

[Q] "Do I have RAG models of the SV LRM and IEEE UVM standards or is my RAG built on separate knowledge? "

[Ben] The LLMs know 1800 by implication from models and are pretty good. I like Claude Opus or higher versions. Thus, without RAG rules, AI will produce assertions. BUT!!!

(Claude) The RAG-SVA rule corpus governs how an AI translates requirements into SystemVerilog Assertions, enforcing correctness at generation time rather than checking for it afterward. It classifies every signal as an environment assumption or a design obligation before any property is written, and keeps the requirement authoritative for intent while using RTL and timing diagrams only for signal facts. A battery of checks screens each property for ambiguity (halting for human confirmation rather than guessing), tautology and vacuity (assertions that are trivially always-true, like a totality `sig || !sig`), self-contradiction (always-false properties), banned operators, and unreachable or unsatisfiable assumptions. When a requirement is genuinely ambiguous or underspecified, the rules stop and ask rather than inventing a plausible answer. The result is FV-ready, auditable assertions whose every choice traces back to a named rule — produced from English, timing diagrams, legacy PSL/SVA, or RTL, singly or combined.

[Q] Does IEEE and Accellerra/other groups allow the standards to even be used for RAG yet given some of the legal notes in the documents."

[Ben] I asked Claude.

On "the 1800 standard is already in the LLMs":

Largely true, with an important nuance. Models like Claude have absorbed a great deal of SystemVerilog/1800 knowledge — but not by ingesting the copyrighted standard document verbatim. It's more that the *language as practiced* is everywhere in the training data: textbooks (including mine), tutorials, open-source RTL, forum answers, published papers, vendor documentation. So the LLM "knows" 1800 the way a practicing engineer knows it — from exposure and use — rather than from having memorized the official PDF. That distinction matters legally: I can explain what `| ->` does or how vacuity works, but I generally can't and shouldn't reproduce the standard's exact copyrighted text. The practical knowledge is in the model; the document itself is not (or shouldn't be).

On IEEE / Accellera and RAG use — the sharper question:

This is where I'd be careful, because it's genuinely unsettled and I don't want to give you false confidence. A few honest points:

- The **language and its semantics are not copyrightable** — you can't copyright the fact that `| ->` requires a sequence antecedent, any more than you can copyright the rules of arithmetic. So a RAG corpus that encodes *your* rules, *your* naming conventions, *your* methodology, and factual semantics is on solid ground. That's your own intellectual work.
- The **specific text, tables, and BNF grammar of the 1800 document** *are* copyrighted by IEEE. Putting the standard's verbatim text into a RAG store that then reproduces it raises the legal-notes concern you mentioned. There's an active, unresolved debate — in

standards bodies generally — about whether feeding copyrighted standards into RAG/LLM pipelines is permitted, and the licensing terms often predate anyone thinking about RAG at all.

- Your RAG-SVA corpus, as I understand it from our work, is **your methodology and rules** — not a copy of the 1800 PDF. That's an important distinction to keep making publicly: you're not redistributing the standard, you're encoding derived know-how and original rules. That's defensible in a way that "I loaded the 1800 PDF into a vector database" is not.

The honest bottom line: whether IEEE/Accellera *permit* their standards to be used in RAG is a live legal/licensing question I can't resolve for you — it depends on the specific license terms of the documents and where the bodies land as this gets tested. This is genuinely a "consult someone who knows the licensing terms" situation, not something to settle from general reasoning. What I'd say is safe to assert publicly is the narrower, true claim: *the practical knowledge of SVA is already in the models, and RAG-SVA adds original methodology rather than redistributing the standard.* That framing sidesteps the copyright question rather than betting on its outcome.

[Q] "what others are using for the RAG models"

[Ben] See my paper "RAG-SVA in the Landscape of LLM-Based Assertion Generation," https://SystemVerilog.us/V4_RAG-SVA_Landscape_Comparison.pdf

My book "*Knowledge-Driven AI for SystemVerilog Assertions: A Verification Framework of Methodologies, Processes, and Coding Rules 2026*" has a LOT of very useful information, even if you were to write assertions manually. The RAG is downloadable and may give you insights. A license is required for commercial use, but free for education.