SEMICONDUCTOR ENGINEERING TRAINING

Formal Verification Training

Six-Day Course Summary

A practical 24-hour programme covering SystemVerilog Assertions, property proof, debug, formal coverage, block-level signoff, advanced formal applications and AI-assisted formal verification.

6	10	9	2
days	formal apps	worked designs	AI tool demos

COURSE INSTRUCTOR

Mike Bartley

CEO and Founder, Alpinum Consulting Ltd.

alpinumconsulting.com/formal-verification-training/

Course Summary

Alpinum's Formal Verification Training helps engineers understand how formal methods can be used to verify hardware designs more exhaustively than simulation alone.

The course introduces practical formal verification concepts, SystemVerilog Assertions, property writing, proof analysis, debug techniques, formal metrics and advanced methods for applying formal verification within real verification projects.

The course is delivered as a six-day programme of four hours a day: 24 hours of teaching in total, building from an introduction to formal verification through to block-level signoff, advanced formal and the use of AI in verification.

The course is taught by Mike Bartley, CEO and Founder of Alpinum Consulting Ltd.

Although the course does not focus on a particular vendor tool, a tool is used to demonstrate and practise the concepts introduced. Example designs are provided, and participants may share suitable designs in advance for possible use during the course.

What the Course Includes

- Ten formal verification application demonstrations, worked on real designs.
- Nine worked design examples, from a synchronous FIFO through to a single-cycle RISC-V processor.
- Two AI tool demonstrations, covering assertion-to-requirement mapping and formal application selection.
- Twenty-four short quizzes - four each day - plus overnight quizzes after Days 1 to 5.
- An equivalent online submission for every worked example, application demonstration and AI tool, allowing participants to complete the same exercise and receive review during the course.

Course Structure at a Glance

DAY	TOPICS COVERED	APPLICATIONS	DESIGN EXAMPLES
Day 1	Introduction to formal verification · Writing basic SystemVerilog Assertions · Constraints and properties	X-Propagation	Synchronous FIFO
Day 2	Properties for formal versus simulation · Proving a basic property · Advanced SVA	Reachability	ALU
Day 3	Local variables and assertion quality · Proving complex properties · Debug and signoff	Formal Coverage Analyzer	Up/Down Counter · Two-Transaction FIFO
Day 4	Full formal verification of a block · SEC vs. LEC · Formal in the design flow · The AHAA model · Formal for designers	Connectivity Checking · Formal Low Power · Formal Testbench Analyzer	SelAB
Day 5	Formal reuse · Formal in context · Design suitability · Advanced formal	Formal Security Verification · Functional Safety	APB4 · RISC-V single-cycle
Day 6	SoC security proofs · Datapath validation and register verification · AI in Verification · AI in Formal	Datapath Validation · Formal Register Verification	Arbiter · Traffic-Light FSM

How Each Day Runs

- Each day opens with a review of the previous overnight quiz and the plan for the day.
- Teaching is delivered in short blocks of 20 to 30 minutes. Each block is followed by a three-question online quiz on the material just covered.
- Worked design examples and application demonstrations are spread through the day, each with an equivalent exercise in the online submission tool.
- Two breaks each day include a short slot to review quiz answers and online submissions before teaching resumes.
- Days 1 to 5 close with a summary and overnight quiz, which is reviewed the following morning. Day 6 closes with a full-course summary.

Times in the daily tables are shown from the start of each four-hour session.

Day 1 - Introduction to Formal Verification and SVA

An introduction to what formal verification is, how it differs from simulation, and how to read and write basic SystemVerilog Assertions.

TIME	SESSION
0:00	**Welcome, logistics and the plan for the day (5 min)**
0:05	**1 Introduction to Formal Verification — part 1 (25 min)** • What formal verification is, and where it fits alongside simulation • Proof versus simulation: what each can and cannot tell you
0:30	**Quiz 1 — why formal, and proof versus simulation (5 min)**
0:35	**Worked example: Synchronous FIFO (20 min)** • Worked through live, with an equivalent online submission
0:55	**1 Introduction to Formal Verification — part 2 (25 min)** • Constraints and properties, and the role of each in a formal run
1:20	**Quiz 2 — properties and constraints (5 min)**
1:25	**Break (20 min)** • Includes a short slot to review quiz answers and online submissions
1:45	**1 Introduction to Formal Verification — part 3 (25 min)** • Assumptions, and how they shape a proof • An introduction to formal coverage
2:10	**Quiz 3 — assumptions and coverage (5 min)**
2:15	**Application demonstration: X-Propagation (20 min)** • Run on a real design, with an equivalent online submission
2:35	**2 Writing Basic SystemVerilog Assertions — part 1 (30 min)** • Introduction to the language • The main combinatorial language constructs (syntax and semantics) • Constraints and properties
3:05	**Quiz 4 — SVA syntax and sequences (5 min)**
3:10	**Break (15 min)** • Includes a short slot to review quiz answers and online submissions

3:25	**2 Writing Basic SystemVerilog Assertions — part 2 (25 min)**
	• Examples of constraints
	  ◦ Reading and understanding
	  ◦ Writing from fresh
	• Examples of properties
	  ◦ Reading and understanding
	  ◦ Writing from fresh

3:50	**Day summary and overnight quiz (10 min)** • The overnight quiz is reviewed at the start of Day 2

Day 2 - Properties, Proving and Advanced SVA

Completing the assertion language, writing properties suited to formal rather than simulation, and proving a first property on a real design.

TIME	SESSION
0:00	**Overnight quiz review and the plan for the day (5 min)**
0:05	**2 Writing Basic SystemVerilog Assertions — completion (25 min)** • Completing the constraint and property examples begun on Day 1 • Writing constraints and properties from fresh
0:30	**Quiz 1 — SVA recap from Day 1 (5 min)**
0:35	**Worked example: ALU (20 min)** • Worked through live, with an equivalent online submission
0:55	**3 Writing properties for formal versus simulation (30 min)** • How a property written for proof differs from a checker written for simulation • Constraining a formal environment versus driving stimulus
1:25	**Quiz 2 — properties for formal versus simulation (5 min)**
1:30	**Break (20 min)** • Includes a short slot to review quiz answers and online submissions
1:50	**4 Proving a basic property (25 min)** • Using a tool to prove a basic property on a real design (e.g. FIFO or a student example design)
2:15	**Quiz 3 — proof results and debug (5 min)**
2:20	**Application demonstration: Reachability (20 min)** • Run on a real design, with an equivalent online submission

2:40	**5 Advanced SVA (30 min)**
	• The main sequential language constructs (syntax and semantics)
	• Examples of sequential constraints
	  • Reading and understanding
	  • Writing from fresh

3:10	**Quiz 4 — advanced SVA operators (5 min)**
3:15	**Break (15 min)** • Includes a short slot to review quiz answers and online submissions

3:30	**6 Writing complex SystemVerilog Assertions (20 min)**
	• Examples of sequential properties
	  • Reading and understanding
	  • Writing from fresh

3:50	**Day summary and overnight quiz (10 min)** • The overnight quiz is reviewed at the start of Day 3

Day 3 - Complex Properties, Debug and Signoff

Writing higher-quality assertions, proving properties that do not converge easily, debugging failures and measuring completeness.

TIME	SESSION
0:00	**Overnight quiz review and the plan for the day (5 min)**
0:05	**7 Local variables, FSM protocol properties and assertion quality (25 min)** • Using local variables in assertions, and when to avoid them • Writing protocol properties for finite state machines • Judging the quality of an assertion set
0:30	**Quiz 1 — assertion quality and local variables (5 min)**
0:35	**Worked example: Up/Down Counter (15 min)** • Worked through live, with an equivalent online submission
0:50	**8 Proving a complex property — part 1 (25 min)** • Using a tool to prove a complex property on a real design (or a student example design) • Interpreting the three possible outcomes: Proved, Failed, Unproven
1:15	**Quiz 2 — complex proof setup (5 min)**
1:20	**Break (20 min)** • Includes a short slot to review quiz answers and online submissions
1:40	**8 Proving a complex property — part 2 (20 min)** • Helping a proof to converge • Introducing abstractions to reduce the proof problem
2:00	**Quiz 3 — convergence and abstraction (5 min)**
2:05	**Worked example: Two-Transaction FIFO (15 min)** • Worked through live, with an equivalent online submission
2:20	**8 Proving a complex property — part 3 (20 min)** • Bringing the techniques together on a complete property set
2:40	**Quiz 4 — debugging a failing property (5 min)**
2:45	**Break (15 min)** • Includes a short slot to review quiz answers and online submissions
3:00	**9 Debug and signoff (20 min)** • Using a tool to debug a failing property on a real design (or a student example design) • Measuring coverage and completeness • Measuring coverage with a particular tool on a real design (or a student example design)
3:20	**Application demonstration: Formal Coverage Analyzer (FCA) (30 min)** • Run on a real design, with an equivalent online submission
3:50	**Day summary and overnight quiz (10 min)** • The overnight quiz is reviewed at the start of Day 4

Day 4 - Block-Level Formal, SEC vs. LEC and Formal in the Design Flow

Verifying a complete block with formal, and understanding where formal fits into a project alongside the rest of the design and verification flow.

TIME	SESSION
0:00	**Overnight quiz review and the plan for the day (5 min)**
0:05	**9 Debug and signoff — completion and summary (25 min)** • Combining formal results into a signoff decision
0:30	**Quiz 1 — signoff metrics (5 min)**
0:35	**Worked example: SelAB (20 min)** • Worked through live, with an equivalent online submission

0:55	**10 Full formal verification of a block (25 min)**
	• Developing constraints for a real design (or a student example design)
	  ◦ Over constraint
	  ◦ Under constraint
	  ◦ Determining a "full" set of properties
	• Run the constraints and properties using the tool on a real design (or a student example design)

1:20	**Quiz 2 — block-level formal setup (5 min)**
1:25	**Break (20 min)** • Includes a short slot to review quiz answers and online submissions
1:45	**11 SEC vs. LEC, and formal in the design flow (25 min)** • Sequential equivalence checking compared with logical equivalence checking • Where each fits in the design and verification flow, and what each can and cannot prove • What are the main formal verification apps (e.g. X propagation) and why are they useful • Running apps on a real design (or a student example design)
2:10	**Quiz 3 — SEC vs. LEC (5 min)**
2:15	**Application demonstration: Connectivity Checking (CC) (20 min)** • Run on a real design, with an equivalent online submission

2:35	**12 The AHAA model, and formal for designers (25 min)**
	• Understanding the various applications of formal verification
	  ◦ Bug avoidance
	  ◦ Bug hunting
	  ◦ Bug absence
	  ◦ Bug analysis
	• Understanding when and where to apply formal during a project
	• Designer bring-up: design visualisation, advanced lint, using apps, reachability
	• Possible reuse by verification

3:00	**Quiz 4 — the AHAA model (5 min)**
3:05	**Break (15 min)** • Includes a short slot to review quiz answers and online submissions
3:20	**Application demonstrations: Formal Low Power (FLP) and Formal Testbench Analyzer (FTA) (25 min)** • Run on a real design, each with an equivalent online submission
3:45	**Day summary and overnight quiz (15 min)** • The overnight quiz is reviewed at the start of Day 5

Day 5 - Formal Reuse, Formal in Context and Advanced Formal

Reusing assertions across static and dynamic verification, placing formal within a verification plan, and applying techniques when a property will not converge.

TIME	SESSION
0:00	**Overnight quiz review and the plan for the day (5 min)**
0:05	**12 The AHAA model and formal for designers — completion and summary (25 min)**
0:30	**Quiz 1 — the AHAA model and formal for designers (5 min)**
0:35	**Worked example: APB4 (20 min)** • Worked through live, with an equivalent online submission
0:55	**13 Formal reuse (25 min)** • Reuse of assertions between dynamic and static verifications • Running simulations with our formal assertions added on a real design (or a student example design) • Reuse in assume/guarantee relationships • Assertions as VIP
1:20	**Quiz 2 — reuse of properties and constraints (5 min)**
1:25	**Break (20 min)** • Includes a short slot to review quiz answers and online submissions
1:45	**14 Formal in context (25 min)** • Formal as part of a verification plan • Combining formal results with other activities for a signoff decision
2:10	**Quiz 3 — formal in the verification flow (5 min)**
2:15	**Worked example: RISC-V single-cycle processor (30 min)** • Worked through live, with an equivalent online submission
2:45	**15 Design suitability and advanced formal (20 min)** • What type of design and size of design is suitable? • What makes some properties hard to prove, and why • Interpreting Unproven results • Simple techniques for resolving Unprovens: reducing the input space, reducing data widths, adding constraints • Advanced techniques: abstractions, cut points, free variables, undriven signals, black boxes and models • Selecting the right technique for the right property
3:05	**Quiz 4 — design type and size suitability (5 min)**
3:10	**Break (15 min)** • Includes a short slot to review quiz answers and online submissions
3:25	**Application demonstrations: Formal Security Verification (FSV) and Functional Safety (FuSa) (25 min)** • Run on a real design, each with an equivalent online submission
3:50	**Day summary and overnight quiz (10 min)** • The overnight quiz is reviewed at the start of Day 6

Day 6 - AI in Verification and Formal

Two further formal applications, followed by how AI is being applied across verification and to formal verification in particular, with live demonstrations of AI-assisted formal tools.

TIME	SESSION
0:00	**Overnight quiz review and the plan for the day (5 min)**
0:05	**16 SoC security proofs and automation (25 min)** • Completing the security and safety material from Day 5 • Automating formal runs at SoC level
0:30	**Quiz 1 — security proofs and automation (5 min)**
0:35	**Worked example: Arbiter (15 min)** • Worked through live, with an equivalent online submission
0:50	**17 Datapath Validation (DPV) and Formal Register Verification (FRV) (25 min)** • Proving datapath equivalence without simulating every data value • Proving a register map against its specification
1:15	**Quiz 2 — DPV and FRV (5 min)**
1:20	**Application demonstrations: DPV and FRV (20 min)** • Run on a real design, each with an equivalent online submission
1:40	**Break (20 min)** • Includes a short slot to review quiz answers and online submissions
2:00	**18 AI in Verification (25 min)** • Where AI is being used across the verification flow today • What it is good at, and where it still needs an engineer in the loop
2:25	**Quiz 3 — AI in verification (5 min)**
2:30	**Worked example: Traffic-Light FSM (20 min)** • Worked through live, with an equivalent online submission
2:50	**19 AI in Formal (20 min)** • Applying AI to property writing, proof convergence and formal debug
3:10	**Quiz 4 — AI in formal (5 min)**
3:15	**Break (15 min)** • Includes a short slot to review quiz answers and online submissions
3:30	**AI tool demonstrations: SVA-to-Requirements Mapping and the Formal Applications Advisor (20 min)** • SVA-to-Requirements Mapping: mapping assertions back to design requirements, walked through end to end • Formal Applications Advisor: guidance on which formal application to reach for on a given design problem • Each with an equivalent online submission
3:50	**Day 6 and full course summary (10 min)**

Course Reference

A. Formal Verification Application Demonstrations

Ten applications, approximately 90 minutes in total. Each has an equivalent online submission.

FORMAL VERIFICATION APPLICATION	EST. TIME	DAY
X-Propagation	10 min	Day 1
Reachability	10 min	Day 2
Formal Coverage Analyzer (FCA)	15 min	Day 3
Connectivity Checking (CC)	10 min	Day 4
Formal Low Power (FLP)	05 min	Day 4
Formal Testbench Analyzer (FTA)	10 min	Day 4
Formal Security Verification (FSV)	05 min	Day 5
Functional Safety (FuSa)	05 min	Day 5
Datapath Validation (DPV)	10 min	Day 6
Formal Register Verification (FRV)	10 min	Day 6

B. Worked Design Examples

Nine designs, approximately 80 minutes in total. Each has an equivalent online submission.

DESIGN EXAMPLE	EST. TIME	DAY
Synchronous FIFO	10 min	Day 1
ALU	10 min	Day 2
Up/Down Counter	10 min	Day 3
Two-Transaction FIFO	05 min	Day 3
SelAB	10 min	Day 4
APB4	10 min	Day 5
RISC-V single-cycle processor	10 min	Day 5
Arbiter	05 min	Day 6
Traffic-Light FSM	10 min	Day 6

Course Reference - Continued

C. AI Tool Demonstrations

AI TOOL DEMONSTRATION	EST. TIME	DAY
SVA-to-Requirements Mapping Tool	10 min	Day 6
Formal Applications Advisor	10 min	Day 6

D. Online Submissions

For every worked example, application demonstration and AI tool, participants submit the same exercise through the online tool. Submissions are reviewed at the next break or review slot so each demonstration is closed out before the course moves on. Across the six days, this accounts for around 70 minutes of hands-on time.

E. Quizzes

- Four short quizzes each day, three questions each, taken immediately after the module they cover and delivered as online quizzes - 24 in total.
- An overnight quiz after Days 1 to 5, reviewed at the start of the following day.
- Quiz answers and online submissions are reviewed in the short slot attached to each break.
- Quiz topics across the six days: why formal, and proof versus simulation; properties and constraints; assumptions and coverage; SVA syntax and sequences; SVA recap; properties for formal versus simulation; proof results and debug; advanced SVA operators; assertion quality and local variables; complex proof setup; convergence and abstraction; debugging a failing property; signoff metrics; block-level formal setup; SEC vs. LEC; the AHAA model; the AHAA model and formal for designers; reuse of properties and constraints; formal in the verification flow; design type and size suitability; security proofs and automation; DPV and FRV; AI in verification; AI in formal.

Enquire About Formal Verification Training

Alpinum can deliver Formal Verification Training for individual engineers, verification teams, graduate engineers and organisations building internal formal verification capability.

Visit the Formal Verification Training page