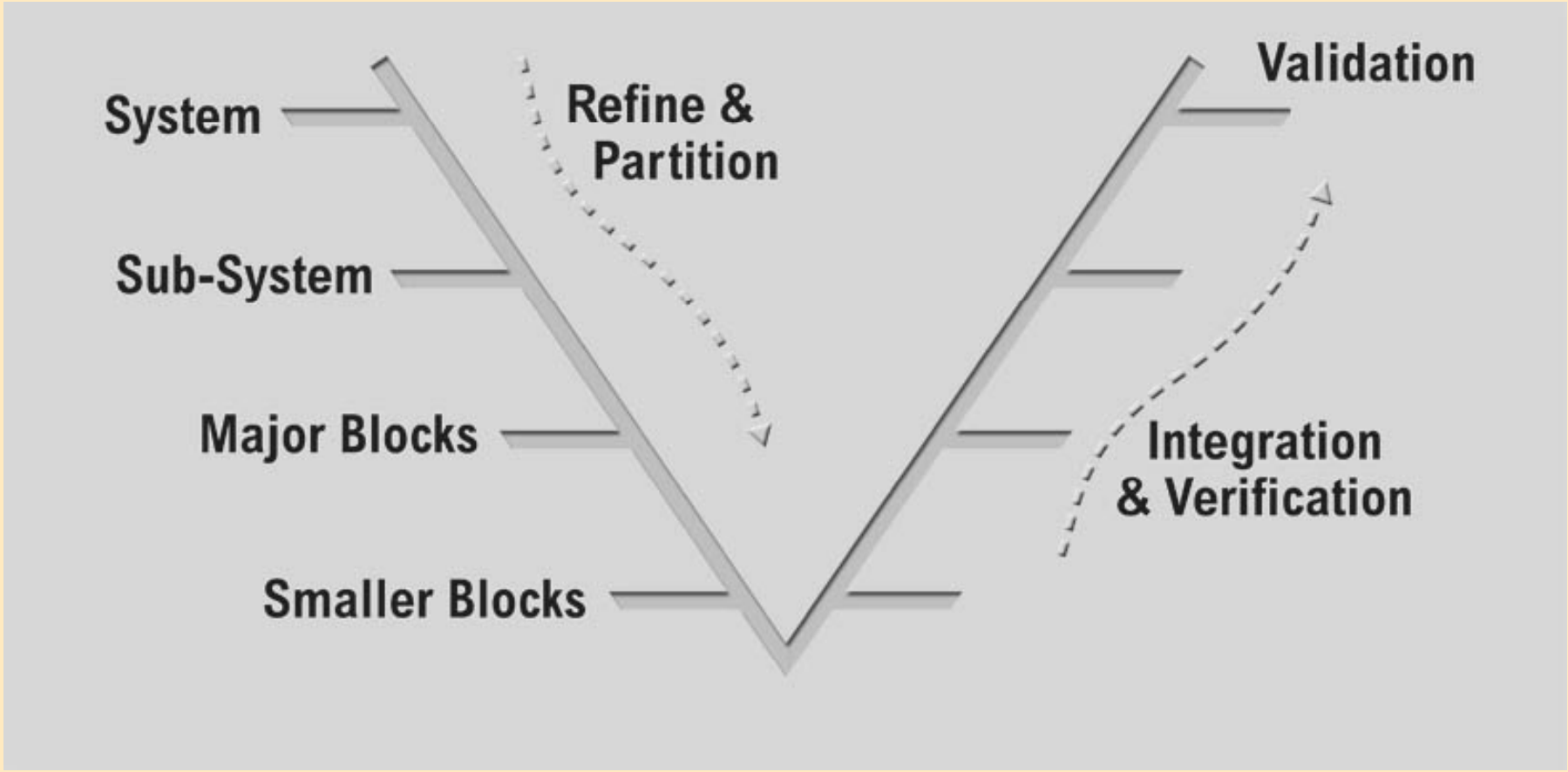
A TLM-driven Design and Verification Methodology

Brian Bailey Consulting

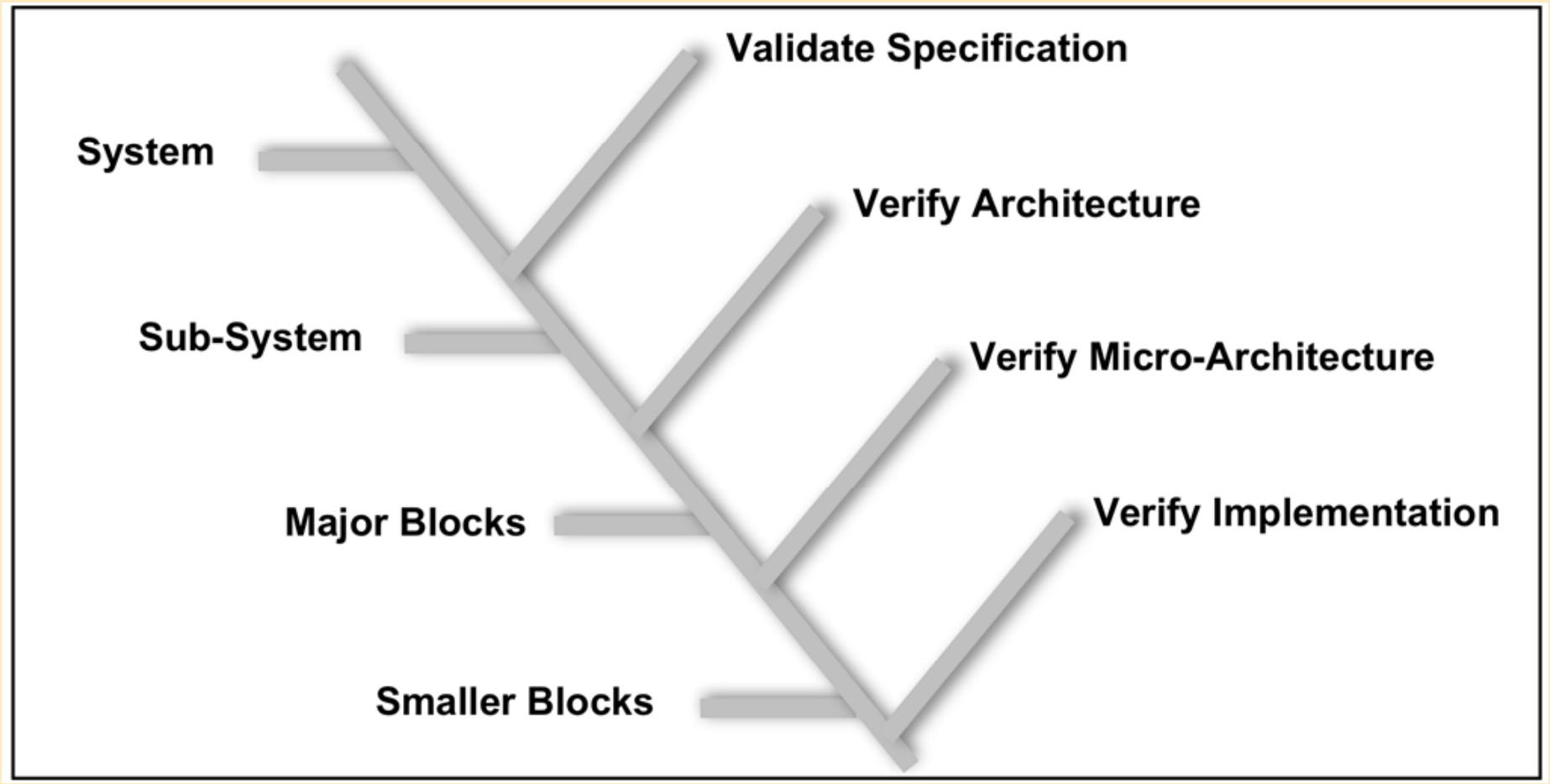
Work while consulting for Cadence Design Systems

Email: brian_bailey@acm.org
Tel: 503 632 7448
Cell: 503 753 6040
Web: brianbailey.us

Existing Methodology



Methodology Objective



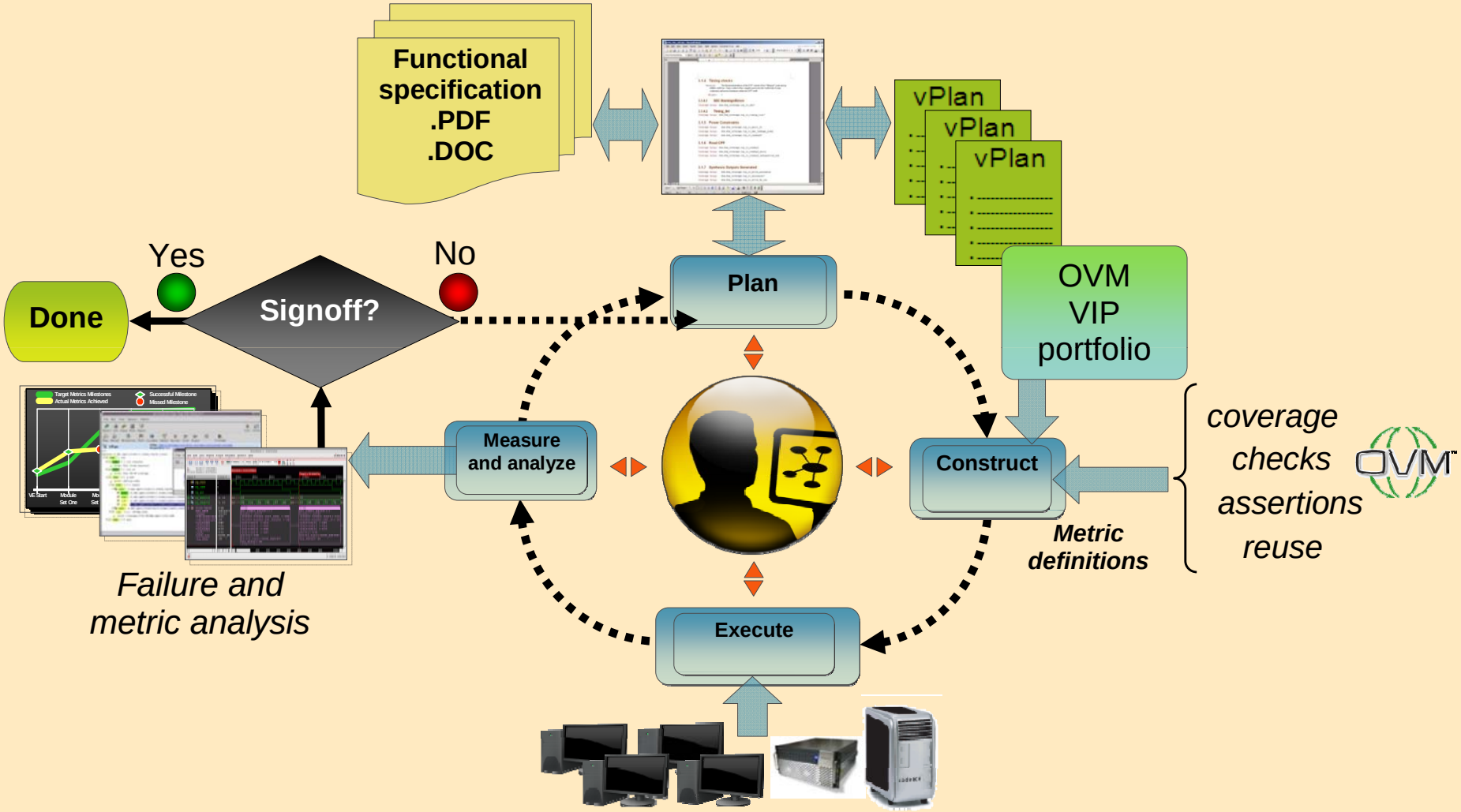
Methodology Advantages

- Increases efficiency
- Early validation
- Incremental verification
- Increases testbench modularity and re-use
- Eliminates integration verification
- Verify important stuff first
  - Implementation details verified later
- More predictable schedules

Separating Concerns

- **Separating Computation and Communication**
  - **Enables greater reuse**
  - **Enables functional virtual prototypes**
  - **Feeds high-level synthesis process**
  - **Independent refinement**
- **Separating Function and Architecture**
  - **One function, multiple implementation**
  - **Verify once**

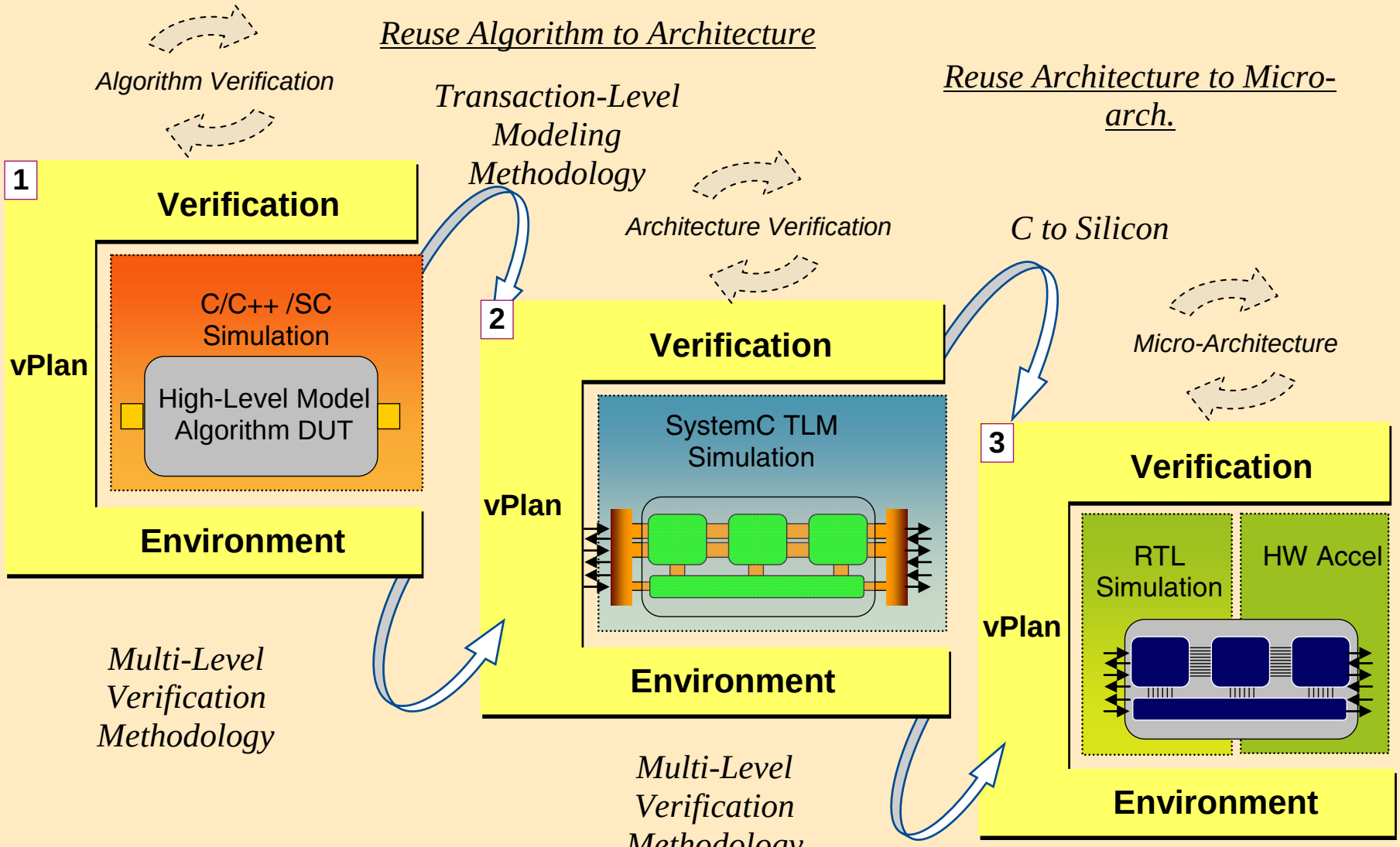
Multi-level metric driven



Testbench simulation, formal, hybrid, HW/SW co-sim, equivalence checking, CDC, constraint checking, LPV, MSV, sim acceleration, emulation

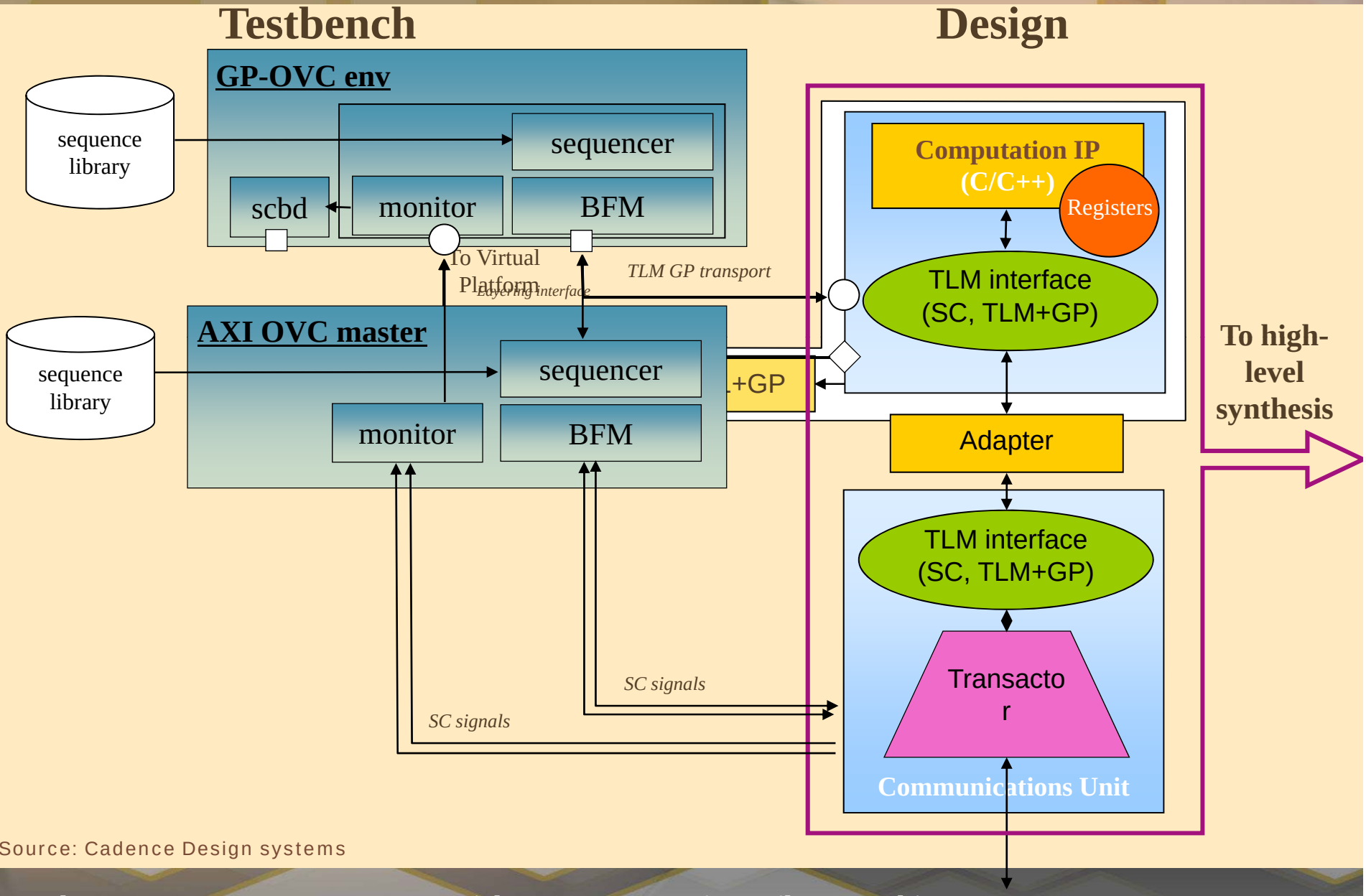
Source: Cadence Design systems

Three Stage Flow



Source: Cadence Design systems

Design and Verification Flows



Source: Cadence Design systems

TLM Interface

- Start with basic transport mechanisms from TLM 1.0
- Must be synthesizable
  - Removes all simulation features
  - Removes all dynamic allocation capabilities
- Needs extra constructs
  - Reset capabilities
- Extend with TLM 2.0 capabilities
  - GP capabilities added
- Synthesizable TLM+GP interface to be offered for standardization