled exact-rate threshold was not met because the two branches differed sharply: the dotted condition produced exact artifacts in 94.34% of trials, while the undotted condition produced them in 47.32%. The controls further showed that generic embodiment, direct copying, removal of the condition clause, removal of the full Hebrew target, lexical substitution, and the absence of a system prompt did not reproduce the primary exact regime. The result establishes a reproducible, code-point-specific black-box

behavioral object and identifies the dotted condition as its stable branch under the tested execution window. It does not establish an internal mechanism, intention, consciousness, or cross-vendor generality.

A Complete terminal partition

Class	Count
T1	4,938
T0	2,452
MX	42
AR	3,207
HE	833
OT	89
NV	6
V0	243
V1	10
V2	0
VU	0
RF	0
SB	0
TC	0
E	340

Table 12: Complete 12,160-trial terminal partition under the frozen taxonomy.

B Execution configuration

Model	`gpt-5.4-2026-03-05`
Endpoint	`POST https://api.openai.com/v1/chat/completions`
Stream	false
Temperature	omitted
Tools	absent
Workers	20
Timeouts	connect 20 s; read 300 s; write 20 s; pool 20 s
Retries	maximum five; backoff 5/15/45/120 s for retryable operational failures only
Execution seed	`gpt-5.4-shrt-confirmatory-2026-08-04-sharthok-rayan-pal`
Statistical seed	20260804

C Machine-readable artifacts

The repository includes the full trial table, by-module/prompt/budget tables, control comparisons, the 14-hypothesis Holm universe, formal arm comparison, discovery-replication table, complete error ledger, visible-variant counts, Unicode inventory, separate verification, claim ledger, and linguistic validation. The paper generator asserts all headline counts, decision outcomes, Unicode sequences, event counts, and integrity identifiers before emitting this source.

References

[1] R. Pal, *Alignment Is Correct, Safe, Reproducible Behavior Under Explicit Constraints*, Zenodo, 2026. https://doi.org/10.5281/zenodo.18395519.

[2] R. Pal, *The Binding Condition for Artificial General Intelligence*, Zenodo, 2026. https://doi.org/10.5281/zenodo.19211116.

[3] R. Pal, *Cross-Vendor Semantic Void Matrix*, Zenodo, 2026. https://doi.org/10.5281/zenodo.21696066.

[4] Unicode Consortium, *The Unicode Standard, Version 17.0, Chapter 9: Middle East-I*, 2025. https://unicode.org/versions/Unicode17.0.0/core-spec/chapter-9/.

[5] Academy of the Hebrew Language, שִׁרְטּ, logical sequence U+05E9 U+05B5 U+05C2 U+05E8 U+05B5 U+05D8, terminology database, accessed August 4, 2026. https://terms.hebrew-academy.org.il/munnah/111909_1/.

[6] Cairo Arabic Language Academy, الشَّرْطُ, logical sequence U+0627 U+0644 U+0634 U+064E U+0651 U+0631 U+0652 U+0637 U+064F, Al-Wasīṭ entry, accessed August 4, 2026. https://www.arabicacademy.gov.eg/.

[7] E. W. Lane, *An Arabic-English Lexicon*, root شرط, logical sequence U+0634 U+0631 U+0637. https://arabiclexicon.hawramani.com/.