esa DOCUMENT

document title/ titre du document

SINGLE EVENT UPSETS SIMULATION TOOL USER MANUAL

prepared by/*préparé par* Daniel González Gutiérrez

reference/*réference* TEC-EDM/DGG-SST1
issue/*édition* 1
revision/*révision* 2
date of issue/*date d'édition* 26/07/04
status/*état* Draft
Document type/*type de document* User Manual
Distribution/*distribution*

European Space Agency
Agence spatiale européenne

APPROVAL

Title *titre*		issue *issue* 1	revision *revision* 2

author *auteur*		date *date* 26/07/04

approved by *approuvé by*		date *date*

CHANGE LOG

reason for change */raison du changement*	issue/*issue*	revision/*revision*	date/*date*
GUI Added	1	1	16/03/04
Split between user manual and functional description	1	2	26/07/04

CHANGE RECORD

Issue: 1 Revision: 2

reason for change/*raison du changement*	page(s)/*page(s)*	paragraph(s)/*paragraph(s)*

TABLE OF CONTENTS

1 SCOPE

The object of this document is to describe how the Single Event Upsets Simulation Tool shall be used.

2 TERMS AND ACRONYMS

DUT	Design Under Test
GUI	Graphical User Interface
SEU	Single Event Upset
SST	Single Event Upsets Simulation Tool

3 APPLICABLE DOCUMENTS

The table 3.1 shows the documents referred in this document.

[ModelCRef]	Modelsim Command Reference.
[SSTFunDes]	SST Functional Description

Table 3.1 Applicable documents

4 OVERVIEW

The SEUs Simulation Tool consist of a set of Perl and Tcl scripts that used in conjunction with a Design Under Test and a Test Bench, allow the user to upset (bit flip) in a controlled and effective manner, any register or internal signal of the DUT, while a simulation is running. The scripts can be invoked either using the Graphical User Interface or the command line mode.

The current version of the SST only works with *Modelsim*. Any simulator supporting the 'force' command could be easily added in the future.

Please, refer to [SSTFunDes] for a more detailed description of the SST.

5 INSTALATION

Prior to copying the SST files into your computer, make sure that your system has Perl installed. It can be downloaded for free from: http://www.perl.com/download.csp.

The following two actions have to be taken before running the tool:

- Copy all the SST files into the same directory.
- The folder in which the scripts are finally placed, should be added both to your path environment variable and to the DOPATH environment variable (see [ModelCRef] for details)

6 DIRECTORY STRUCTURE

6.1 Files supplied by the user

- **./**

  HDL testbench_files

6.2 Files generated by the tool

The purpose and contents of the following files is described in [SSTFunDes]

- **./SST/control_files**

  all_instances.dat, *all_wires_parser.log, hierarchy.dat,sst.do*, *SST_perl_package.pm*
- **./SST/wire_files**

  In this folder we can find all the wire files.

7 RUNNING THE TOOL

To invoke any tcl script, type: 'do tcl_script.tcl' in a *Modelsim* window
To call any Perl sript, type: 'exec perl -S perl_sript.pl' in a *Modelsim* window.

Either if the GUI is being used or the command line mode has been chosen, the use of the tool requires four sequential steps. Note that the first two steps, which correspond to the initialisation phase, are only necessary the first time the tool is run, or if any configuration value has been changed after running the initialisation phase for the first time.

For now on we assume that the GUI is going to be used. Commands equivalent to the GUI options and buttons will be commented in parenthesis. To invoke the GUI, type: 'do SST_gui.tcl' in the *Modelsim* command line interface. The window of figure 7.1 should appear.

Figure 7.1 SST Graphical User Interface main window

7.1 Load the Test Bench of the DUT in the simulator

Prior to start executing the SST it is necessary to load the test bench that checks the correct functionality of the DUT.

7.2 Gather information about the design

In order to let the SST know about the design to be tested (basically, its hierarchy and the number and type of signals found in each module), the user will have to click in the 'Actions/Gather design information' menu button of the GUI (or directly invoke the tcl script *SST_startup.tcl*)

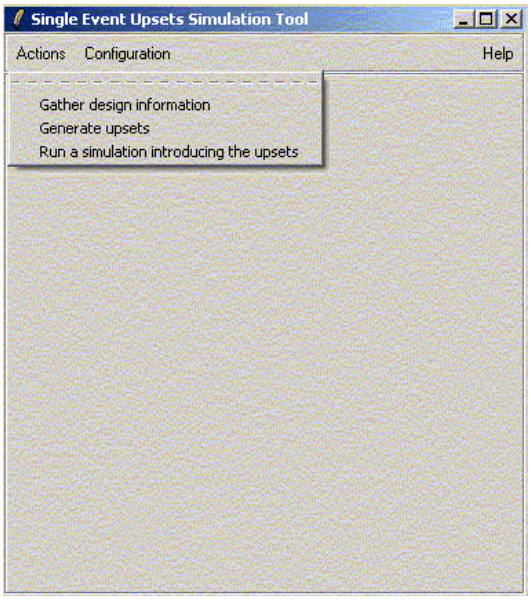

Figure 7.2 SST GUI 'Actions' menu button.

7.3 Select the wires that will be upset and when.

Once the information about the design has been collected, the selection of the wires that are going to be upset has to be done by clicking in the 'Actions/Generate Upsets' GUI menu button, and filling the entry boxes and check buttons of the interface window, accordingly (or by executing the Perl script *SST_upset_generator.pl* with some options and switches). Figure 7.3 shows the 'Generate Upsets' window.

7.4 Run a simulation introducing the upsets

Once the wires to be upset have been selected, the user will have to run a simulation upsetting the chosen wires at the specified times values. This can be accomplished by clicking in the 'Actions/Run a Simulation introducing the upsets' GUI menu button (or by executing the macro *sst.do* generated by the the script *SST_upset_generator.pl*).

The test bench of the DUT should flag any error produced by the presence of a non expected value in a register that will drive to a failure on the simulation.

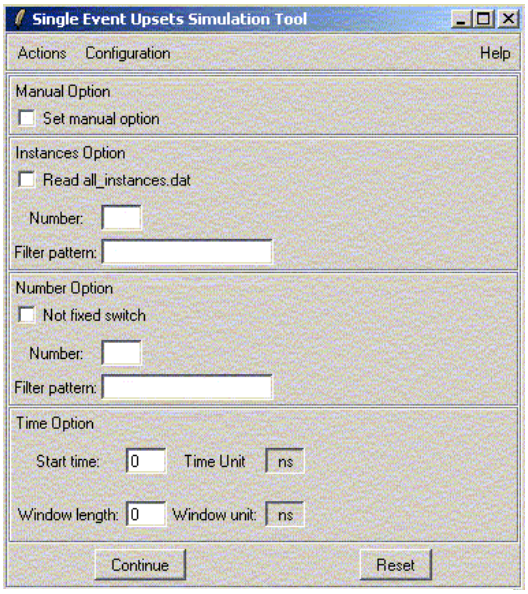

Figure 7.3 SST GUI, 'Generate upsets'.