STING

A Bare Metal Software System for Design Verification of SoC/CPU Implementations

Overview

STING is an architecture agnostic software system for development of constrained random, focused and coverage based test stimulus. It generates intelligently crafted self-checking test stimulus/code sequences (in form of bootable ELF images) for all the IPs present in the system from user defined test configurations/specifications. Since the tests are self-sufficient in nature, executing them without any checker errors guarantees the functional correctness of the system implementation.

The software stack consists of offline test generators, light-weight device drivers, test applications, checkers and a micro-kernel. The components are highly configurable and can be combined to create a portable bare-metal embedded program as per the needs of the verification environment. For example, since the instruction footprint for slow pre-silicon simulation environments has to be small, the program will just consist of a kernel and the tests with majority of setup and checking being done offline. Whereas for post-silicon environment, we can execute a program which has all the components stitched together and running on the target at full speed.

Rich test specification schemes allow any system scenario to get mapped to a test which is portable across pre-silicon simulations, FPGA, emulation and post-silicon. Very debug friendly, scalable and extensible in design, it embodies the best methodologies and practices in the industry whilst providing innovative solutions for the challenges in client's ecosystem.

Key Features/Advantages

<u>TECHNICAL</u>

- Enables seamless execution of test stimulus across verification environments (pre-silicon simulations, emulation, FPGA prototypes and post-silicon) without any external dependencies or code change

  ADVANTAGE

  - Test stimulus gets enabled early on the design
  - Little or no effort for validation teams to ramp up from one verification environment to another
  - No loss of stimulus when moving from one environment to another.
  - Failing code sequence can be easily migrated to simulation environment for faster debug
  - Removes redundant and duplicate test stimulus for different environments
- Enables cross product of scenarios for maximum coverage

ADVANTAGE

- Validation engineer writes each scenarios as a discrete unit which is as easy as writing a simple directed test case
- STING enables cross product of various scenarios giving rise to exponential number of scenarios with very little effort from the users
- Easily cover complex scenarios which take a lot of time and effort to cover using hand written tests
- Provides coverage for all the IPs at the same time. Concurrent execution of traffic results in maximum system stress and thorough testing of IP(s).

- Extremely fast test generation and execution

  ADVANTAGE

  - Execute as many tests as possible covering a large amount of validation space in very short time

  > STING is able to get a throughput of 200 tests/sec on a 16-CPU silicon for a 1000-instruction long random stream of instructions per CPU generated by its ALU stressor algorithm. That gave us close to over 17 million tests on a single system in a day's time.

  - Efficient use of simulation/emulation cycles
  - Generate tight sequences of code. Get a faster closure on coverage.
  - Focused test development framework and advanced scheduling schemes gives a tight control on generation of test stream and resources required by it, with little or no overhead.

- Supports multiple modes of execution. For example, offline mode for slow top-level simulations, on-target mode for faster environments like FPGA and post-silicon, etc.

  ADVANTAGE

  - Improves test efficiency by removal of operations which do not contribute to test stimulus in slower environments
  - Increased test throughput on faster targets

- Highly intuitive and configurable test stimulus specification mechanism

  ADVANTAGE

  - Address complex specification requirements with sufficient ease
  - Enables scenario driven test content development. Parameters in input configurations carefully chosen to allow any system scenario/test stimulus to get easily mapped.
  - Test configurations for the existing IP test plan scenarios can be created easily.
  - Easy reuse of test stimulus across multiple generations of SoC and across heterogenous hardware configurations

- Customers can retain existing legacy tests by easily integrating them within Sting's kernel framework and get benefits of its execution methodology.

- Custom mechanisms for test stimulus analysis and evaluation

  ADVANTAGE

  - Log based mechanisms to analyze the intent and coverage from a test stimulus
  - Evaluation of hardware performance counters to measure regression in performace
  - Coverage based feedback to users enable generation of high quality stimulus
  - Mechanisms can be extended easily to meet customer's requirements

- Checking and debug mechanisms that vastly improves the throughput of failure disposition.

  ADVANTAGE

  - Checking mechanism ingrained at different stages in Sting catches any behavioral anomaly as quickly as possible.
  - Ability to start the test at a later point enables failures to recreate faster
  - A host of debug features are implemented for test stimulus reduction and generation of debug information around the failure scenario.

- Provides an OS/application like layered architecture before more complex software is enabled on the target system

- Access to a library of test configurations available. Comprises of test stimulus for a large number of validation algorithms, device drivers, architectural use-cases e.g. MMU aborts, virtual page size change, recurring interrupts, memory ordering, cache coherency, branch prediction, register renaming, SD host controller device driver, CPU-IO cache-line false sharing etc.

  ADVANTAGE

  - Provides a starting base for validation activities
  - Driver framework allows integration of new devices easily

<u>STRATEGIC</u>

- Valtrix Systems provides a number of flexible engagement models which includes binary and full source code licenses

- Full source code licensing

  ADVANTAGE

  - Enables the team to debug failures and even extend tools if required
  - Retain comptetive advantage by non-disclosure of confidential design details
  - Company wide unlimited use

- STING based validation plans and test packages for IP(s)

- STING is easily portable to new CPU architectures should the customer choose to do so.

It already supports functional verification for ARMv8 and RISC-V implementations and can be easily ported to new implementations enabling the customer to have an architecture independent test suite.

About Valtrix Systems

Valtrix Systems was formed with a mission of creating world class tools and validation methodologies that will help customer effectively validate the designs of IPs and SoCs. The founding team comprises of system validation veterans with many years of experience in validation of complex microprocessors/systems. Founded in 2015, it is based in Bangalore, India.