Minitracer: A minimalist requirements tracer for HDL designs



Carlos López-Melendo, Hipólito Guzmán-Miranda Department of Electronic Engineering, Universidad de Sevilla, Sevilla, Spain

carloslm2019@gmail.com, hguzman@us.es

Abstract

When designing systems using HDLs (Hardware Description Languages), it is important to track the implementation status of the project requirements. Having tools for quick assessment of the status of both the implementation and verification of the project requirements can ease management job, give the engineers more perspective and control over their deadlines, and reduces overall project risk.

This work presents a minimalist requirements tracer for HDL designs, whose main purpose is to obtain and represent the knowledge about the state of an HDL development project, considering its requirements and test suite.

Objectives

- 1.- Manage a high number of tests: pass / fail status?
- 2.- Trace the test-requirement status: which requirements are tested by which tests?
- 3.- Trace the code-requirement relation: where in the code is each requirement implemented?

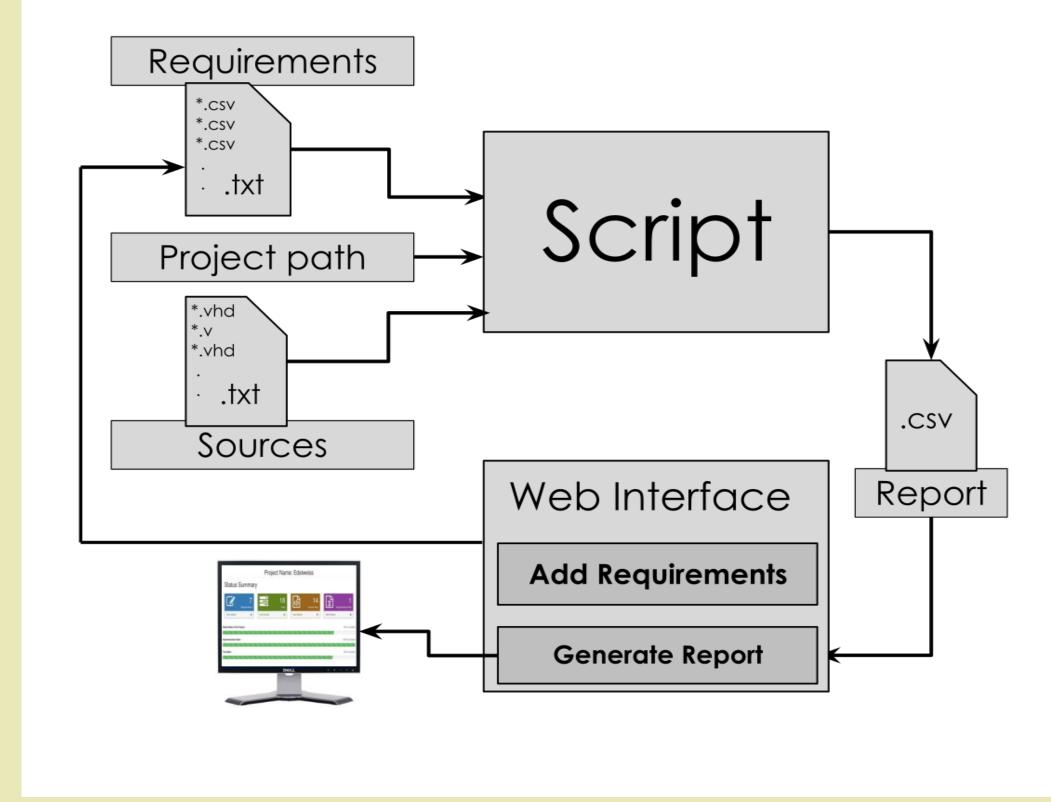
Architecture

The application is divided into three components:

- 1. A web interface where the user can specify the project requirements, a description of each one and the associated tests.
- 2. A script which generates a report with the state of the project, based on the requirements list and the pass/fail state of all tests in the project.
- 3. A second web interface which shows the state of the project, according to which requirements are implemented and which ones are also verified in the tests which have executed correctly. Verification and traceability matrices are also generated for easier assessment of project status.

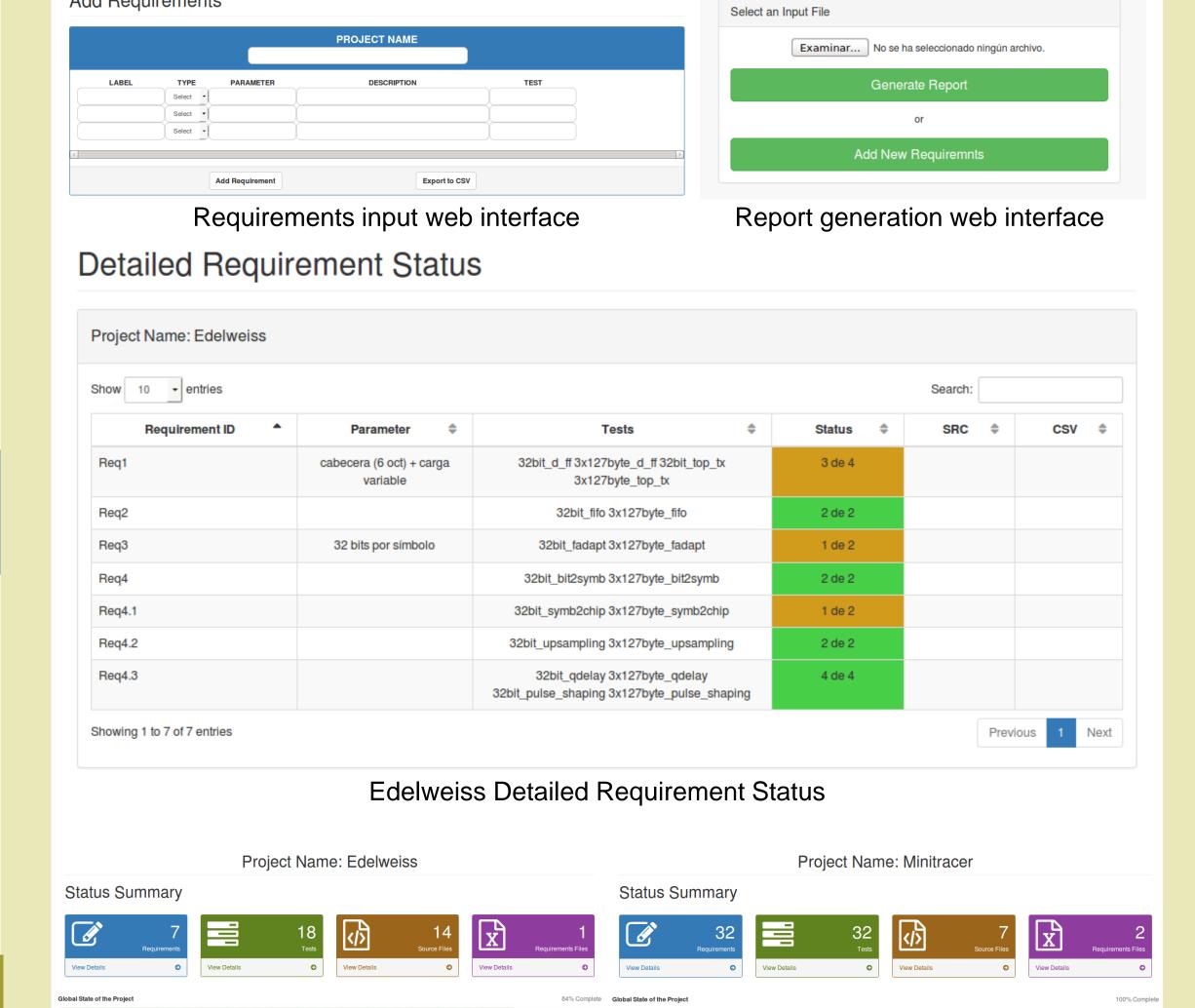
Workflow

- 1.- Write each requirement using the web interface.
- 2.- Click on Export to CSV.
- 3.- Execute the script (arguments: project path, text file with all paths to the .csv requirement files, text file with all the paths to the HDL sources). A .csv file is generated with all the information about the state of the project.
- 4.- View the results on the web interface.



Results

This software was tested with two projects. One of them is the transmitter module, developed in VHDL, of the Edelweiss project (Zigbee PHY interface) for which Minitracer generated a report with the current status of the project. The second project where Minitracer was tested was Minitracer itself.



Edelweiss Status Summary



Requirements Description

ow 10 <u></u> entries	Search:
Requirement ID	△ Description
Arg_linea_comand	Se leen los argumentos de linea de comandos
Añade_filas_req	Función que añada filas al formulario para insertar requisitos
Barras_progreso	Barras de progreso que indiquen el estado del proyecto
Busca_req_csv	Existe función que busca los requisitos en los archivos csv
Busca_req_src	Función que busque requisitos en ficheros del proyecto
Calculo_estado_proyecto	Se calcula el estado del proyecto
Compara_resultados	Se comparan los requisitos de los csv y los ficheros del proyecto
Consulta_test	Consulta test generados con CTest
Detalle_files	Página con los requisitos ordenador por archivos
Detalle_req	Página con los detalles de los requisitos
showing 1 to 10 of 32 entries	Previous 1 2 3 4 Next

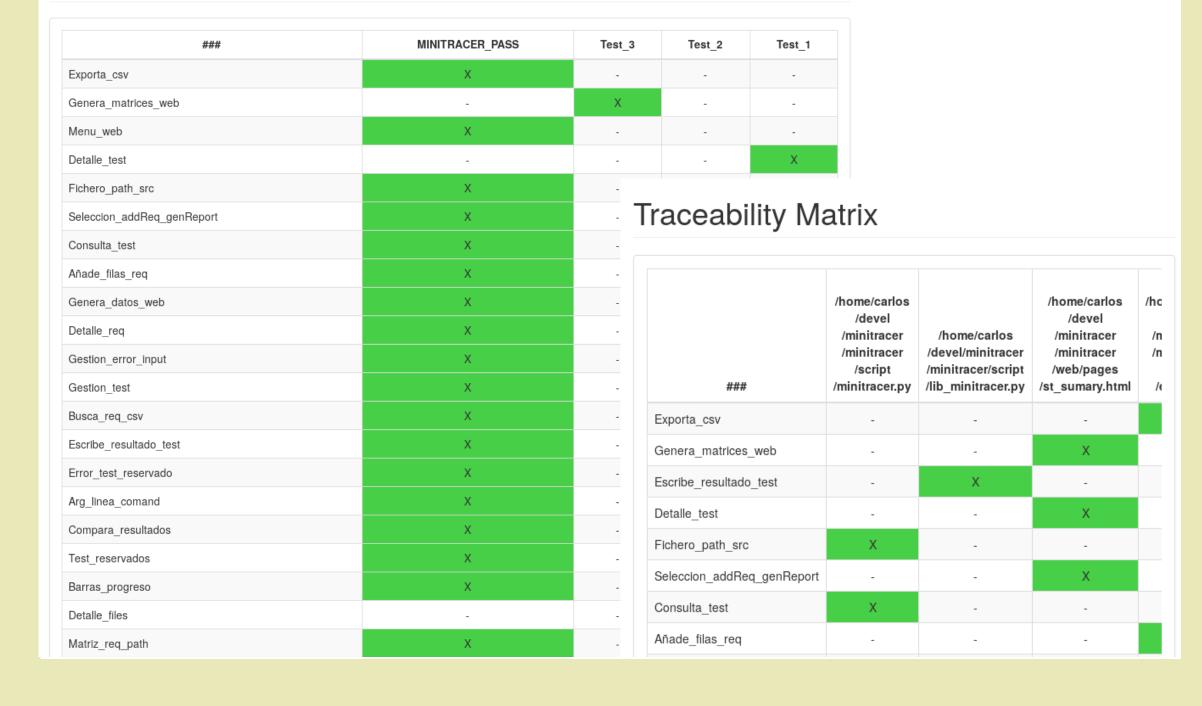
Requirements Description

Tracing Files

	Project Name: Minitracer		
Requirements and files where they are implemented	Show 10 • entries	Search:	
	Requirement ID	File Path	
	Arg_linea_comand	/home/carlos/devel/minitracer/minitracer/script/minitracer.py	
	Añade_filas_req	/home/carlos/devel/minitracer/minitracer/web/js/export.js	
	Barras_progreso	/home/carlos/devel/minitracer/minitracer/web/pages /st_sumary.html	
	Busca_req_csv	/home/carlos/devel/minitracer/minitracer/script/minitracer.py	
	Busca_req_src	/home/carlos/devel/minitracer/minitracer/script/minitracer.py	
	Calculo_estado_proyecto	/home/carlos/devel/minitracer/minitracer/script/minitracer.py	
	Compara_resultados	/home/carlos/devel/minitracer/minitracer/script/minitracer.py	
	Consulta_test	/home/carlos/devel/minitracer/minitracer/script/minitracer.py	
	Detalle_files	/home/carlos/devel/minitracer/minitracer/web/pages /st_sumary.html	
	Detalle_req	/home/carlos/devel/minitracer/minitracer/web/pages /st_sumary.html	
	Showing 1 to 10 of 32 entries	Previous 1 2 3 4 Next	

The verification matrix shows whether all requirements are actually verified, and the traceability matrix shows whether all requirements are actually implemented, and in which files.

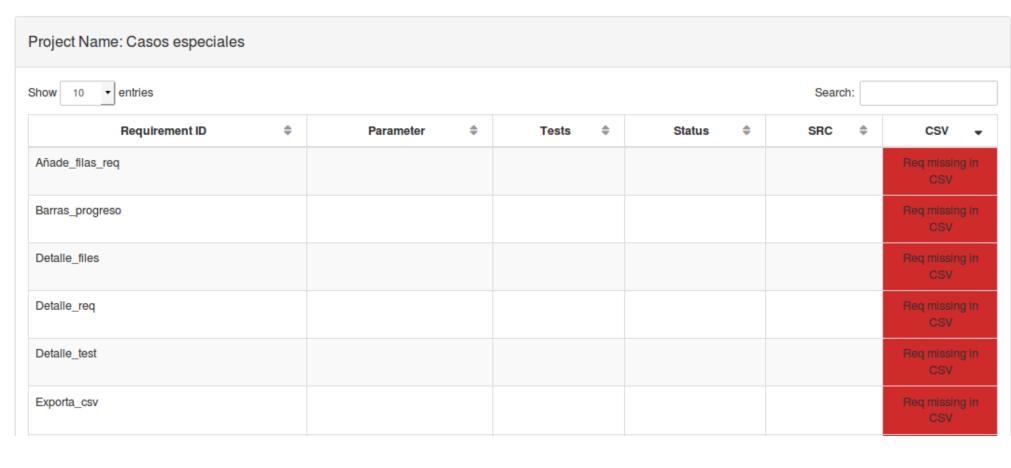
Verification Matrix



Special case detection

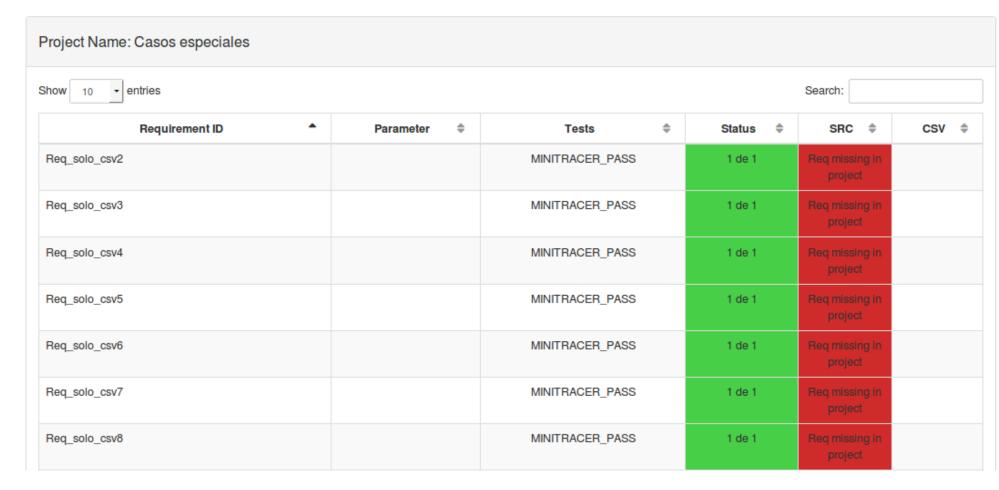
The tool detects the special cases that can happen when the defined requirements do not match the implemented code and tests. This allows designers to check the consistency of their own requirements and test organization, for example when requirements are missing in either the sources or the requirements files:

Detailed Requirement Status



Test appears in test result files but not in requirements files

Detailed Requirement Status



Requirement not implemented in project source files

Conclusions

A minimalist tool for requirements tracing in HDL designs has been presented, which implements the required features described in the Objectives section: tracking both implementation and verification of requirements, relating requirements to their tests, and relating requirements to where in the source code they have been implemented.

The tool allows the user to either use the web interface to specify the requirements or to write them directly in a .csv file. The tool reads the implementation status of the requirements in the HDL source files and the verification status from CTest test result files, but can be extended to integrate with other automated testing tools.

A project status web page can be generated for quick assessment of requirement implementation and verification status, in which both verification and traceability matrixes can be reviewed.

Future work will focus on adding new features to the tool, such as using the integrating with different automatic testing frameworks and using the git version control system to automate tracking the changes to the requirements.

Licensing

Minitracer is licensed under the GNU GPL (General Public License) V3.0. It can be found at: https://github.com/carlopmel/minitracer



Acknowledgement

This work was partially supported by the Spanish Ministerio de Economía y Competitividad, through the project «Evaluación Temprana de los Efectos de Radiación Mediante Emulación y Plataformas de Testeo. Estrategias de Mitigación en Dispositivos Reconfigurables y Sistemas Empotrados.» (RENASER3), project reference ESP2015-68245-C4-2-P (MINECO/FEDER)