CircuitSense: A Hierarchical Circuit System Benchmark Bridging Visual Comprehension and Symbolic Reasoning in Engineering Design Process

Arman Akbari[1] **Jian Gao**[1] **Yifei Zou**[1] **Mei Yang**[1]

Jinru Duan[2] **Dmitrii Torbunov**[2] **Yanzhi Wang**[2] **Yihui Ren** [2] **Xuan Zhang**[2]

[1]Northeastern University [2]Brookhaven National Laboratory

Abstract

Engineering design operates through hierarchical abstraction from system specifications to component implementations, requiring visual understanding coupled with mathematical reasoning at each level. While Multi-modal Large Language Models (MLLMs) excel at natural image tasks, their ability to extract mathematical models from technical diagrams remains unexplored. We present **CircuitSense**, a comprehensive benchmark evaluating circuit understanding across this hierarchy through 8,006+ problems. Our benchmark uniquely examines the complete engineering workflow: Perception, Analysis, and Design, with a particular emphasis on the critical but underexplored capability of deriving symbolic equations from visual inputs. We introduce a hierarchical synthetic generation pipeline with auto-derived symbolic equation labels. Comprehensive evaluation of six state-of-the-art MLLMs, reveals fundamental limitations in visual-to-mathematical reasoning. Closed-source models achieve over 85% accuracy on perception tasks involving component recognition and topology identification, yet their performance on symbolic derivation and analytical reasoning falls below 19%, exposing a critical gap between visual parsing and symbolic reasoning. Models with stronger symbolic reasoning capabilities consistently achieve higher design task accuracy, confirming the fundamental role of mathematical understanding in circuit synthesis and establishing symbolic reasoning as the key metric for engineering competence. **Project page:** https://circuitsense-benchmark.github.io

1 Introduction

Mathematical modeling forms the foundation of all engineering disciplines. Engineers translate visual representations like circuit schematics, mechanical systems, and optical layouts into equations to predict behavior and prevent failures. In electronics, a phase-locked loop with insufficient phase margin will oscillate catastrophically, yet this can only be predicted by analyzing the transfer function's poles and zeros [6]. This visual-to-mathematical translation determines engineering success: without deriving equations, engineers cannot verify stability, optimize performance, or predict failures. While this capability defines human engineering expertise, no current AI system can perform this fundamental translation. Unlike single-domain problems in geometry or physics, engineering requires mathematical reasoning across hierarchical levels from components to complete systems which is a capability that remains beyond current AI.

While Multi-modal Large Language Models (MLLMs) excel at visual perception tasks, they exhibit a critical limitation: the inability to derive symbolic equation from visual representations [8, 9, 12]. This

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: MATH-AI.

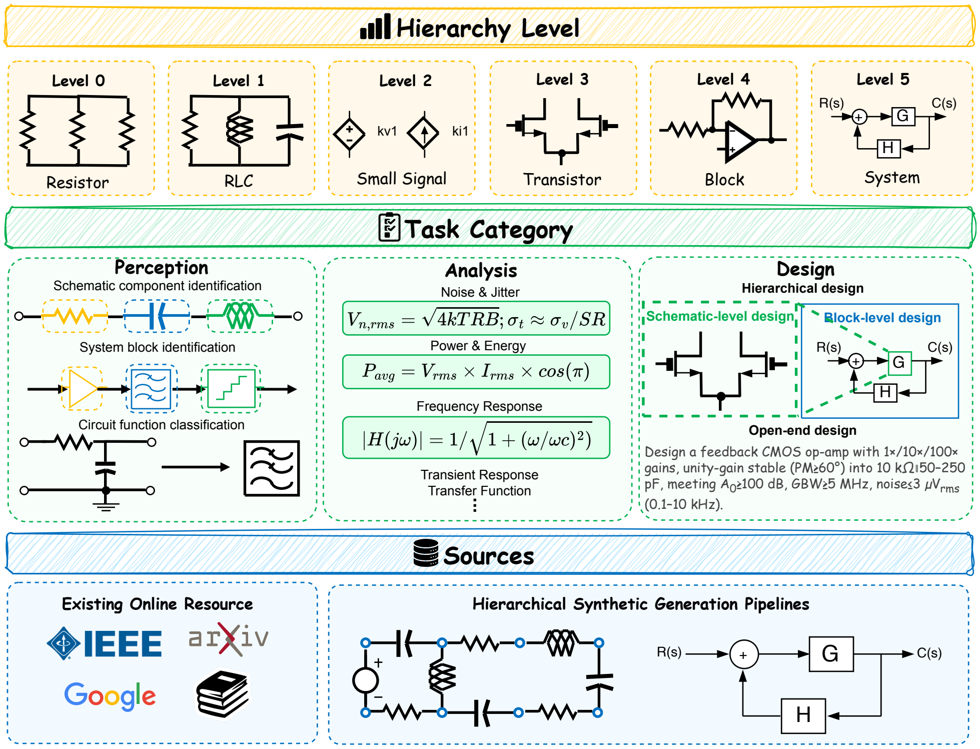

Hierarchy Level
Level 0
Resistor
Level 1
RLC
Level 2
kv1
ki1
Small Signal
Level 3
Transistor
Level 4
Block
Level 5
R(s)
C(s)
G
H
System
Task Category
Perception
Schematic component identification
System block identification
Circuit function classification
Analysis
Noise & Jitter
$V_{n,rms} = \sqrt{4kTRB}; \sigma_t \approx \sigma_v/SR$
Power & Energy
$P_{avg} = V_{rms} \times I_{rms} \times cos(\pi)$
Frequency Response
$|H(j\omega)| = 1/\sqrt{1+(\omega/\omega c)^2)}$
Transient Response
Transfer Function
⋮
Design
Hierarchical design
Schematic-level design
Block-level design
R(s)
C(s)
G
H
Open-end design
Design a feedback CMOS op-amp with 1×/10×/100× gains, unity-gain stable (PM≥60°) into 10 kΩ∥50–250 pF, meeting A_0≥100 dB, GBW≥5 MHz, noise≤3 μV_{rms} (0.1–10 kHz).
Sources
Existing Online Resource
IEEE
arXiv
Google
Hierarchical Synthetic Generation Pipelines
+
−
R(s)
+
G
C(s)
H

Figure 1: Benchmark overview. CircuitSense evaluates circuit systems understanding across six hierarchy levels (resistor networks to system block diagrams), three task category (Perception, Analysis with equation derivation, and Design), using both curated problems and synthetically generated circuits systems with ground-truth symbolic equations.

failure is not merely technical but fundamental: equation derivation distinguishes true engineering comprehension from pattern matching. Existing visual circuit benchmarks [17, 16] focus primarily on recognition-based tasks like identifying component types, answering basic multiple-choice questions, or performing shallow numerical calculations. The core capability that defines circuit understanding remains untested: the ability to extract mathematical relationships from visual circuit topology that is consistent across multiple system hierarchies.

To fill this gap, we propose **CircuitSense**, the first benchmark that systematically evaluates circuit understanding through hierarchical mathematical reasoning. CircuitSense comprises 8,006 problems organized across six hierarchy levels from resistor networks to system-level block diagrams with open-ended and multiple-choice formats, testing three task categories that mirror the engineering workflow: Perception, Analysis, and Design. Our benchmark combines 2,986 carefully curated problems from authoritative textbooks and documents with 5,020 synthetically generated circuits, uniquely emphasizing symbolic derivation. We introduce a hierarchical synthetic generation pipeline consisting of a circuit schematic generator with guaranteed symbolic ground-truth equations, and a block diagram generator with symbolic transfer function ground-truth. This dual approach ensures both component-level depth and system-level breadth while preventing dataset contamination.

As illustrated in Figure 2, we evaluated CircuitSense over 6 state-of-the-art MLLMs and Gemini-2.5-Pro [4] showed the best performance among all tasks. Our main contribution and findings can be summarized as:

- **First Multi-Level Visual-to-Analytical Benchmark:** We introduce the first benchmark that systematically evaluates understanding across engineering abstraction levels, from system-level block diagrams to component-level schematics, testing how models connect visual patterns at different scales to their mathematical representations.

Table 1: Benchmark statistics.

Task	Subcategory	Count
Perception		*806*
	Component Detection	200
	Connection Identification	200
	Function classification	406
Analysis		*7043*
	Frequency Response	184
	Transient Response	3811
	Transfer Function Analysis	1736
	Small Signal Analysis:	915
	CMR & PSRR	54
	Noise & Jitter Analysis	121
	Power & Energy Analysis	222
Design		*157*
	Schematic-level	63
	Block-level	56
	Hierarchical	38
Total		**8,006**

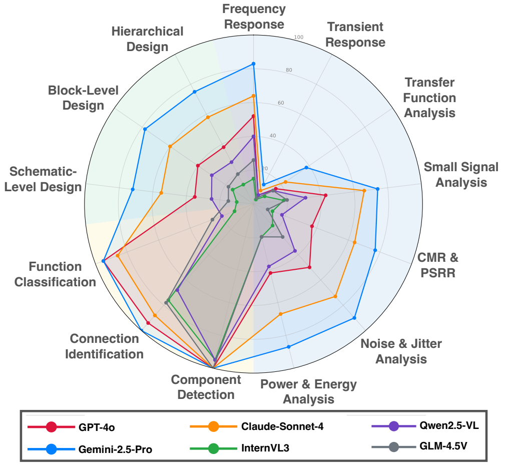


Figure 2: Result of 6 representative MLLMs on Perception, Analysis, and Design tasks.

- **Hierarchical Synthetic Generation Pipeline:** We developed two synthetic generation pipeline producing samples with guaranteed ground-truth equations: (i) component-level circuits with controlled complexity progression, and (ii) system-level block diagrams with hierarchical feedback structures, enabling isolated evaluation of visual comprehension and mathematical reasoning at each abstraction level.
- **Extensive Multi-Scale Performance Analysis:** Through systematic evaluation of six state-of-the-art MLLMs and detailed analysis of derivation attempts, we demonstrate that while closed-source models achieve over 85% accuracy on perception tasks, they fail catastrophically at symbolic analysis (below 19% accuracy), with specific bottlenecks identified including systematic output impedance misinterpretation and algebraic manipulation errors. Our experiments confirm that models with stronger equation derivation capabilities consistently achieve higher design task performance, establishing mathematical understanding as prerequisite for AI-assisted circuit synthesis.

2 CircuitSense

In this section we introduce **CircuitSense** which evaluates visual circuit understanding through a hierarchical framework that mirrors the complete engineering design process, from high-level system architecture to detailed component implementation. As shown in Figure 1, the benchmark is organized along two primary axes: task categories and hierarchy levels. The dataset spans three task categories: Perception (890), Analysis (7043), and Design (157), with Analysis comprising the majority of problems as it directly tests the critical capability of extracting mathematical models from visual circuits. Problems are distributed across six hierarchical levels from basic resistor networks to system-level block diagrams, enabling fine-grained assessment of where visual-to-mathematical translation fails.

2.1 Hierarchical Synthetic Generation Pipeline

Circuit Schematic Generator Our schematic generator extends the MAPS framework [20] for Linear Pure Resistive Circuits (LPRC) to support the full spectrum of analog components. We construct circuit on an $m \times n$ grid where dimensions are sampled from a Discrete Probability Distribution to ensure topological diversity. We support 18 component types organized by complexity. The grid topology is translated into SPICE-compatible netlists through systematic node labeling and component enumeration. Circuit validation occurs at three levels: topological verification ensures no shorted components and proper control relationships, SPICE simulation confirms DC operating points and AC responses, and symbolic analysis through Lcapy [7] extracts ground-truth transfer

Table 2: Perception task results

Model	Component Detec.(%)	Connection Ident.(%)	Function Class.(%)
GPT-4o	100	94	95
Gemini-2.5-Pro	**100**	**100**	**95**
Claude-Sonnet-4	100	88	86
InternVL3-72B	95	76	12
Qwen2.5-VL	95	68	20
GLM-4.5V	100	78	26

Table 3: Design task results

Model	Schematic-level(%)	Block-level(%)	Hierarchical-Design(%)
GPT-4o	10.52	36.36	18.92
Gemini-2.5-Pro	**36.38**	**67.27**	**51.35**
Claude-Sonnet-4	17.54	51.83	29.83
InternVL3-72B	7.01	52.73	29.73
Qwen2.5-VL	8.76	30.91	18.92
GLM-4.5V	15.79	50.91	32.35

Table 4: Accuracies of different models on Analysis subcategories.

Model	Frequency Response	Transient Response	Transfer Function Analysis	Small Signal Analysis	CMR & PSRR	Noise & Jitter Analysis	Power & Energy Analysis
GPT-4O	52	6	16	43	37	50	42
Gemini-2.5-Pro	**83**	**13**	**38**	**74**	**77**	**90**	**87**
Claude-Sonnet-4	64	9	23	66	64	73	67
InternVL3-78B	15	3	8	18	12	17	20
Qwen2.5-VL-72B-Instruct	40	6	14	31	18	37	38
GLM-4.5V	26	4	14	20	9	26	20

functions $H(s) = V_{out}(s)/V_{in}(s)$ and nodal equations via Modified Nodal Analysis. To manage computational complexity, we implement adaptive timeouts based on circuit complexity scores, bypassing symbolic analysis for circuits exceeding practical computation limits.

Block Diagram Generator The block diagram pipeline constructs control systems through a two-phase approach. First, we build a main signal path with $n \in [\tau_b, \tau_e]$ components (transfer functions and summing junctions) placed sequentially. Then we add $n_{fb} \in [0, \tau_{fb}]$ feedback loops and $n_{ff} \in [0, \tau_{ff}]$ feedforward paths, with each auxiliary path having probability $p_{block} = 0.5$ of containing an intermediate block. The algorithm prevents duplicate connections through set-based tracking and randomly assigns sign conventions at summing junctions. This generates diverse architectures from simple unity feedback to complex multi-loop systems typical of ADCs and PLLs.

This hierarchical approach to synthetic generation ensures comprehensive coverage from low-level component interactions to high-level system behavior, providing the multi-scale evaluation necessary for assessing true circuit understanding.

Evaluation Framework We employ two evaluation strategies depending on problem format. For multiple-choice questions, we use exact answer matching after standardized formatting. For open-ended questions requiring numerical or short-form answers, we employ LLM-as-a-judge evaluation, accounting for equivalent representations and unit conversions, determining correctness based on mathematical equivalence rather than exact string matching. For design tasks that require simulations, we simulate them by Ngspice [11] with Skywater 130nm PDK [18].

Evaluating symbolic mathematical expressions presents unique challenges since a single equation can be represented in numerous algebraically equivalent forms. To address this, we implement a rigorous symbolic comparison pipeline using SymPy [10] that performs: (1) parsing both predicted and ground-truth equations into symbolic expression trees, (2) algebraic simplification, (3) verification through symbolic subtraction, and (4) numerical validation by evaluating both expressions at 100 random complex frequency points when symbolic comparison is computationally intractable. This multi-pronged approach ensures robust evaluation even when models produce correct but differently formatted equations.

3 Experiments

Perception Task Table 2 reveals a clear divide between closed and open-source models in visual circuit understanding. Closed-source models excel with 86%+ accuracy—GPT-4o and Gemini-2.5-Pro achieve near-perfect performance (94-100%), while Claude-Sonnet-4 maintains 85%+. In contrast, open-source models struggle with basic recognition, with GLM-4.5V achieving only 26%

Table 5: Performance comparison on our hierarchical synthetic problems with symbolic equation ground truth.

Model	**Level 0** **(Resistor)**	**Level 1** **(RLC)**	**Level 2** **(Small Signal)**	**Level 4** **(Block)**	**Level 5** **(System)**	**Overall**
GPT-4o	1.50	3.33	5.80	7.33	9.65	4.98
Claude-Sonnet-4	2.83	5.16	5.80	11.64	7.89	6.29
Gemini-2.5-Pro	**3.49**	**11.67**	**38.00**	**12.33**	**35.96**	**19.06**
InternVL3-78B	1.50	3.67	6.68	3.72	0.44	3.50
Qwen2.5-VL-72B-Instruct	0.83	4.17	6.03	6.64	10.09	4.96
GLM-4.5V	0.33	7.33	4.00	4.50	5.70	4.09

on Function Classification. This performance gap confirms that while closed-source models have mastered visual perception, open-source models face fundamental visual processing limitations.

Analysis We examined performance of models across Analysis subcategories. Table 4 reveals that Gemini-2.5-Pro dominates across all categories (13-90%), followed by GPT-4o and Claude-Sonnet-4 (6-73%), while open-source models struggle significantly (below 40%). Furtheremore, models achieve higher accuracy on traditionally complex tasks like Noise & Jitter Analysis (up to 90%) and Power & Energy Analysis (up to 87%) compared to fundamental tasks like Transient Response (3-13%) and Transfer Function Analysis (8-38%). This counterintuitive result occurs because our synthetic problems are concentrated in these two fundamental subcategories, exposing the critical gap between memorized textbook solutions and genuine mathematical understanding. When models cannot rely on pattern matching from training data and must derive equations from novel circuits, their performance collapses dramatically.

We assessed 5,020 synthetic circuits requiring direct equation derivation without any answer choices. To evaluate symbolic equations, we implement a rigorous symbolic comparison pipeline using SymPy [10] that performs: (1) parsing both predicted and ground-truth equations into symbolic expression trees, (2) algebraic simplification, (3) verification through symbolic subtraction, and (4) numerical validation by evaluating both expressions at 100 random complex frequency points when symbolic comparison is computationally intractable. This multi-pronged approach ensures robust evaluation even when models produce correct but differently formatted equations. Table 5 shows that on synthetic circuits requiring equation derivation, it catastrophically fails at 19.06%. Other models show even steeper degradation: Claude-Sonnet-4 falls to 6.29%, while open-source models barely exceed 4% on synthetic problems. This systematic collapse confirms that models rely on answer elimination and pattern matching rather than mathematical reasoning.

Design Task Table 3 reveals a clear hierarchical pattern in design capabilities across all models. Models demonstrate significantly stronger performance at block-level design (30.91-67.27%) compared to schematic-level design (7.01-36.38%), with hierarchical design falling between these extremes. This pattern indicates that models can more readily manipulate abstract functional blocks than translate specifications into detailed component-level implementations. Notably, Gemini-2.5-Pro, which demonstrated superior symbolic equation derivation capabilities in the Analysis tasks, also dominates the Design tasks with 36.38% schematic-level, 67.27% block-level, and 51.35% hierarchical design accuracy. This correlation between symbolic reasoning and design performance suggests that equation derivation capability serves as a fundamental prerequisite for circuit synthesis.

4 Conclusion

We introduce CircuitSense, a comprehensive benchmark of 8,006 problems for evaluating visual-to-mathematical reasoning in circuit understanding which combines curated questions with synthetic problems focused on symbolic equation derivation. Our hierarchical synthetic generation pipeline produces novel circuits across six levels with guaranteed ground-truth symbolic equations, enabling rigorous evaluation. Our extensive evaluation on perception, analysis, and design tasks shows that models demonstrate adequate perception (85%+ for closed-source) but fail catastrophically at mathematical symbolic modeling (below 19%). This mathematical weakness directly undermines their design capabilities.

References

[1] P. E. Allen and Douglas R. Holberg. *CMOS Analog Circuit Design*. The Oxford Series in Electrical and Computer Engineering. Oxford University Press, USA, 3rd ed edition.

[2] Erik Bruun. CMOS Analog IC Design: Problems and Solutions.

[3] Jian Gao, Weidong Cao, Junyi Yang, and Xuan Zhang. Analoggenie: A generative engine for automatic discovery of analog circuit topologies, 2025.

[4] Google DeepMind. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025.

[5] Paul R. Gray, editor. *Analysis and Design of Analog Integrated Circuits*. Wiley, 5th ed edition.

[6] Pavan Kumar Hanumolu, Merrick Brownlee, Kartikeya Mayaram, and Un-Ku Moon. Analysis of charge-pump phase-locked loops. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 51(9):1665–1674, 2004.

[7] Michael Hayes. Lcapy: symbolic linear circuit analysis with Python. *PeerJ Computer Science*, page e875, February 2022.

[8] Jiaqi Liu, Songning Lai, Pengze Li, Di Yu, Wenjie Zhou, Yiyang Zhou, Peng Xia, Zijun Wang, Xi Chen, Shixiang Tang, Lei Bai, Wanli Ouyang, Mingyu Ding, Huaxiu Yao, and Aoran Wang. Mimicking the physicist's eye:a vlm-centric approach for physics formula discovery, 2025.

[9] Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning, 2021.

[10] Aaron Meurer, Christopher P. Smith, Mateusz Paprocki, Ondřej Čertík, Sergey B. Kirpichev, Matthew Rocklin, Amit Kumar, Sergiu Ivanov, Jason K. Moore, Sartaj Singh, Thilina Rathnayake, Sean Vig, Brian E. Granger, Richard P. Muller, Francesco Bonazzi, Harsh Gupta, Shivam Vats, Fredrik Johansson, Fabian Pedregosa, Matthew J. Curry, Andy R. Terrel, Štěpán Roučka, Ashutosh Saboo, Isuru Fernando, Sumith Kulal, Robert Cimrman, and Anthony Scopatz. Sympy: symbolic computing in python. *PeerJ Computer Science*, 3:e103, January 2017.

[11] ngspice Development Team. ngspice circuit simulator. `http://ngspice.sourceforge.net`, 2024. Version 43.

[12] Yicheng Pan, Zhenrong Zhang, Pengfei Hu, Jiefeng Ma, Jun Du, Jianshu Zhang, Quan Liu, Jianqing Gao, and Feng Ma. Enhancing the geometric problem-solving ability of multimodal llms via symbolic-neural integration, 2025.

[13] Mehdi Rahmani-Andebili. *Advanced Electrical Circuit Analysis: Practice Problems, Methods, and Solutions*. Springer International Publishing.

[14] Behzad Razavi. *Design of Analog CMOS Integrated Circuits*. McGraw-Hill Education, second edition edition.

[15] Md. Abdus Salam and Quazi Mehbubar Rahman. *Fundamentals of Electrical Circuit Analysis*. Springer Singapore.

[16] Yichen Shi, Ze Zhang, Hongyang Wang, Zhuofu Tao, Zhongyi Li, Bingyu Chen, Yaxin Wang, Zhiping Yu, Ting-Jung Lin, and Lei He. Amsbench: A comprehensive benchmark for evaluating mllm capabilities in ams circuits, 2025.

[17] Lejla Skelic, Yan Xu, Matthew Cox, Wenjie Lu, Tao Yu, and Ruonan Han. Circuit: A benchmark for circuit interpretation and reasoning capabilities of llms, 2025.

[18] SkyWater Technology Foundry. SkyWater SKY130 Open Source Process Design Kit. `https://github.com/google/skywater-pdk`, 2020. 130nm CMOS Technology.

[19] Xiaomeng Yang, Jian Gao, Yanzhi Wang, and Xuan Zhang. Zerosim: Zero-shot analog circuit evaluation with unified transformer embeddings. In *Proceedings of the 44th IEEE/ACM International Conference on Computer-Aided Design*, pages 1–9, 2025.

[20] Erle Zhu, Yadi Liu, Zhe Zhang, Xujun Li, JinZhou, Xinjie Yu, Minlie Huang, and Hongning Wang. MAPS: Advancing multi-modal reasoning in expert-level physical science. In *The Thirteenth International Conference on Learning Representations*, 2025.

A Appendix

A.1 Data Collection

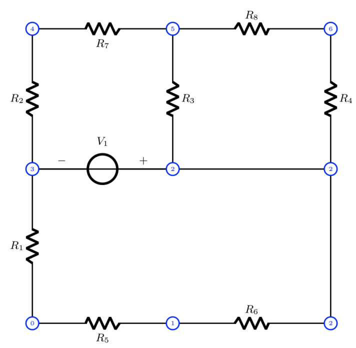
We gather 2,986 curated problems for from authoritative sources to ensure broad topical coverage across CMOS analog, RLC network analysis, and system-level circuit design. For Analysis task, our collection drew from two primary categories: (1) canonical textbooks widely adopted in undergraduate and graduate curricula including [5, 14, 1, 2, 13, 15]; (2) university course repositories, including University of Toronto ECE331 (*Analog Electronics*), Georgia Tech ECE6412 (*Analog Integrated Circuit Design*), and Georgia Tech ECE3050 (*Analog Electronics*). For Perception questions we used a subset of circuit images from AnalogGenie [3] and our hierarchical synthetic generation pipeline. For Design task, we collected data from canonical analog circuit design textbooks such as [5, 14, 1, 2], along with representative problem sets curated from university courses and design problems from ZeroSim [19].

However, curated problems suffer from potential dataset contamination and rarely test equation derivation systematically. To ensure unbiased evaluation, we developed a hierarchical synthetic generation pipeline producing novel circuits with guaranteed ground-truth equations across different hierarchy levels, detailed in Section 2.1.

A.2 Synthetic Examples

Synthetic Example Q1: Nodal Equation

Question: Derive the nodal equation for node 2 in the s-domain. Express the equation using only the circuit elements and their values as labeled in the diagram. Make sure the final answer is just the symbolic equation Vn2(s) = ..., where the right side contains only the labeled components and sources from the circuit diagram.



Corresponding Netlist:

```
R5 1 0 R5
R1 0 3 R1
R6 1 2 R6
V1 2 3 V1
R2 3 4 R2
R3 5 2 R3
R4 6 2 R4
R7 5 4 R7
R8 5 6 R8
```

Ground-truth:

$$Vn2(s) = V1 * (R5 + R6)/(s * (R1 + R5 + R6))$$

Claude-Sonnet-4:

$$Vn2(s) = V3 + V1$$

Gemini-2.5-Pro:

$$Vn2(s) = V1 * (R5 + R6)/(R1 + R5 + R6)$$

GPT-4o:

$$V_2(s) = \frac{\frac{V_5(s)}{R3} + \frac{V_6(s)}{R4} + \frac{V_1(s)}{R6} - \frac{V1}{R2}}{\frac{1}{R3} + \frac{1}{R4} + \frac{1}{R6}}$$

InternVL3-78B:

$$Vn2(s) = (V5(s)/R3 + V6(s)/R4 + V1(s)/R6 + V3(s)/R8)/(1/R3 + 1/R4 + 1/R6 + 1/R8)$$

Qwen2.5-VL-72B-Instruct:

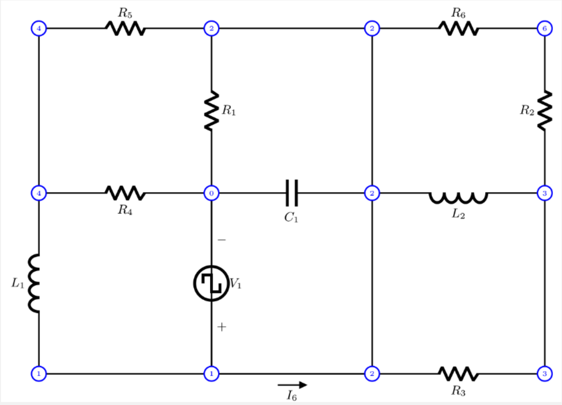
$$Vn2(s) = \frac{\frac{Vn1(s)}{R6} + \frac{Vn3(s)}{R2}}{\frac{1}{R3} + \frac{1}{R4} + \frac{1}{R6} + \frac{1}{R2}}$$

GLM-4.5V:

$$V_{n2}(s) = \frac{\frac{V_{n1}(s)}{R_6} + \frac{V_{n4}(s)}{R_2} + \frac{V_{n5}(s)}{R_3} + \frac{V_{n6}(s)}{R_4} + V_1(s)\left(\frac{1}{R_1} + \frac{1}{R_2}\right)}{\frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \frac{1}{R_4} + \frac{1}{R_6}}$$

Synthetic Example Q2: RLC Transfer Function

Question: Derive the nodal equation for node 3 in the s-domain. Express the equation using only the circuit elements and their values as labeled in the diagram. Make sure the final answer is just the symbolic equation Vn3(s) = ..., where the right side contains only the labeled components and sources from the circuit diagram.



Netlist Extraction:

```
L1 1 4 L1
V1 1 0 step
R3 3 2 R3
R4 0 4 R4
C1 2 0 C1
R1 2 0 R1
L2 3 2 L2
R2 3 6 R2
R5 4 2 R5
R6 2 6 R6
```

Ground-turth: Vn3(s) = R1*R4*V1/(s*(C1*(R1*(L1*R4*s**2 + L1*R5*s**2 + R4*R5*s)) + L1* (R1*s + R4*s + R5*s) + R4*(R1 + R5))

Claude-Sonnet-4:

$$V3(s) = (V6(s) * s * L2 * R3 + V2(s) * R2 * (R3 + s * L2))/(s * L2 * (R2 + R3) + R2 * R3)$$

Gemini-2.5-Pro:

$$Vn3(s) = V1(s)$$

GPT-4o:

$$V_3(s) = \frac{\frac{V_6(s)}{R_2} + \left(\frac{1}{sL_2} + \frac{1}{R_3}\right) V_2(s)}{\frac{1}{R_2} + \frac{1}{sL_2} + \frac{1}{R_3}}$$

InternVL3-78B:

$$Vn3(s) = (V6(s) * (1/R2) + V2(s) * (1/R3 + 1/(s * L2)))/(1/R2 + 1/R3 + 1/(s * L2))$$

Qwen2.5-VL-72B-Instruct:

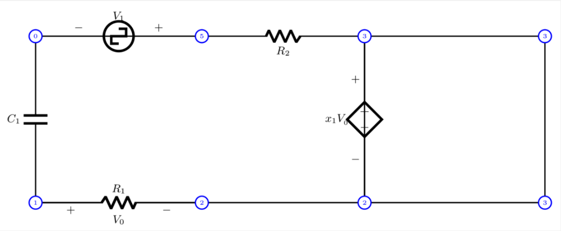
$$Vn3(s) = (Vn2(s)/R2 + Vn1(s)/R3)/(1/R2 + 1/R3 + 1/(s * L2))$$

GLM-4.5V:

$$V_{n3}(s) = sL_2R_3V_{n6} + R_2(sL_2 + R_3)V_{n2}sL_2(R_2 + R_3) + R_2R_3$$

Synthetic Example Q3: Small Signal (Dependent Sources)

Question: What is the transfer function from V1 to R1 in this circuit?



Netlist Extraction:

```
R1 1 2 R1
C1 1 0 C1
E1 3 2 1 2 x_1 0
V1 5 0 step
R2 3 5 R2
```

Ground-truth:

$H(s) = ((R1 * s/(R1 * x_1 - R1 - R2))/(s - 1/(C1 * R1 * x_1 - C1 * R1 - C1 * R2))) * 1$

Claude-Sonnet-4:

$$H(s) = (R1 * s * C1)/((1 + x1) * R1 * s * C1 + 1)$$

Gemini-2.5-Pro:

$$H(s) = (-s * R1 * C1)/(1 + s * C1 * (R1 * (1 - x1) + R2))$$

GPT-4o:

$$H(s) = \frac{sC_1}{sC_1 + \frac{x_1}{R_1}}$$

InternVL3-78B:

$$H(s) = 1/(1 + s * C1 * R1 * (1 - x1))$$

Qwen2.5-VL-78B-Instruct:

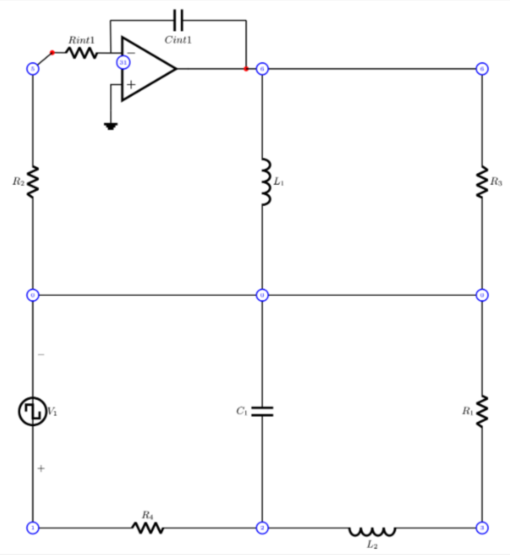
$$H(s) = \frac{\frac{1}{R_2} + V_2\left(\frac{1}{R_1} - \frac{1}{R_2}\right)}{\frac{1}{R_1} - s * C_1 - \frac{x_1}{R_2}}$$

GLM-4.5V:

$$H(s) = \frac{sC_1R_1}{sC_1(R_2 + R_1(x_1 - 1)) + 1}$$

Synthetic Example Q5: Ideal Op-amp

Question: Derive the nodal equation for node 3 in the s-domain. Express the equation using only the circuit elements and their values as labeled in the diagram. Make sure the final answer is just the symbolic equation Vn3(s) = ..., where the right side contains only the labeled components and sources from the circuit diagram.



Netlist Extraction:

```
R4 1 2 R4
V1 1 0 step
L2 3 2 L2
C1 2 0 C1
R1 3 0 R1
R2 0 5 R2
L1 6 0 L1
R3 6 0 R3
Rint1 5 31 Rint1
Cint1 6 31 Cint1
Eint1 6 0 0 31 Ad 0
```

Ground-truth:

$$Vn3(s) = R1 * V1/(s * (C1 * L2 * R4 * s * *2 + C1 * R1 * R4 * s + L2 * s + R1 + R4))$$

Gemini-2.5-Pro:

$$Vn3(s) = (R1 * Vn2(s) + s * L2 * Vn7(s))/(R1 + s * L2)$$

Claude-Sonnet-4:

$$Vn3(s) = (V2(s) * R1 * s * L2 + V5(s) * R4 * R1)/(R1 * R4 + s * L2 * (R1 + R4))$$

GPT-4o:

$$V_3(s) = \left(\frac{V_0(s)}{R1} + \frac{V_2(s)}{sL2} + \frac{V_2(s)}{R4}\right)\left(\frac{1}{R1} + \frac{1}{sL2} + \frac{1}{R4}\right)^{-1}$$

Qwen2.5-VL-72B:

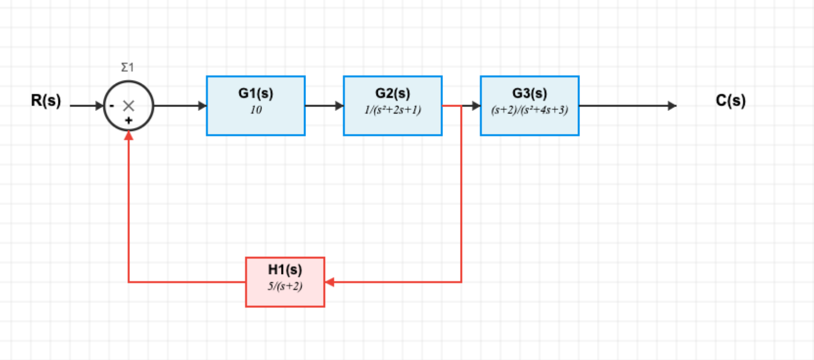
$$Vn3(s) = Vn2(s)/(1 + R1/(s * L2) + R1/R3)$$

GLM4.5V:

$$V_{n3}(s) = R_1V_1(R_1 + R_4) + s(L_2 + C_1R_4R_1) + s^2(C_1R_4L_2)$$

Synthetic Example Q6: Enhanced Control System

Question: What is the Transfer function of the provided block diagram?



Ground-truth:

$$\frac{C(s)}{R(s)} = \frac{(\frac{-10}{(s^2+2s+1)})}{(1 - \frac{50}{(s^2+2s+1).(s+2))}}$$

Claude-Sonnet-4:

$$C(s)/R(s) = 10(s+2)/[(s+1)^3(s+2)^2(s+3)+50]$$

Gemini-2.5-Pro:

$$C(s)/R(s) = (10s^2+40s+40)/(s^5+8s^4+24s^3+84s^2+223s+156)$$

GPT-4o:

$$C(s)/R(s) = \frac{10(s+2)}{s^6+2s^5+s^4+2s^3+s^2+6s+54}$$

InternVL3-78B:

$$C(s)/R(s) = \frac{20(s+2)}{(s^2+2s+1)(s+3)(s+4)+100s(s+2)/(s+6)}$$

Qwen2.5-VL-72B:

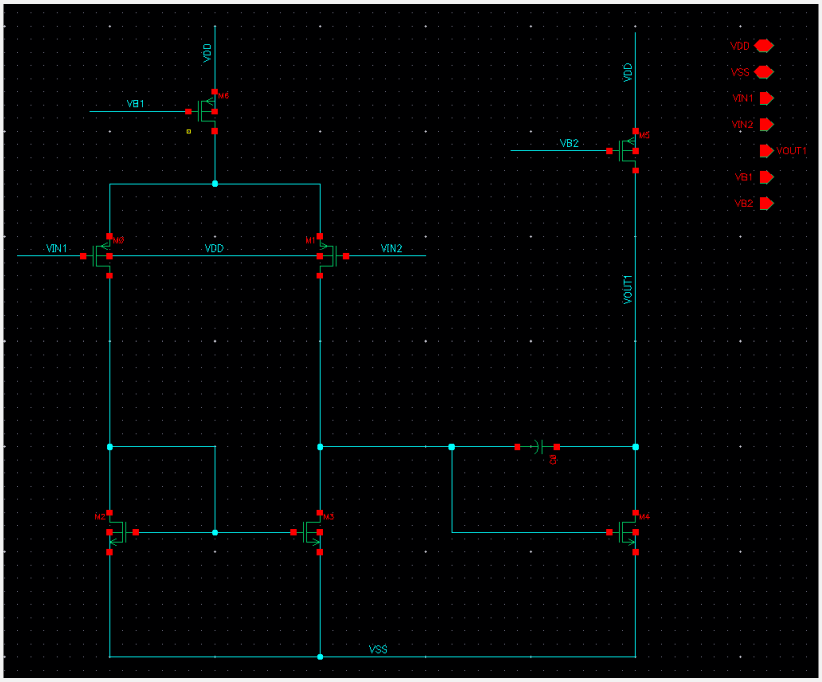
$$\frac{C(s)}{R(s)} = \frac{10(s+2)}{(s^2+2s+1)(s+1)(s+3)+50}$$

GLM-4.5V:

$$C(s)/R(s) = \frac{10(s+2)}{(s+1)^3(s+2)(s+3)+50}$$

Design Example Q6: Simulation-needed Schematic-level question

Question: Design the sizing and biasing voltage of an Op-Amp in SKY130nm (VDD 1.8 V) as shown in the provided circuit image.

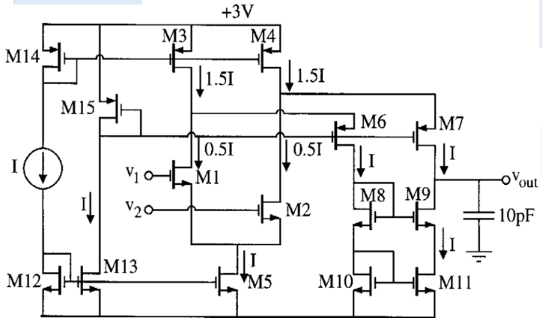


A.3 Curated Problems

Question:

This problem deals with the op amp shown in the provided circuit image. All device lengths are $1\,\mu m$, the slew rate is $\pm 10\,V/\mu s$, the GB is $10MHz$, the maximum output voltage is $+2V$, the minimum output is $-2V$, and the input common mode range is from $-1V$ to $+2V$. Design all W value of all transistors in this op amp. Your design must meet or exceed the specifications. Ignore bulk effects in this problem.

Image:



Source:

ECE 6412-Spring 2003-HW8

Text Answer: $W_{15} = 4\mu m$

$W_1 = W_2 = 36\mu m \quad W_{14} = 16\mu m$

$W_3 = W_4 = W_6 = W_7 = 24\mu m$

$W_8 = W_9 = W_{10} = W_{11} = 121\mu m$

$W_{12} = W_{13} = W_5 = 1.4\mu m$

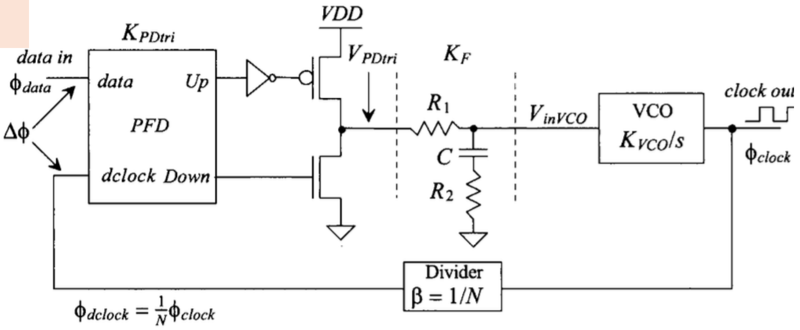
Level: Level 2

Example 1: schematic-level design

Question:

Design a DPLL using the tri-state topology seen in the provided circuit image that generates a clock signal at a frequency of $100MHz$ from a 50 MHz square wave input. This application of the DPLL is called frequency synthesis.

Image:



Source: CMOS Circuit Design, Layout, and Simulation
P577
Example19.4

Level: Level 2

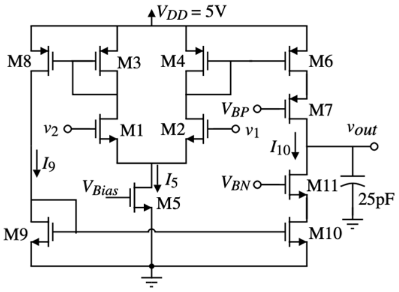
Text Answer: $C = 10pF, R_2 = 20k\Omega, R_1 = 42.5k\Omega$

Example 2: block-level design

Question:

Design a CMOS operational amplifier powered from a single $5V$ supply in which all MOSFET channel lengths are fixed at $L = 1\mu m$ and every device operates in saturation; choose the width W of every transistor so that the amplifier meets or exceeds the following specifications: slew rate = $\pm 10V/\mu s$ maximum and minimum output voltage $V_{out(max)} = 4V$, $V_{out(min)} = 1V$, input common-mode range $V_{IC(min)} = 1.5V$ to $V_{IC(max)} = 4V$. And unity-gain bandwidth(GB) =10 MHz; ignore bulk/body effects. Provide a summary table (round each to the nearest micron) listing the W of every transistor.

Image:



Level: Level 2

Source: ECE 6412-Spring 2005- HW07

Text Answer:

$W_1 = 90\mu m,$
$W_3 = W_4 = W_6 = W_7 = W_8 = 40\mu m,$
$W_9 = W_{10} = W_{11} = 18\mu m,$
$I_5 = 250\mu A$

Example 3: hierarchical design

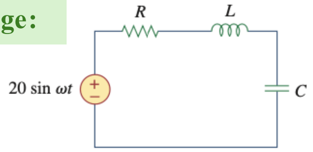
Question:

In the circuit, R=2Ω , L=1mH, and C=0.4μF. Find the resonant frequency.

Level: Level 0.5

Source: Fundamentals of Electric Circuits
P674
Example 14.7

Image:



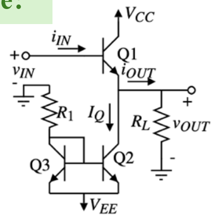
Text Answer: ω_0=50 $krad/s$

Example 4: Level 0.5

Question:

For the emitter follower output stage shown below, find the value of efficiency when $R_I = \frac{-VEE-VBE}{IQ} = 7.826\text{K}\Omega$ and $V_{CC} = -VEE = 2.5V, VCE = 0.2V, VBE = 0.7V, RL = 10K\Omega$

Image:



Level: Level 2

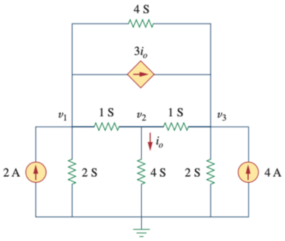
Source: ECE 6412-Spring 2004-Homework02

Text Answer: 23 %

Example 5: Level 2

Question: Use nodal analysis to determine voltages V_2 in the circuit.

Image:



Level: Level 1.5

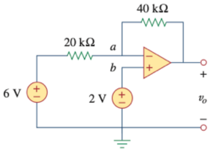
Source: Fundamentals of Electric Circuits P118 Example3.27

Text Answer: 375 mV

Example 6: Level 1.5

Question: Determine v_0 in the op amp circuit

Image:



Level: Level 1

Source: Fundamentals of Electric Circuits P674 Example 5.4

Text Answer: -6v

Example 7: Level 1

A.4 Prompt Templates

Prompt Template for Circuit Schematic Synthetic Pipeline

You are an expert electrical engineer specializing in circuit analysis. Analyze the circuit diagram and solve for the requested symbolic expression.
Task: {Main Question}
Instructions: 1. Use EXACT component labels as shown in the circuit (e.g., R1, R2, C1, C2, L1, not generic R, C, L) 2. For Laplace domain, use lowercase 's' as the complex frequency variable 3. Use standard impedances: R for resistors, 1/(sC) for capacitors, sL for inductors 4. For op-amps: Apply virtual short (V+ = V-) if in negative feedback, use Ad for gain if specified
Response Format: You MUST structure your response exactly as follows:
`<think>`
[Show your reasoning and intermediate steps here] - Identify components and nodes - Intermediate steps - Show equations - Show algebraic manipulation - Any simplification steps
`</think>`
`<answer>`
[Only the final symbolic equation here, e.g., H(s) = ..., Vn1(s) = ..., etc.]
`</answer>`
Make sure to use standard mathematical notation with for multiplication, / for division, and f̂or powers."