CHIPREASONING: BENCHMARKING EVIDENCE DISCIPLINE IN AGENTIC CHIP IMPLEMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Chip-implementation decisions are made from tool reports. An agent must choose which reports to inspect and restrict its conclusions to the evidence it has opened. We introduce CHIPREASONING, a benchmark of **170 cases arranged into 85 counterfactual pairs** across **six chip-engineering domains**. Each case offers six labeled but hidden report packets under a fixed inspection budget. Pair members share the request and visible selectors, while one decision-critical packet changes. A seven-component Process score evaluates both the inspection trajectory and the submitted engineering disposition. Across 11 model configurations, Process scores range from 41.63 to 70.53, and **10 of 11 exceed the strongest deterministic content-blind baseline** of 50.41. The leading configurations inspect the changed packet in both members of at least 96% of pairs; the best standard-interface both-correct pair rate is 45.5%. The dominant errors occur after evidence access, when retrieved measurements must be converted into the correct engineering commitment.

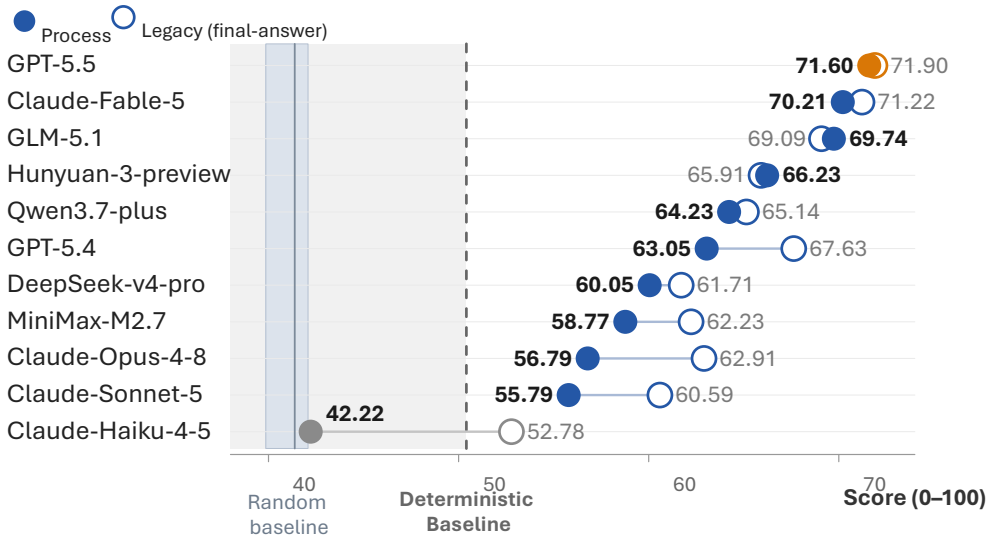

Figure 1: Process (filled, with run SD) and final-answer (open) scores. The gray band spans the random-policy mean to maximum (39.86–42.08), with its 95th percentile at 41.38. The dashed line marks the strongest deterministic content-blind baseline (50.41); orange marks GPT-5.5.

1 INTRODUCTION

Chip implementation produces reports across synthesis, placement, routing, timing, and power analysis (Ajayi et al., 2019). Answering an engineering request requires choosing relevant reports, inspecting them, and grounding the final disposition in their contents. The inspection record shows which evidence was available when the answer was submitted.

General agent benchmarks evaluate tool use and software repair (Mialon et al., 2024; Jimenez et al., 2024), and process-oriented work evaluates more than the final answer (Lightman et al., 2024).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

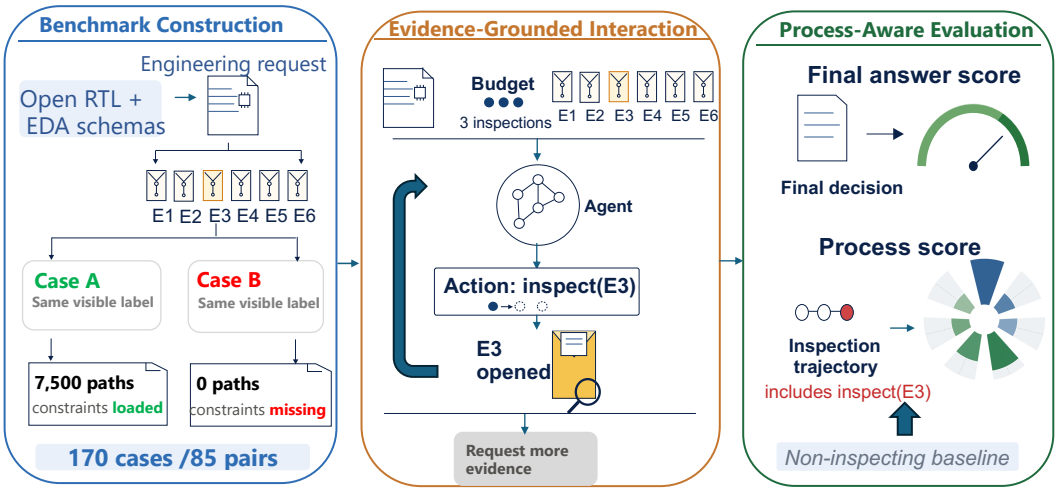

Figure 2: Overview of ChipReasoning. **(a)** We construct 170 cases in 85 counterfactual pairs across six domains. Cases in a pair share the request and selectors; only one decision-critical packet—the *carrier*—changes. **(b)** Selectors are retrieval handles; packet facts remain hidden until inspection. At each interaction step, the agent selects a packet, receives its contents, and updates its next action using the accumulated evidence. The agent either inspects again or terminates with a final disposition. The illustrated example uses $K = 3$; benchmark budgets range from three to five. The trajectory follows Case B: `inspect(E3)` reveals zero analyzed paths and missing constraints, after which the agent requests more evidence. **(c)** Final-answer evaluation scores the submitted response alone, whereas process-aware evaluation additionally scores the executed inspection trajectory. The content-blind baseline issues inspections but discards every returned packet.

Hardware benchmarks cover RTL generation, engineering assistance, EDA execution, report comprehension, root-cause analysis, and physical-design flows (Liu et al., 2023b;a; Wu et al., 2024; Li et al., 2026). These evaluations combine report-retrieval failures with report-interpretation failures.

ChipReasoning records report access. Each case presents an engineering request and six labeled packets whose contents remain hidden until inspection. The agent has a limited budget and must eventually choose one of five engineering dispositions. The action log records every opened packet, separating report-selection errors from errors in the final judgment.

The cases are arranged into 85 counterfactual pairs. Two members of a pair have the same request, selectors, budget, and five identical packets. The sixth packet crosses an engineering decision boundary, changing the preferred disposition. A content-blind policy sees no difference between the two cases. Our Process score combines required-evidence coverage, inspection efficiency, budget discipline, quantitative checks, decision calibration, invalid-inference control, and engineering closure. A separate final-answer score ignores the trajectory.

Figure 1 gives the overall ranking. Ten configurations beat the 50.41 content-blind baseline, with Process scores between 41.63 and 70.53. The maximum Process–Legacy gap is 10.20 points, produced when the final response cites evidence absent from the inspection record. The strongest systems usually open the changed packet; the best both-correct rate is 45.5%. The dominant error occurs when retrieved evidence is converted into the final disposition.

The paper contributes:

- a benchmark of 170 evidence-gated chip implementation decisions, organized into counterfactual pairs and grounded in open RTL designs and EDA report schemas;
- a deterministic Process score and content-blind baselines that distinguish evidence-grounded decisions from label matching; and
- an evaluation of 11 model configurations that measures report retrieval separately from evidence-conditioned engineering decisions.

Table 1: A worked counterfactual pair. The request, report selectors, synthesis result, and lint result are identical; only the hidden timing packet changes. The pair tests whether a model distinguishes a valid timing result with small positive slack from an invalid run in which no paths were analyzed.

	Case A	**Case B**
Review request	Can the UART design pass the post-synthesis review and proceed to physical implementation?	
Shared evidence	The synthesis structure matches expectation, lint reports no errors, and the six report selectors are identical.	
Changed packet	E3: OpenSTA timing report	
Constraints	Loaded	Missing
Paths analyzed	7,500	0
Worst slack	$+0.005$ ns	Not valid
Interpretation	Timing was evaluated with a positive setup margin of 0.005 ns.	No timing paths were evaluated; the reported zero slack is invalid as a timing-closure result.
Disposition	**Conditional approve**	**Request more evidence**
Next action	Continue to implementation and complete the physical checks.	Load the missing constraints and rerun static timing analysis.

2 Related Work

LLMs for chip design. Chip-design benchmarks cover RTL generation, executable correctness, engineering assistance, EDA operation, and physical-design workflows (Liu et al., 2023b; Lu et al., 2024b; Thakur et al., 2023a; Liu et al., 2023a; Wu et al., 2024; Li et al., 2026). Domain-adapted models, compiler-guided revision, simulation feedback, and hierarchical debugging extend this line (Liu et al., 2025a; Cui et al., 2024; Liu et al., 2025b; Thakur et al., 2023b; Tsai et al., 2023; Yu et al., 2026). ChipReasoning evaluates the report selection and evidence-to-decision stage that precedes artifact revision.

Agent and process evaluation. Agent benchmarks study software repair, web interaction, tool use, and multi-step state transitions (Jimenez et al., 2024; Liu et al., 2024; Zhou et al., 2024; Mialon et al., 2024; Yao et al., 2025; Zhuang et al., 2023; Yao et al., 2023; Schick et al., 2023; Yang et al., 2024; Xie et al., 2024; Ma et al., 2024; Lu et al., 2024a; Trivedi et al., 2024). Process supervision, calibration, selective prediction, and citation grounding evaluate intermediate correctness and evidential support (Uesato et al., 2022; Lightman et al., 2024; Guo et al., 2017; Kadavath et al., 2022; Lin et al., 2022; Kamath et al., 2020; Gao et al., 2023). T-Eval decomposes tool utilization and RGB tests responses to changed retrieved evidence (Chen et al., 2024b;a). Our benchmark combines observable report access, single-carrier counterfactuals, and an engineering commitment rubric.

Shortcut controls. Partial-input baselines reveal annotation shortcuts (Gururangan et al., 2018; Poliak et al., 2018), and behavioral tests localize errors hidden by aggregate accuracy (Ribeiro et al., 2020). Content-blind policies quantify selector-label shortcuts. Paired cases isolate the response to one changed engineering fact.

3 Evidence-Grounded Chip Reasoning

Each case contains a visible engineering request, six labeled but hidden evidence packets, an inspection budget, and a structured action interface. The model chooses packets to inspect and then submits a disposition based on what it has seen (Figure 2). Both parts of the trajectory are scored.

3.1 Task Formulation

A case is $(q, \mathcal{P}, K)$, where q is the engineering request, $\mathcal{P}$ contains six evidence packets, and $K \in \{3, 4, 5\}$ is the inspection budget. At the start, the model sees q, K, and a selector for each packet. A selector names the artifact type and design, such as an STA report for a UART block, but reveals

no report facts. The model uses `inspect` to reveal one packet and `final` to submit a disposition and supporting fields.

Evidence acquisition. The budget is smaller than the six-packet collection, making packet selection necessary. Each case has an annotated required set for the preferred disposition (1.62 packets on average), and coverage is measured against that set. Optional packets provide corroborating evidence. "Required" denotes the benchmark's annotated evidence path. The action log determines which facts were available at submission time.

Decision calibration. The disposition must match the strength of the inspected evidence. A failed measurement supports rejecting an action; root-cause attribution requires additional evidence. Missing constraints make a timing result unusable, while a timing-failure claim requires an analyzed and constrained run. The final record includes a disposition, derived quantities and source packets, missing-evidence and inference checks, and a next action. Scoring uses these fields and the recorded trajectory.

Decision space. The benchmark defines five target dispositions:

- **Approve:** the inspected evidence supports the proposed claim or action;
- **Conditional approve:** the proposal proceeds under an explicit, bounded condition;
- **Request more evidence:** a decision-critical report or measurement is missing or inconclusive;
- **Reject:** the inspected evidence is sufficient to reject the proposed claim or action; and
- **Reject root-cause attribution:** the evidence establishes a failure and leaves the proposed cause unsupported.

The response schema also retains the auxiliary action `accept_root_cause`. A supported root-cause claim maps to **approve**; the scorer assigns the auxiliary action a calibration error.

3.2 Benchmark Design Rationale

Report choice as an action. Selectors expose the artifact type and design; report contents remain hidden. This is the information available when an engineer decides which report to open. Since every budget is smaller than the six-packet set, report choice affects the outcome. An opened packet remains available for the rest of the case. A wrong disposition after required-packet inspection is a post-retrieval error.

Commitment strength. The five dispositions preserve distinctions hidden by a binary pass/fail label. A violated check rules out a proposed action while leaving the proposed explanation unsupported, and an unconstrained timing run provides no timing decision. We score the disposition along with reported values, source packets, inference checks, and the proposed next action. Credit follows this evidence record.

Pairwise control. Topic priors and a conservative default solve some individual cases. Pair members remove that shortcut by holding the request, selectors, budget, and five packets constant. Only the carrier supports the change in preferred disposition. The pair metrics measure whether the same configuration reaches that packet and changes its answer accordingly, jointly measuring acquisition and response.

3.3 Benchmark Construction

Engineering scenarios and evidence packets. The benchmark uses four open RTL designs—`secworks-aes`, `verilog-uart`, `picorv32`, and `wbuart32`—and packet schemas based on open-source EDA reports and documentation (Secworks, 2026; Forencich, 2026; Wolf, 2026; Gisselquist, 2026; Wolf and Glaser, 2013; The OpenROAD Project and Parallax Software, 2026). Each case has one packet for each of six artifact classes: synthesis, static timing, lint, simulation, formal verification, and regression. Packets contain structured report fields. For example, a timing packet records setup and hold slack, total negative slack, analyzed-path count, and constraint status. The

Table 2: Core diagnostic questions and their primary observable signals. The supplementary appendix gives the detailed construction record.

Core question	Primary signal
Needed evidence found?	Required-packet coverage and inspection efficiency
Decision changed with the carrier?	Carrier acquisition, both-correct pairs, and unchanged-decision rate
Final commitment supported?	Strict decision accuracy and Process calibration components
Beyond label and protocol cues?	Margin over the 50.41 content-blind baseline

cases cover physical readiness, verification, test, design, implementation debugging, and power; their distribution appears in the appendix. Decision-critical values vary across cases, with fixed field names, units, and report structure. The packets are standardized records derived from open report schemas.

Case annotation. Each annotation contains a preferred disposition, any acceptable alternatives, required/optional/distractor packet labels, derived values and their sources, invalid inferences, engineering concepts, and a next action. The author team manually reviewed every retained case and its complete annotation record after generation. Review covered the engineering request, all six packet records, selector labels, inspection budget, preferred and acceptable dispositions, required/optional/distractor roles, sourced quantities, invalid-inference entries, engineering concepts, and next action. Pair-level review checked the shared visible inputs, the five unchanged packets, the carrier change, and the corresponding disposition boundary. These fields define the deterministic rubric. Strict Decision Accuracy counts only the preferred disposition; Acceptable Accuracy also counts annotated alternatives.

Counterfactual construction. The 170 cases form 85 counterfactual pairs. Within a pair, the request, selectors, budget, and five packet contents are identical. The remaining packet, the **carrier**, crosses an engineering decision boundary and changes the preferred disposition (Gardner et al., 2020). In Table 1, one timing packet reports loaded constraints, 7,500 analyzed paths, and positive setup slack. The paired packet reports missing constraints and zero analyzed paths. The first case supports conditional approval; the second requires more evidence. Systematic pair success requires using the carrier. Independently sampled content-blind decisions solve individual pairs by chance; Section 4.2 quantifies that stochastic null.

Quality control and dataset statistics. The structural validator checks paired visible inputs, the single-packet change, selector validity, budget reachability, and annotation completeness. A boundary audit removes eight simulation pairs, and an ambiguity audit removes seven more. The final set contains 170 cases in 85 complete pairs. Table 2 condenses the benchmark's central diagnostic questions. The supplementary appendix gives the complete construction record and additional validation analyses.

Reading the decomposition. The first three rows locate an error in retrieval, response to the carrier, or the final commitment. The last row measures how much of the score is available without reading packet contents.

3.4 Evaluation Metrics

Process score. Process is computed from observable actions and structured outputs. Let R, E, B, D, Q, I, and C denote normalized scores for required-evidence coverage, inspection efficiency, budget discipline, decision calibration, quantitative checks, invalid-inference control, and engineering closure. The per-case score is

$$S_{\text{process}} = 25R + 10E + 10B + 20D + 15Q + 10I + 10C. \quad (1)$$

Required-evidence coverage is the fraction of required packets inspected. Efficiency is based on required and optional packets among distinct inspections; the budget term penalizes distractors, overruns, invalid actions, and repeats. Missing required evidence caps decision credit, most strongly for

Table 3: Results on 170 cases. Each configuration has three completed runs; Process is mean ± sample SD and all other metrics are run means. Δ is Process minus Legacy. Decision accuracy is preferred-disposition accuracy; Full required is complete required-packet coverage. Bold marks the best comparable non-reference result. †GPT-5.5 is the generator-involved reference.

Model	Process ± run SD ↑	Legacy ↑	Δ	Dec. Acc. ↑	Full Req. ↑
GPT-5.5†	70.53 ± 1.50	73.80	−3.27	60.4	94.7
GLM-5.1	**70.19 ± 0.65**	67.78	+2.41	**69.4**	88.0
Claude-Fable-5	69.28 ± 1.75	**71.65**	−2.37	67.5	84.1
Hunyuan-3-preview	65.99 ± 0.19	65.09	+0.90	64.5	**98.6**
Qwen3.7-plus	64.05 ± 1.99	63.37	+0.68	66.7	81.6
GPT-5.4	63.54 ± 1.40	66.84	−3.30	54.5	76.9
MiniMax-M2.7	60.16 ± 1.14	63.98	−3.82	54.1	81.4
DeepSeek-v4-pro	59.23 ± 0.32	63.23	−4.00	57.6	96.5
Claude-Opus-4-8	56.89 ± 0.03	61.76	−4.87	56.9	61.0
Claude-Sonnet-5	55.07 ± 0.52	59.79	−4.72	52.9	71.4
Claude-Haiku-4-5	41.63 ± 1.20	51.83	−10.20	42.5	44.9

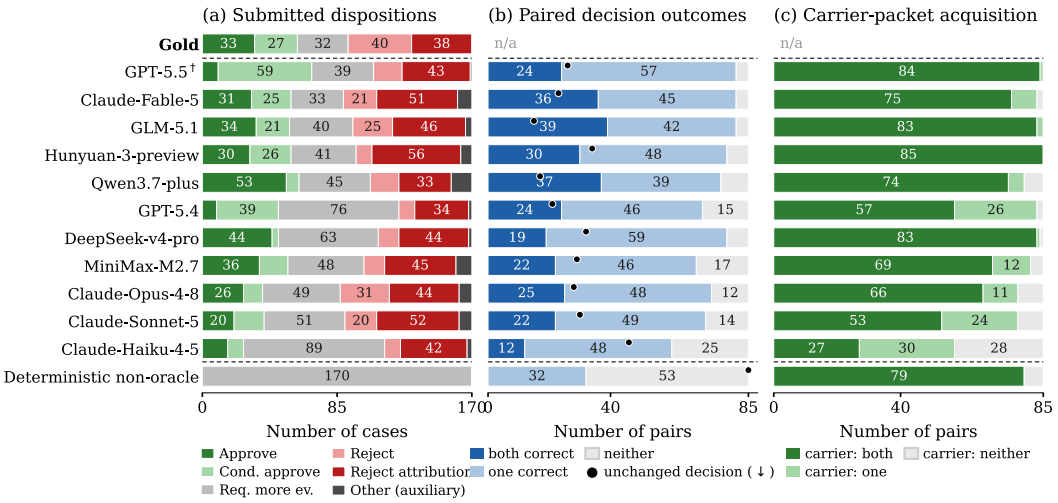

Figure 3: Counterfactual diagnostics averaged over three completed runs; content-blind nulls average 1,600 simulations. **(a)** Pairs with both, one, or neither preferred decision correct; dots mark unchanged decisions. **(b)** Carrier inspection in both, one, or neither case. **(c)** Both-correct rate conditional on inspecting the carrier in both cases. Null policies use visible inputs but discard packet contents.

approval. Quantitative credit requires both a compatible value and its source packet. The remaining terms check unsupported inferences, required engineering concepts, and a useful next action. If a component is inapplicable, the active weights are renormalized to 100.

Final-answer score. Legacy scores the final response on the same 0–100 scale without consulting the action history. It combines disposition, cited evidence, values, concepts, inference checks, next action, and output format. Strict Decision Accuracy is an exact match to the preferred disposition; Acceptable Accuracy in the appendix also counts annotated alternatives. We use Process as the primary metric and report the Process–Legacy difference to show when the inspection record changes the assessment.

4 Experimental Setup

4.1 Models and Protocol

The evaluation includes 11 model configurations: GPT-5.5, Claude-Fable-5, GLM-5.1, Hunyuan-3-preview, Qwen3.7-plus, GPT-5.4, DeepSeek-v4-pro, MiniMax-M2.7, Claude-Opus-4-8, Claude-Sonnet-5, and Claude-Haiku-4-5. Each configuration is evaluated in three completed runs on the same frozen set of 170 cases. Reported metrics average the three runs; pair-level analysis uses the 85 counterfactual pairs. Run-level sample deviations measure repeated execution. A 10,000-sample pair bootstrap (seed 20260726) measures variation across benchmark pairs.

The three runs produce 5,610 case trajectories and 2,805 pair-run records across the 11 configurations. Each run preserves separate action logs and submitted responses. Scoring is computed per run, and the reported configuration metrics average the three run-level results.

Every configuration receives the same task description, decision definitions, selectors, budgets, and response schema. INSPECT reveals one packet; FINAL submits the response and ends the case. Inspection consumes budget; invalid, repeated, and over-budget actions remain in the log and affect the score. Cases run in fresh contexts and tool states. Pair members run separately, and each case starts with an empty message and packet state. The supplementary appendix lists model identifiers, transports, completed-run counts, dates, and inference limits.

Alongside Process (Equation 1), we report Legacy, exact preferred-disposition accuracy, complete required-packet coverage, and the counterfactual pair measures. Process and Legacy use a 0–100 scale; the other columns are percentages. The appendix reports run variability, stochastic nulls, conditional outcomes, weight sensitivity, preloading transitions, human performance, and packet complexity. The Code and Data Supplement contains the benchmark, trajectories, scorers, experiment code, schemas, RTL/EDA artifacts, manifests, and reproduction tools.

4.2 Content-Blind Inspection Baselines

The baselines see the request, selector labels, and budget. They issue normal INSPECT actions, so packet choice and budget use are scored in the same way as model trajectories, but returned contents are discarded. The deterministic sweep combines label-based selectors with fixed decision rules and never uses required-packet annotations. Its strongest policy matches selector keywords, spends the available budget, and always requests more evidence. It scores 50.41, and its final answer is independent of every returned report value.

The 1,600 randomized policies score 39.86 on average, with a 95th percentile of 41.38 and maximum of 42.08. A request/selector-conditioned prior solves 6.51 pairs on average, and a test-prior policy solves 9.07, compared with 38.67 for the strongest model. The main Process comparison uses the strongest deterministic content-blind score, 50.41.

5 Results and Analysis

5.1 Overall Performance

Comparison with content-blind baselines. Process spans 28.90 points (Table 3 and Figure 1). GPT-5.5 scores 70.53, GLM-5.1 70.19, and Claude-Fable-5 69.28. Ten of 11 configurations exceed 50.41. Selector labels alone produce the baseline score; the leading scores require packet contents.

Process versus final-answer evaluation. Legacy exceeds Process most strongly for DeepSeek-v4-pro (4.00 points), Claude-Opus-4-8 (4.87), Claude-Sonnet-5 (4.72), and Claude-Haiku-4-5 (10.20). The **10.20-point gap** is the largest. In these runs, the final response names evidence that is missing from the action log. GLM-5.1 shows the reverse pattern by 2.41 points.

Disagreement among metrics. The metrics identify distinct leaders: GPT-5.5 leads Process at **70.53**, Claude-Fable-5 leads comparable Legacy at **71.65**, and GLM-5.1 leads strict Decision Accuracy at **69.4%**. Hunyuan-3-preview reaches **98.6%** full coverage with 64.5% accuracy; DeepSeek-v4-pro reaches 96.5% coverage with 57.6% accuracy; GLM-5.1 reaches 88.0% coverage with the

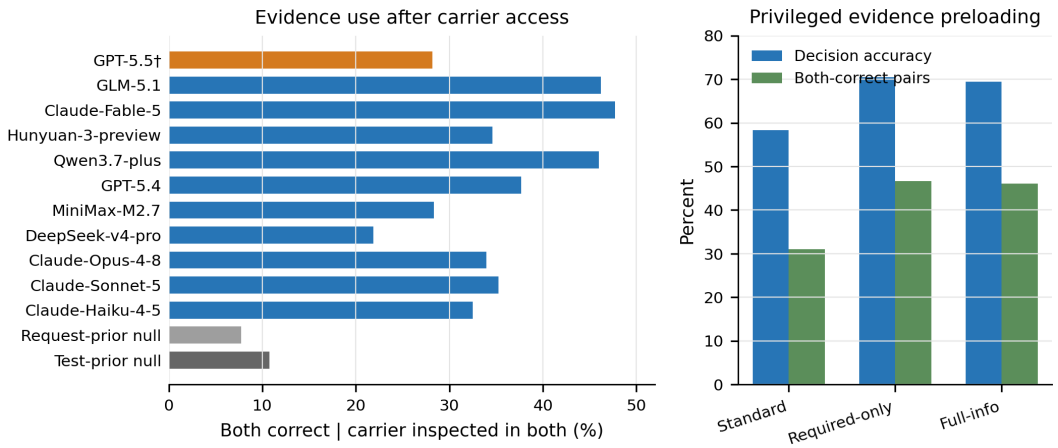

Figure 4: Completed experiments isolating evidence utilization. **Left:** both-correct pair rate after the carrier was inspected in both pair members; content-blind nulls score below the evaluated models. **Right:** mean strict decision accuracy and both-correct rate for the ten configurations with completed preloading runs, under the standard interface, required-evidence preloading, and all-packet preloading.

best accuracy. Coverage, response quality, and preferred-disposition accuracy rank the configurations differently.

5.2 Evidence Acquisition and Decision Calibration

Evidence-acquisition errors. Hunyuan-3-preview, DeepSeek-v4-pro, and GPT-5.5 cover every required packet in **98.6%, 96.5%, and 94.7%** of cases, respectively; their Decision Accuracy spans 57.6–64.5%. After both carriers are inspected, both-correct is 21.9–47.7% across models, compared with 7.7% for the request-conditioned content-blind null. After carrier access, the pair error rate is 52.3–78.1%.

Privileged evidence preloading. Required-only preloading raises mean decision accuracy from **58.3% to 70.5%** and both-correct from **31.0% to 46.6%**. Full-information reaches 69.4% and 46.1%. Appendix Table A5 reports every configuration.

The two preloading conditions produce different model-level orderings. Case-matched transitions show that Full-info changes correctness in 9.4–21.8% of cases: six models lose accuracy, one is unchanged, and three improve. The number of supplied reports changes the final commitment.

Decision-calibration errors. The observed range is **42.5–69.4% accuracy**. Among errors, the share occurring after complete required-evidence access reaches 92.0% for GPT-5.5 and DeepSeek-v4-pro and 95.6% for Hunyuan-3-preview. Figure 3 separates missing access, unchanged answers after access, and changes to another nonpreferred disposition.

A failed check can reject an action without supporting the proposed root cause. An incomplete measurement requires another report, not a rejection. The plain-rejection results and the response distributions in Figure 3 show errors in the strength and type of commitment. A binary accept/reject label merges those cases.

Weighting and human reference. Equal weighting, seven leave-one-component-out variants, and acquisition- and decision-heavy schemes retain 10 of 11 models above the rescored content-blind baseline; rank correlations with the original Process ranking are 0.936–1.000. Across 1,000 independent $\pm 50\%$ weight perturbations, the fifth-percentile correlation is 0.982 and the 10-of-11 conclusion is retained in every draw. On a matched 72-case (36-pair) subset, three engineers average 81.9% strict accuracy, 64.8% both-correct, and 94.4% full coverage (nominal $\kappa = 0.69$), establishing a human performance reference.

5.3 Counterfactual Response

GLM-5.1 leads with 38.7 both-correct pairs (**45.5%**); Qwen3.7-plus reaches 37.3 and Claude-Fable-5 36.0. Four configurations inspect the carrier in both cases of at least **96%** of pairs; their both-correct rates are 22.0–45.5%. Across configurations, the disposition is unchanged in 17.6–54.1% of pairs. Figures 3 and 4 separate access, response, and correctness.

The log reveals three failure modes: missing the carrier, ignoring it after inspection, and changing to the wrong disposition. A missing carrier marks a report-selection failure. A repeated disposition after carrier inspection marks unused evidence. A third trajectory opens the carrier and changes the answer to the wrong new disposition, marking a calibration error. The three error types correspond to report selection, evidence-state tracking, and disposition rules. A final-answer score merges them.

The deterministic content-blind policy selects the carrier in 79 pairs and solves zero pairs completely. Both-correct and unchanged-decision separate correct evidence-conditioned changes from arbitrary answer variation.

5.4 Agent Design Requirements

Evidence ledger. Each engineering quantity needs a source packet, validity conditions, and an inspection state. Enforcing these fields at submission time prevents citations to unopened reports and distinguishes observed, missing, and invalid measurements. The Process–Legacy differences measure violations of this link.

Boundary rules. Timing slack, analyzed-path count, constraint status, proof termination, coverage, and regression deltas determine the disposition through fixed predicates. Counterfactual pairs keep the language constant and change these fields. Carrier access above 96% and both-correct below 45.5% place many errors in boundary evaluation rather than report retrieval.

Commitment hierarchy. Approve, conditional approve, request more evidence, reject, and reject root-cause attribution represent different evidence states. The hierarchy separates a failed check from an unsupported explanation and an incomplete measurement from a measured violation. This mapping addresses unchanged answers and wrong-direction changes after carrier inspection.

Evidence volume. Required-only preloading raises mean accuracy to 70.5%; Full-info reaches 69.4%. Case-matched transitions include both recoveries and regressions. Packet selection acquires the controlling measurement and limits unrelated facts that change the commitment.

6 Discussion and Conclusion

ChipReasoning makes evidence acquisition observable and measures how retrieved engineering facts control the final commitment. Ten of eleven configurations beat the deterministic content-blind baseline, and the leading systems retrieve the carrier in nearly every pair. The best both-correct rate is 45.5%, and the pair error rate after carrier access is 52.3–78.1%. Most errors in the leading configurations occur when a retrieved measurement is converted into a disposition.

Stochastic nulls quantify content-blind pair success, preloading removes report selection, and the human reference calibrates task performance. These analyses separate evidence access from evidence use. A submission check binds each disposition to its controlling measurement and validates that measurement before the answer is returned.

Reproducibility Statement

The appendix and Code and Data Supplement provide the complete benchmark, three-run trajectories, model configurations, scorers, schemas, RTL and EDA artifacts, content-blind simulations, preloading outputs, human-reference records, manifests, and reproduction commands.

AI USE STATEMENT

Generative AI tools were used for benchmark candidate construction, analysis code, figure generation, and language editing. The authors checked the benchmark records, analyses, figures, and reported values.

REFERENCES

Tutu Ajayi, Vidya A. Chhabria, Mateus Fogaça, Soheil Hashemi, Abdelrahman Hosny, Andrew B. Kahng, et al. Toward an open-source digital flow: First learnings from the OpenROAD project. In *Proceedings of the 56th Annual Design Automation Conference (DAC)*, pages 1–4, 2019. doi: 10.1145/3316781.3326334.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. Benchmarking large language models in retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17754–17762, 2024a.

Zehui Chen, Weihua Du, Wenwei Zhang, Kuikun Liu, Jiangning Liu, Miao Zheng, Jingming Zhuo, Songyang Zhang, Dahua Lin, Kai Chen, and Feng Zhao. T-Eval: Evaluating the tool utilization capability of large language models step by step. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9510–9529, 2024b.

Fan Cui, Chenyang Yin, Kexing Zhou, Youwei Xiao, Guangyu Sun, Qiang Xu, Qipeng Guo, Demin Song, Dahua Lin, Xingcheng Zhang, and Yun Liang. OriGen: Enhancing RTL code generation with code-to-code augmentation and self-reflection, 2024.

Alex Forencich. verilog-uart: A basic UART-to-AXI stream verilog core. Software repository, 2026. URL https://github.com/alexforencich/verilog-uart. Accessed July 26, 2026.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.398.

Matt Gardner, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1307–1323. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.findings-emnlp.117. URL https://aclanthology.org/2020.findings-emnlp.117/.

Dan Gisselquist. wbuart32: A formally verified wishbone UART core. Software repository, 2026. URL https://github.com/ZipCPU/wbuart32. Accessed July 26, 2026.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR, 2017.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2018.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swe-bench: Can language models resolve real-world github issues? In *International Conference on Learning Representations (ICLR)*, 2024.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know, 2022. URL https://arxiv.org/abs/2207.05221.

Amita Kamath, Robin Jia, and Percy Liang. Selective question answering under domain shift. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

Qiufeng Li, Rongqian Chen, Quan Cheng, Chengxuan Wang, Sizhe Tang, Wuxi Li, Duo Ding, Chia-Tung Ho, Haoxing Ren, David Z. Pan, Tian Lan, and Weidong Cao. PDAgent-Bench: Characterizing, grounding, and architecting LLM/VLM agents for VLSI physical design, 2026.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *Proceedings of the Twelfth International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2305.20050.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research (TMLR)*, 2022.

Mingjie Liu, Teodor-Dumitru Ene, Robert Kirby, Chris Cheng, Nathaniel Pinckney, Rongjian Liang, Jonah Alben, Himyanshu Anand, Sanmitra Banerjee, Ismet Bayraktaroglu, et al. ChipNeMo: Domain-adapted LLMs for chip design. *arXiv preprint arXiv:2311.00176*, 2023a.

Mingjie Liu, Nathaniel Pinckney, Brucek Khailany, and Haoxing Ren. VerilogEval: Evaluating large language models for verilog code generation. In *Proceedings of the 2023 IEEE/ACM International Conference on Computer-Aided Design*, pages 1–8, 2023b. doi: 10.1109/ICCAD57390.2023.10323812.

Shang Liu, Wenji Fang, Yao Lu, Jing Wang, Qijun Zhang, Hongce Zhang, and Zhiyao Xie. RTLCoder: Fully open-source and efficient LLM-assisted RTL code generation technique. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 44(4):1448–1461, 2025a. doi: 10.1109/TCAD.2024.3483089.

Shang Liu, Yao Lu, Wenji Fang, Mengming Li, and Zhiyao Xie. OpenLLM-RTL: Open dataset and benchmark for LLM-aided design RTL generation, 2025b.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, et al. Agentbench: Evaluating llms as agents. In *International Conference on Learning Representations (ICLR)*, 2024.

Jiarui Lu, Thomas Holleis, Yizhe Zhang, Bernhard Aumayer, Feng Nan, Felix Bai, Shuang Ma, Shen Ma, Mengyu Li, Guoli Yin, Zirui Wang, and Ruoming Pang. ToolSandbox: A stateful, conversational, interactive evaluation benchmark for LLM tool use capabilities, 2024a.

Yao Lu, Shang Liu, Qijun Zhang, and Zhiyao Xie. Rtllm: An open-source benchmark for design rtl generation with large language model. In *Proceedings of the 29th Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 722–727, 2024b. doi: 10.1109/ASP-DAC58780.2024.10473904.

Chang Ma, Junlei Zhang, Zhihao Zhu, Cheng Yang, Yujiu Yang, Yaohui Jin, Zhenzhong Lan, Lingpeng Kong, and Junxian He. AgentBoard: An analytical evaluation board of multi-turn LLM agents. In *Advances in Neural Information Processing Systems*, 2024.

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: A benchmark for general AI assistants. In *Proceedings of the Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/25ae35b5b1738d80f1f03a8713e405ec-Abstract-Conference.html.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics (*SEM)*, 2018.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.442.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Secworks. secworks/aes: Verilog implementation of the AES block cipher. Software repository, 2026. URL https://github.com/secworks/aes. BSD-2-Clause license; accessed July 26, 2026.

Shailja Thakur, Baleegh Ahmad, Hammond Pearce, Benjamin Tan, Brendan Dolan-Gavitt, Ramesh Karri, and Siddharth Garg. VeriGen: A large language model for verilog code generation, 2023a.

Shailja Thakur, Jason Blocklove, Hammond Pearce, Benjamin Tan, Siddharth Garg, and Ramesh Karri. AutoChip: Automating HDL generation using LLM feedback, 2023b.

The OpenROAD Project and Parallax Software. OpenSTA: Parallax static timing analyzer. Software repository, 2026. URL https://github.com/The-OpenROAD-Project/OpenSTA. Accessed July 26, 2026.

Harsh Trivedi, Tushar Khot, Mareike Hartmann, Ruskin Manku, Vinty Dong, Edward Li, Shashank Gupta, Ashish Sabharwal, and Niranjan Balasubramanian. AppWorld: A controllable world of apps and people for benchmarking interactive coding agents, 2024.

Yun-Da Tsai, Mingjie Liu, and Haoxing Ren. RTLFixer: Automatically fixing RTL syntax errors with large language models, 2023.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback. *arXiv preprint arXiv:2211.14275*, 2022.

Claire Xenia Wolf. PicoRV32: A size-optimized RISC-V cpu. Software repository, 2026. URL https://github.com/YosysHQ/picorv32. ISC license; accessed July 26, 2026.

Clifford Wolf and Johann Glaser. Yosys: A free verilog synthesis suite. In *Proceedings of the 21st Austrian Workshop on Microelectronics*, 2013. URL https://yosyshq.net/yosys/files/yosys-austrochip2013.pdf.

Haoyuan Wu, Zhuolun He, Xinyun Zhang, Xufeng Yao, Su Zheng, Haisheng Zheng, and Bei Yu. ChatEDA: A large language model powered autonomous agent for EDA. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 43(10):3184–3197, 2024. doi: 10.1109/TCAD.2024.3383347.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In *Advances in Neural Information Processing Systems*, 2024.

John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. SWE-agent: Agent-computer interfaces enable automated software engineering. In *Advances in Neural Information Processing Systems*, 2024.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. In *Proceedings of the Thirteenth International Conference on Learning Representations*, 2025.

Zhongkai Yu, Chenyang Zhou, Yichen Lin, Hejia Zhang, Haotian Ye, Junxia Cui, Zaifeng Pan, Jishen Zhao, and Yufei Ding. ChipBench: A next-step benchmark for evaluating LLM performance in AI-aided chip design, 2026.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. Webarena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations (ICLR)*, 2024.

Yuchen Zhuang, Yue Yu, Kuan Wang, Haotian Sun, and Chao Zhang. ToolQA: A dataset for LLM question answering with external tools. In *Advances in Neural Information Processing Systems*, volume 36, pages 50117–50143, 2023.

A Additional Validation

This appendix reports the completed-run analyses. Unless stated otherwise, model results average three independently completed runs on 170 cases (85 counterfactual pairs). Run-level sample standard deviations measure repeated-execution variability. Pair-bootstrap intervals in the submitted machine-readable results measure variation across the 85 benchmark pairs. We report both quantities separately.

A.1 Repeated-Run Variability

Table A1 reports means and sample standard deviations over the three completed runs. The Process standard deviation is at most 1.99 points. Pair-count means are computed directly from the three completed runs.

Table A1: Mean $\pm$ sample standard deviation over three completed runs. Pair counts have denominator 85 per run. †GPT-5.5 is the generator-involved reference.

Model	Process	Decision acc. (%)	Full req. (%)	Both-correct pairs	Carrier-both pairs
GPT-5.5†	70.53 $\pm$ 1.50	60.39 $\pm$ 2.23	94.71 $\pm$ 1.56	24.00 $\pm$ 2.65	84.67 $\pm$ 0.58
GLM-5.1	70.19 $\pm$ 0.65	69.41 $\pm$ 0.59	88.04 $\pm$ 1.48	38.67 $\pm$ 2.08	82.33 $\pm$ 0.58
Claude-Fable-5	69.28 $\pm$ 1.75	67.45 $\pm$ 0.68	84.12 $\pm$ 2.35	36.00 $\pm$ 1.00	77.00 $\pm$ 1.73
Hunyuan-3-preview	65.99 $\pm$ 0.19	64.51 $\pm$ 2.38	98.63 $\pm$ 1.48	30.33 $\pm$ 2.08	84.00 $\pm$ 1.00
Qwen3.7-plus	64.05 $\pm$ 1.99	66.67 $\pm$ 2.90	81.57 $\pm$ 0.34	37.33 $\pm$ 0.58	74.33 $\pm$ 1.15
GPT-5.4	63.54 $\pm$ 1.40	54.51 $\pm$ 2.96	76.86 $\pm$ 0.34	24.67 $\pm$ 1.15	58.00 $\pm$ 2.00
MiniMax-M2.7	60.16 $\pm$ 1.14	54.12 $\pm$ 1.56	81.37 $\pm$ 2.38	22.33 $\pm$ 2.52	69.00 $\pm$ 1.00
DeepSeek-v4-pro	59.23 $\pm$ 0.32	57.65 $\pm$ 2.70	96.47 $\pm$ 1.18	18.67 $\pm$ 2.52	83.67 $\pm$ 1.15
Claude-Opus-4-8	56.89 $\pm$ 0.03	56.86 $\pm$ 1.89	60.98 $\pm$ 1.22	26.67 $\pm$ 2.31	67.00 $\pm$ 1.00
Claude-Sonnet-5	55.07 $\pm$ 0.52	52.94 $\pm$ 0.59	71.37 $\pm$ 0.34	22.67 $\pm$ 1.53	54.33 $\pm$ 0.58
Claude-Haiku-4-5	41.63 $\pm$ 1.20	42.55 $\pm$ 0.90	44.90 $\pm$ 0.34	12.67 $\pm$ 1.53	26.00 $\pm$ 1.00

A.2 Stochastic Content-Blind Pair Nulls

The deterministic content-blind sweep scores 50.41 Process points. Independently sampled content-blind policies use the visible request and selector labels, discard returned packet contents, and sample the final disposition from either a cross-fitted prior or the test-set prior. Table A2 reports their pair success.

A.3 Performance after Evidence Access

Table A3 conditions outcomes on observable access events. After both carriers are inspected, the conditional both-correct rate is 21.85–47.70%. The final column reports the share of decision errors in cases with complete required-evidence access.

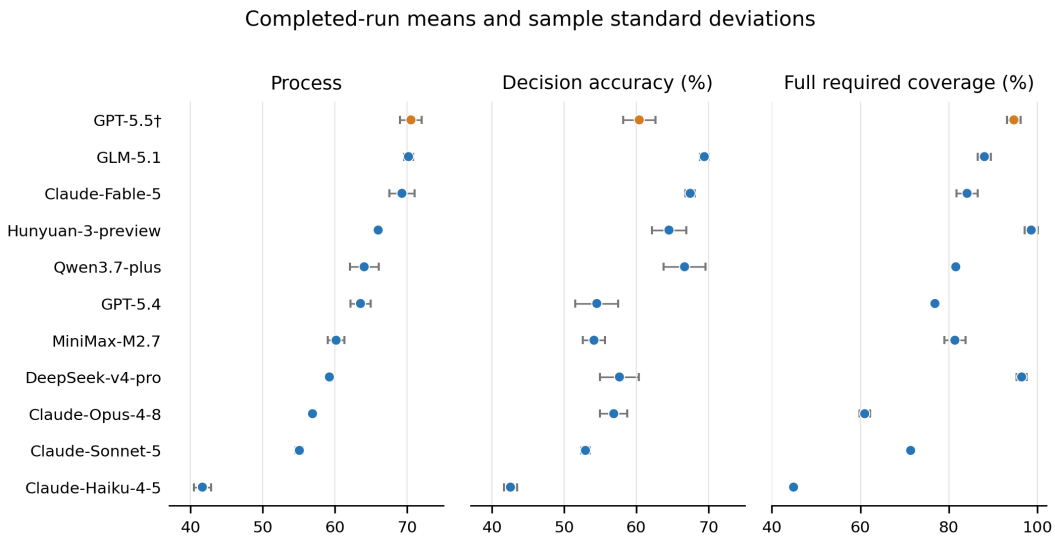

Figure A1: Completed-run variability for Process, strict decision accuracy, and complete required-evidence coverage. Points are three-run means and error bars are sample standard deviations.

Table A2: Pair-level stochastic content-blind nulls. "Both" is the number of both-correct pairs out of 85. Both | carrier conditions on selecting the carrier in both pair members.

Content-blind policy	Simulations	Both mean	Both p95	Both max	Decision acc. (%)	Both | carrier (%)
Request/selector prior, cross-fitted	1,600	6.51	11	18	27.59	7.73
Test-set disposition prior	1,600	9.07	14	19	32.63	10.77

Table A3: Conditional evidence-utilization diagnostics (percent). The final column is the share of all decision errors occurring after complete required-evidence access.

Model	Both correct | carrier both	Decision correct | full req.	Unchanged | carrier both	Changed wrong | carrier both	Errors after full req.
GPT-5.5†	28.18	63.78	40.24	32.71	91.96
GLM-5.1	46.20	73.50	17.81	36.62	79.61
Claude-Fable-5	47.70	76.64	27.83	24.67	65.36
Hunyuan-3-preview	34.59	64.32	34.33	30.12	95.60
Qwen3.7-plus	45.98	76.53	23.79	27.65	57.37
GPT-5.4	37.68	68.12	31.58	30.77	58.60
MiniMax-M2.7	28.33	60.30	29.66	42.31	63.53
DeepSeek-v4-pro	21.85	59.31	34.91	43.23	92.01
Claude-Opus-4-8	33.96	73.99	34.66	30.44	34.30
Claude-Sonnet-5	35.27	65.49	31.55	32.33	51.30
Claude-Haiku-4-5	32.49	62.50	51.83	13.84	30.91

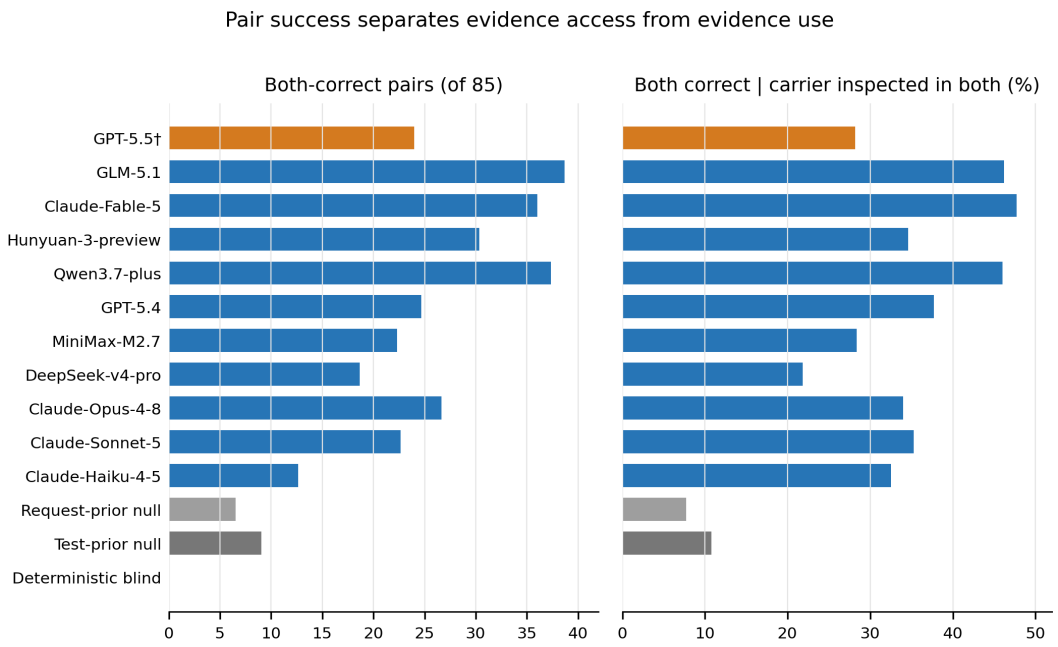

Figure A2: Pair-level success for model configurations and content-blind nulls. The left panel reports both-correct pair counts; the right panel conditions success on inspecting the carrier in both pair members.

A.4 Process-Weight Sensitivity

We recomputed Process after equal weighting, leaving out each component in turn, and emphasizing either acquisition or decision components. In every scheme, 10 of 11 configurations score above the corresponding rescored content-blind baseline. Rank correlation with the original ranking is 0.936–1.000. Across 1,000 independent $\pm 50\%$ perturbations of the seven weights, the median Spearman correlation exceeds 0.99, the fifth percentile is 0.982, the leading-three set is unchanged in 96.1% of draws, and the 10-of-11 baseline conclusion is retained in every draw.

Table A4: Representative Process-weight sensitivity results. Each blind score is recomputed under the same weights as the model scores.

Scheme	Spearman ρ	Top-three overlap	Models above blind	Rescored blind
Equal weights	0.991	3/3	10/11	51.05
Leave required evidence out	0.936	3/3	10/11	46.63
Leave decision calibration out	0.973	2/3	10/11	56.57
Other five leave-one-out variants	0.991–1.000	3/3	10/11	49.29–50.37
Acquisition-heavy	0.936	2/3	10/11	53.24
Decision-heavy	0.945	3/3	10/11	46.96

B Privileged Evidence Preloading

Required-only supplies the annotated required packets, whereas Full-info supplies all six packets. These conditions directly measure strict decision accuracy and both-correct pairs after preloading. The reported means cover the ten configurations with completed preloading runs.

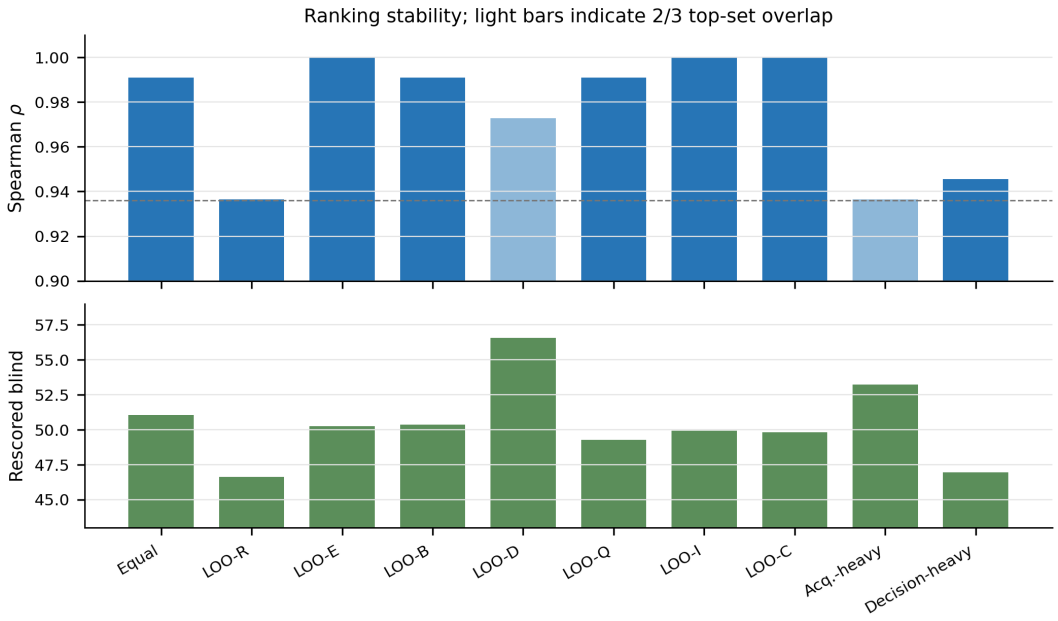

Figure A3: Process-weight sensitivity. Rank correlations are at least 0.936 across deterministic alternatives, and every scheme retains 10 of 11 configurations above its rescored content-blind baseline.

Table A5: Completed privileged evidence-preloading results. Blue bold cells mark the highest non-reference result in each column.

	Preferred decision accuracy (%)			Both-correct pairs (%)		
Model	Standard	Required-only	Full-info	Standard	Required-only	Full-info
GPT-5.5†	60.4	77.1	71.8	28.2	55.3	45.9
Claude-Fable-5	67.5	78.8	**77.7**	42.4	58.8	**56.5**
GLM-5.1	**69.4**	66.5	66.5	**45.5**	43.5	41.2
Qwen3.7-plus	66.7	70.6	71.2	43.9	42.4	49.4
GPT-5.4	54.5	62.4	68.8	29.0	36.5	41.2
DeepSeek-v4-pro	57.6	61.8	58.2	22.0	30.6	29.4
MiniMax-M2.7	54.1	63.5	55.9	26.3	37.6	31.8
Claude-Opus-4-8	56.9	**82.3**	73.5	31.4	**67.1**	55.3
Claude-Sonnet-5	52.9	63.5	75.9	26.7	35.3	55.3
Claude-Haiku-4-5	42.5	78.2	74.7	14.9	58.8	55.3
Mean	**58.3**	**70.5**	**69.4**	**31.0**	**46.6**	**46.1**

B.1 Case-Matched Required-Only to Full-Info Transitions

Table A6 matches each case across the two preloaded conditions. Six configurations lose net accuracy, three improve, and GLM-5.1 is unchanged. Additional reports recover some errors and change some correct commitments.

Table A6: Case-matched correctness transitions from Required-only to Full-info, over 170 cases per configuration.

Model	Correct → incorrect	Incorrect → correct	Net accuracy (points)	Correctness changed (%)
GPT-5.5†	17	8	−5.29	14.71
Claude-Fable-5	9	7	−1.18	9.41
GLM-5.1	13	13	0.00	15.29
Qwen3.7-plus	11	12	+0.59	13.53
GPT-5.4	8	19	+6.47	15.88
DeepSeek-v4-pro	15	9	−3.53	14.12
MiniMax-M2.7	21	8	−7.65	17.06
Claude-Opus-4-8	22	7	−8.82	17.06
Claude-Sonnet-5	8	29	+12.35	21.76
Claude-Haiku-4-5	13	7	−3.53	11.76

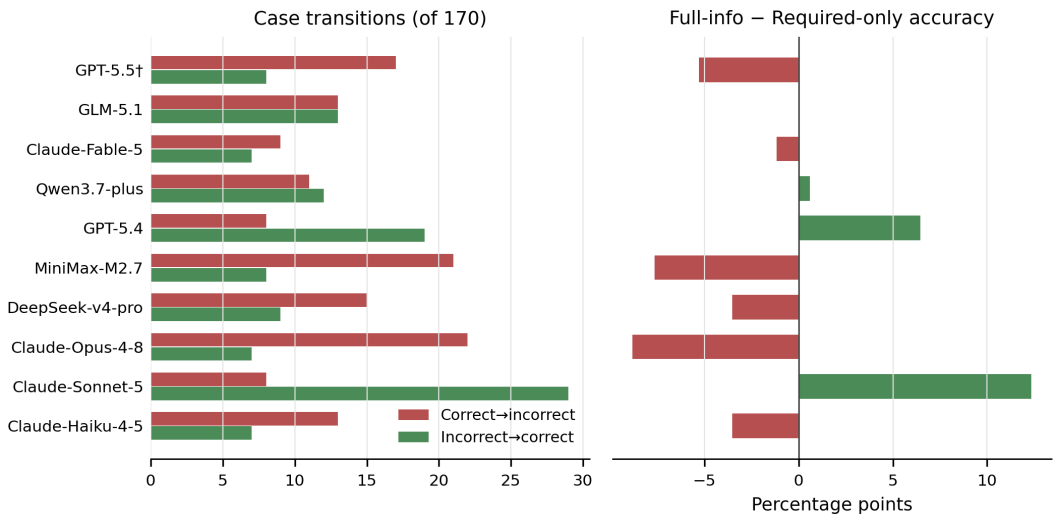

Figure A4: Case-matched correctness transitions from Required-only to Full-info. The left panel separates harmful and recovered transitions; the right panel reports the net accuracy change.

C Human Reference and Packet Complexity

C.1 Human Engineer Reference

Three engineers independently completed the same-budget interface on a matched 72-case subset comprising 36 complete pairs. The panel reaches 84.0% raw agreement with nominal $\kappa = 0.69$ and a median of 15 minutes per pair.

C.2 Packet and Raw-Artifact Size

The benchmark normalizes heterogeneous EDA outputs into schema-grounded packets containing report fields, values, status flags, and supporting text. Table A8 gives token and byte distributions by artifact class. Across the paired raw-artifact audit, the median raw-to-packet ratio is 2.77 by tokens and 2.54 by bytes. These measurements quantify the compression from raw artifacts to benchmark packets.

Table A7: Human engineer reference on 72 cases (36 complete pairs) under the same packet-inspection budgets.

Evaluator	Decision acc. (%)	Both-correct (%)	Full required (%)	Median sec./case
Engineer 1	79.17	61.11	93.06	7.6
Engineer 2	84.72	69.44	95.83	8.0
Engineer 3	81.94	63.89	94.44	6.9
Mean	**81.94**	**64.81**	**94.44**	–

Table A8: Packet-size distributions over the completed benchmark data.

Artifact class	Median tokens	Token p95	Median bytes	Byte p95
Synthesis	206	541	829	2,222
Static timing	312	880	1,201	3,396
Lint	154	425	644	1,724
Simulation/experiments	263	691	1,101	2,666
Formal verification	409	992	1,609	4,205
Regression	294	779	1,177	3,004

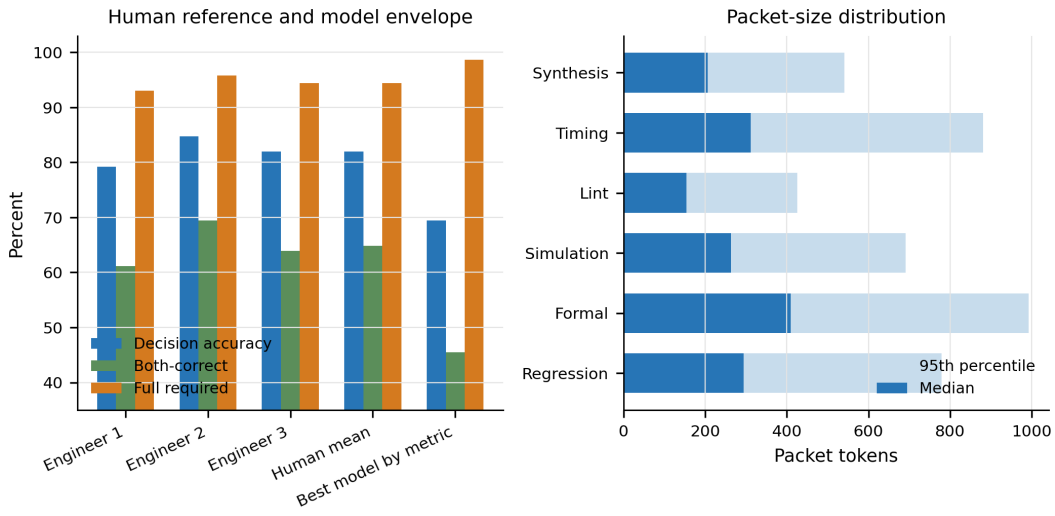

Figure A5: Human-reference performance and packet complexity. The model envelope uses the strongest observed model value for each metric; packet bars report median and 95th-percentile token counts.

D Execution and Benchmark Record

Table A9: Model identifiers, transports, completed-run counts, dates, and inference limits.

Display name	Exact identifier	Transport	Runs	Run date	Max turns	Max tokens
GPT-5.5	gpt-5.5	Codex-hosted	3	2026-07-13/14	6	1800
Claude-Fable-5	claude-fable-5	Claude CLI	3	2026-07-13/14	6	1800
GLM-5.1	glm-5.1	External API	3	2026-07-13/14	6	1800
Hunyuan-3-preview	hunyuan-3-preview	External API	3	2026-07-13/14	6	1800
Qwen3.7-plus	qwen3.7-plus	External API	3	2026-07-13/14	6	1800
GPT-5.4	gpt-5.4	Codex-hosted	3	2026-07-13/14	6	1800
DeepSeek-v4-pro	deepseek-v4-pro	External API	3	2026-07-13/14	6	1800
MiniMax-M2.7	minimax-m2.7	External API	3	2026-07-13/14	6	1800
Claude-Opus-4-8	claude-opus-4-8	Claude CLI	3	2026-07-13/14	6	1800
Claude-Sonnet-5	claude-sonnet-5	Claude CLI	3	2026-07-13/14	6	1800
Claude-Haiku-4-5	claude-haiku-4-5-20251001	Claude CLI	3	2026-07-13/14	6	1800

Table A10: Benchmark construction and annotation summary.

Design element	Benchmark implementation	Measurement role
Coverage	170 cases in 85 pairs; four open RTL designs; six engineering domains; six artifact classes	Exercises evidence decisions across implementation and verification
Pair control	Pair members share the request, selectors, budget, and five packet contents; only the decision-critical carrier changes	Localizes response to one changed fact
Retrieval pressure	Six hidden packets with an inspection budget of three to five	Makes evidence selection an observable action
Context isolation	Every case starts in a fresh context; paired variants are hidden	Prevents direct counterfactual comparison
Decision target	Approve 35, conditional approve 35, request more evidence 36, reject 34, and reject root-cause attribution 30	Tests the supported strength of commitment
Audit trail	$200 \rightarrow 184 \rightarrow 170$ cases, retaining complete pairs after boundary and ambiguity checks	Preserves pair structure through filtering

E Representative Trajectories and Reproducibility

E.1 Representative Trajectories

Three trajectories show a supported rejection after decision-critical timing evidence, a calibrated request for additional evidence after a formal-analysis timeout, and an unsupported root-cause attribution made without the required evidence. The complete machine-readable records are included in the submitted Code and Data Supplement.

Table A11: Representative trajectory: correct rejection from timing evidence (case `v18_counterfactual100_c002`, GPT-5.5).

Field	Recorded trajectory
Request	Determine whether AES meets the physical-implementation targets from timing and lint evidence; budget $B = 3$.
Selectors	E1 simulation; E2 synthesis; E3 timing; E4 lint; E5 formal; E6 regression.
Inspections	E3: constraints loaded, 8,300 paths, setup WNS -0.07 ns, setup TNS -13.4 ns, hold WNS -0.01 ns. E4: zero warnings and errors across 43 files.
Final JSON fields	`decision=reject`; selected E3/E4; derived values include setup WNS, setup TNS, hold WNS, and zero lint issues with their source selectors. Missing evidence is empty. The next action is to fix or re-constrain the failing paths and rerun STA until setup/hold WNS are nonnegative and TNS is zero.

Field	Recorded trajectory
Preferred / score	Preferred disposition: reject. Process 97.0; component points $(R, E, B, D, Q, I, C) = (25, 10, 10, 20, 15, 10, 7)$.

Table A12: Representative trajectory: calibrated request after an inconclusive formal run (case `v18_counterfactual100_c034`, Claude-Fable-5).

Field	Recorded trajectory
Request	Determine whether formal, simulation, and lint evidence establish a known-good AES implementation state; budget $B = 4$.
Selectors	E1 lint; E2 timing; E3 simulation; E4 synthesis; E5 formal; E6 regression.
Inspections	E5: return code 124, no counterexample, run not terminated, two properties checked. E3: 50/50 tests pass with 96% coverage. E1: zero warnings and errors across 44 files.
Final JSON fields	`decision=request_more_evidence`; selected E5/E3/E1; derived values include the formal return code and termination flag, 100% simulation pass rate, 96% coverage, and zero lint issues. Missing evidence records a completed formal run and the achieved proof bound. The next action is to rerun formal verification with a larger timeout or documented bound and then re-review the combined evidence.
Preferred / score	Preferred disposition: request more evidence. Process 97.0; component points $(R, E, B, D, Q, I, C) = (25, 10, 10, 20, 15, 10, 7)$.

Table A13: Representative failure: unsupported root-cause disposition without the required packets (case `v18_counterfactual100_c115`, Claude-Fable-5).

Field	Recorded trajectory
Request	Assess whether a reported AES power regression is attributable to a switching-activity change, using synthesis, simulation, timing, and lint evidence; budget $B = 5$.
Selectors	E1 simulation; E2 synthesis; E3 regression; E4 timing; E5 lint; E6 formal.
Inspections	Only E1 is inspected: 51/51 tests pass, coverage is available at 96%, and 21 transitions are reported.
Final JSON fields	`decision=reject_root_cause`, with no inspection of the required synthesis and timing packets. The response reports a 10% power-proxy delta, a fixed 0.5 toggle-rate assumption, no activity file, and zero failing timing endpoints using uninspected E2/E4 as sources. It asks for activity-annotated power estimation and a pre/post synthesis-netlist diff.
Preferred / score	Preferred disposition: request more evidence. Process 24.0; component points $(R, E, B, D, Q, I, C) = (0, 10, 10, 0, 0, 4, 0)$.

E.2 Reproducibility Record

ChipReasoning evaluates evidence acquisition and evidence-to-decision behavior over four open RTL designs, six engineering domains, and six report classes. The paired construction isolates a single decision-controlling report change, and the action log records report access before the final commitment.

The author team reviewed every retained case. GPT-5.5 participated in candidate construction before the benchmark freeze and is marked as a generator-involved reference. The submitted Code and Data Supplement contains the complete benchmark, all three-run trajectories, preloading outputs, content-blind simulations, human-reference records, scorers, schemas, RTL and EDA artifacts, manifests, licenses, and reproduction tools.