OsvvmGlobalPkg Package User Guide

User Guide for Release 2015.01

By

Jim Lewis

SynthWorks VHDL Training

Jim@SynthWorks.com

http://www.SynthWorks.com

Table of Contents

1 OsvvmGlobalPkg Overview

OsvvmGlobalPkg provides types and global settings for the OSVVM packages CoveragePkg and AlertLogPkg.

2 Package User Subprogram Reference

2.1 Package Reference

```
library osvvm ;
use osvvm.OsvvmGlobalPkg.all ;
```

2.2 OsvvmOptionsType: Type for OSVVM options

```
type OsvvmOptionsType is (OPT_INIT_PARM_DETECT, OPT_USE_DEFAULT, DISABLED, FALSE,
    ENABLED, TRUE) ;
```

2.3 SetOsvvmGlobalOptions: Configuring OSVVM Options

Configuration values for the output from Alert, Log, and ReportAlerts.

```
procedure SetOsvvmGlobalOptions (
  WritePassFail  : OsvvmOptionsType := OPT_INIT_PARM_DETECT ;
  WriteBinInfo   : OsvvmOptionsType := OPT_INIT_PARM_DETECT ;
  WriteCount     : OsvvmOptionsType := OPT_INIT_PARM_DETECT ;
  WriteAnyIllegal : OsvvmOptionsType := OPT_INIT_PARM_DETECT ;
  WritePrefix    : string := OSVVM_STRING_INIT_PARM_DETECT ;
  DoneName       : string := OSVVM_STRING_INIT_PARM_DETECT ;
  PassName       : string := OSVVM_STRING_INIT_PARM_DETECT ;
  FailName       : string := OSVVM_STRING_INIT_PARM_DETECT
) ;
```

The following options are for WriteBin and ReportAlerts are:

WritePassFail	WriteBin prints Pass / Fail information
WriteBinInfo	WriteBin prints Bin ranges
WriteCount	WriteBin prints Count
WriteAnyIllegal	WriteBin always prints illegal bins
WritePrefix	Prefix for each line of ReportAlerts. Default "%% "
DoneName	Value printed after ReportPrefix on first line of ReportAlerts. Default "DONE"
PassName	Value printed when a test passes. Default "PASSED".
FailName	Value printed when a test fails. Default "FAILED"

SetOsvvmGlobalOptions will change as the packages are revised. To ensure forward compatibility into the future call it using named association.

```
SetOsvvmGlobalOptions (
    WritePrefix      => ">> ",
    DoneName         => "FIN"
) ;
```

After setting a value, a string value can be reset using OSVVM_STRING_USE_DEFAULT and an OsvvmOptionsType value can be reset using OPT_USE_DEFAULT.

2.4   OsvvmDeallocate

Deallocates all temporary storage in OsvvmGlobalPkg.

3   Package Developer Subprogram Reference

```
function IsEnabled (A : OsvvmOptionsType) return boolean ;

function ResolveOsvvmOption(A, B, C: OsvvmOptionsType) return OsvvmOptionsType;
function ResolveOsvvmOption(A, B, C, D: OsvvmOptionsType) return OsvvmOptionsType;
function IsOsvvmStringSet(A : string) return boolean ;
function ResolveOsvvmOption(A, B : string) return string ;
function ResolveOsvvmOption(A, B, C : string) return string ;
function ResolveOsvvmOption(A, B, C, D : string) return string ;

impure function ResolveOsvvmWritePrefix(A : String)  return string ;
impure function ResolveOsvvmWritePrefix(A, B : String)  return string ;
impure function ResolveOsvvmDoneName(A : String)  return string ;
impure function ResolveOsvvmDoneName(A, B : String)  return string ;
impure function ResolveOsvvmPassName(A : String)  return string ;
impure function ResolveOsvvmPassName(A, B : String)  return string ;
impure function ResolveOsvvmFailName(A : String)  return string ;
impure function ResolveOsvvmFailName(A, B : String)  return string ;

impure function ResolveCovWritePassFail(A, B : OsvvmOptionsType)
    return OsvvmOptionsType ;  -- Cov
impure function ResolveCovWriteBinInfo(A, B : OsvvmOptionsType)
    return OsvvmOptionsType ; -- Cov
impure function ResolveCovWriteCount(A, B : OsvvmOptionsType)
    return OsvvmOptionsType ; -- Cov
impure function ResolveCovWriteAnyIllegal(A, B : OsvvmOptionsType)
    return OsvvmOptionsType ;  -- Cov
```

4   Compiling OsvvmGlobalPkg

See OSVVM_release_notes.pdf for the current compilation directions. Rather than referencing individual packages, we recommend using the context declaration:

```
library OSVVM ;
  context osvvm.OsvvmContext ;
```

5   Future Work

OsvvmGlobalPkg.vhd is a work in progress and will be updated from time to time.

6   About the Author - Jim Lewis

Jim Lewis, the founder of SynthWorks, has thirty plus years of design, teaching, and problem solving experience. In addition to working as a Principal Trainer for SynthWorks, Mr Lewis has done ASIC and FPGA design, custom model development, and consulting.

Mr. Lewis is chair of the IEEE 1076 VHDL Working Group (VASG) and is the primary developer of the Open Source VHDL Verification Methodology (OSVVM.org) packages. Neither of these activities generate revenue. Please support our volunteer efforts by buying your VHDL training from SynthWorks.

If you find bugs these packages or would like to request enhancements, you can reach me at jim@synthworks.com.