



**PARADIGM<sup>®</sup>**  
**WORKS**

*Is my test done?*

**Shutdown Manager  
SystemVerilog**

*CDNLive! Nice, France - 2006*

*Ambar Sarkar, Matt Trostel*



## Overview

- ▶ The problem: How do I know when my test is done?
- ▶ The solution: Take a unified approach to stopping tests utilizing a centralized objection-based scheme
  
- ▶ What is Shutdown Manager?
- ▶ Benefits
- ▶ Features supported by Shutdown Manager



## The Problem: How do I know when my test is done?

- ▶ From one verification environment to the next, the common issue arises as to indicating end-of-test
- ▶ Common approaches for specifying test duration include
  - ▶ Absolute time
  - ▶ Number of transactions
  - ▶ When a functional coverage goal has been met
- ▶ In any complex distributed verification environment, it can be difficult to predict when the test is “really done”
- ▶ Need a mechanism to coordinate end-of-test amongst verification components



## The Solution: Take an objection-based approach

- ▶ Any verification component (transactor, scoreboard, generator, etc) should not stop the test
- ▶ Verification components are allowed to say:
  - ▶ “Do not stop the test until I think I’m done”
    - ▶ `raiseObjection();`
  - ▶ “I am now done”
    - ▶ `dropObjection();`



## What is Shutdown Manager?

- ▶ SystemVerilog base class implementation
- ▶ Automatically handles end-of-test coordination
- ▶ Provides simple API for raising and lowering of objections



## Benefits

- ▶ Simplifies end-of-test handling
- ▶ Intuitive and easy to use
- ▶ Easy to integrate into any verification environment
- ▶ Facilitates reusability and maintainability
- ▶ Provides debug features



## Features of Shutdown Manager

- ▶ Provides the following 2 services
  - ▶ Objection mechanism
    - ▶ Provides normal end-of-test notification via `raiseObjection` & `dropObjection`
  - ▶ Activity monitor
    - ▶ Provides notification of abnormal end-of-test conditions based on user-configurable design/environment watchdog timeout value



# Shutdown Manager

## Example

### Packet Generator:

```
while (num_packets > 0)
...
shutdown_mgr.raiseObjection();
scoreboard.post(packet);
shutdown_mgr.dropObjection();
```

### Packet Scoreboard:

```
task post (pkt_data a_packet)
...
shutdown_mgr.raiseObjection();

task check (pkt_data a_packet)
...
shutdown_mgr.dropObjection();
```

