Functional specification document for fpga-design-re GUI

Sebastian Wallat
sebastian.wallat@rub.de

September 10, 2015

Contents

1 Target acquisition

The purpose of this chapter is the definition of target without specifying the required functions to implement the targets.

1.1 Mandatory ('must') criteria

This section enumerates the required functions of the final product. These are imperative functions that are needed in the final product. The system is without these criteria not usable for the desired purpose.

1. Graphical view of data model
2. Plugin executable
3. Plugin effect visible in Graphic view
4. Textview (Nodes & Edges)
5. Items Selectable
6. Source View
7. Color sync different views

1.2 Optional ('should') criteria

The fulfillment of the following criteria is desirable but not mandatory. The implementation of these criteria should be intended if enough capacities are available.

1. Color specification of plugins
2. Grouping of items
3. Selection model for graphical and standard views
4. Project information files
5. Search for node/edge type or specific information
6. Syntax highlighting VHDL
7. Miniview in file

8. Line numbers text view
9. Tabbed view
10. Keyboard shortcuts

1.3 Distinction criteria

The following criteria are explicitly intended not to be fulfilled.

2 Product application

In this chapter the environment and boundary conditions are specified.

2.1 Area of application

The final GUI should be used in laboratory and open source environments.

2.2 Target audiences

The main audience are hardware reverse engineers and cryptanalysts, who are trying to reproduce and modify a given netlist and produce a synthesizeable file format.

2.3 Operating conditions

This section should define the physical constraints of the system. Such conditions are daily usage time, the information if the system is under steady observation by an operator or weather an unattended use is intended.

attended or unattended The GUI is intended to be used attended only.

number of parallel netlists The GUI is intended to operate on one netlist at a time.

3 Environment

3.1 Software

Which software needs to be available to use the specified system. Libraries and version numbers.

OS The development is mainly done on a Apple MacBook Pro. Beside Mac OS X 10.10 support for Linux (Ubuntu LTS) and basic support for Windows 10 should be provided.

Qt 5.5 The basic graphic framework used is Qt 5.5. Here mainly the widget framework is used.

3.2 Hardware

The supported input and output platform for netlists is Xilinx Spartan 6.

4 Product overview

Figure 4.1 depicts the system context.

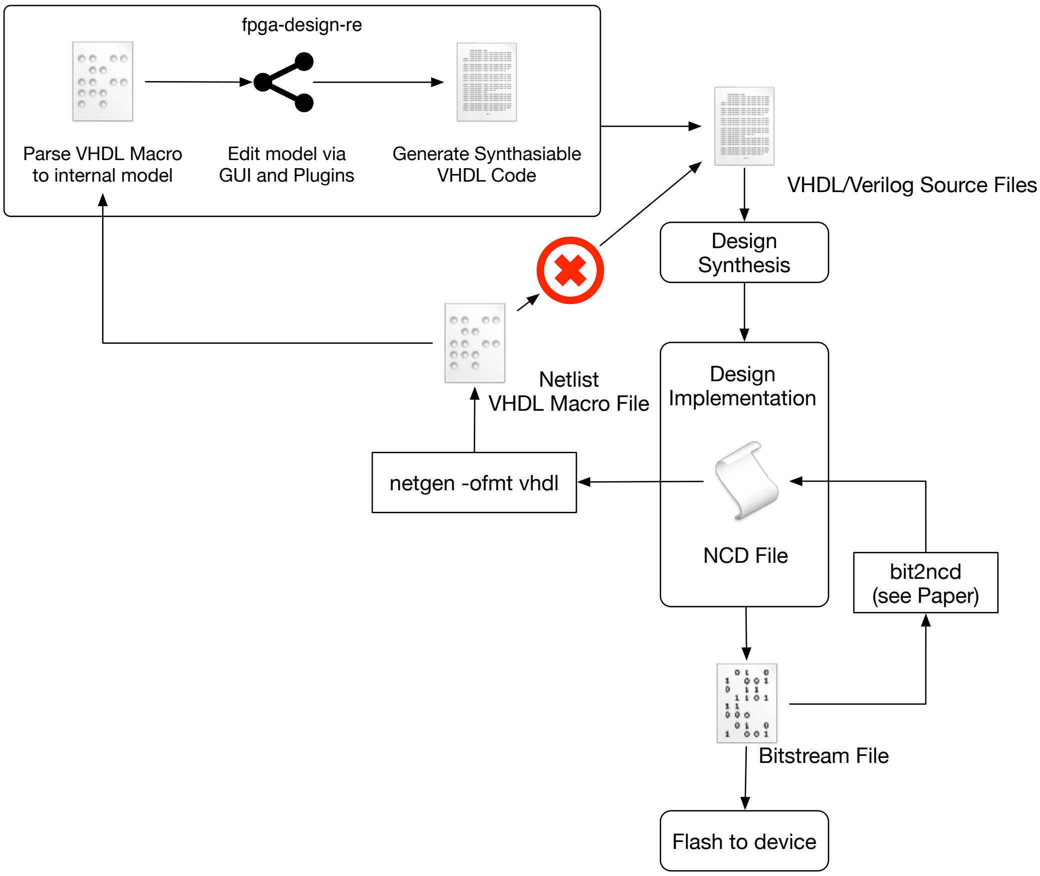


Figure 4.1: System Context

4.1 GUI

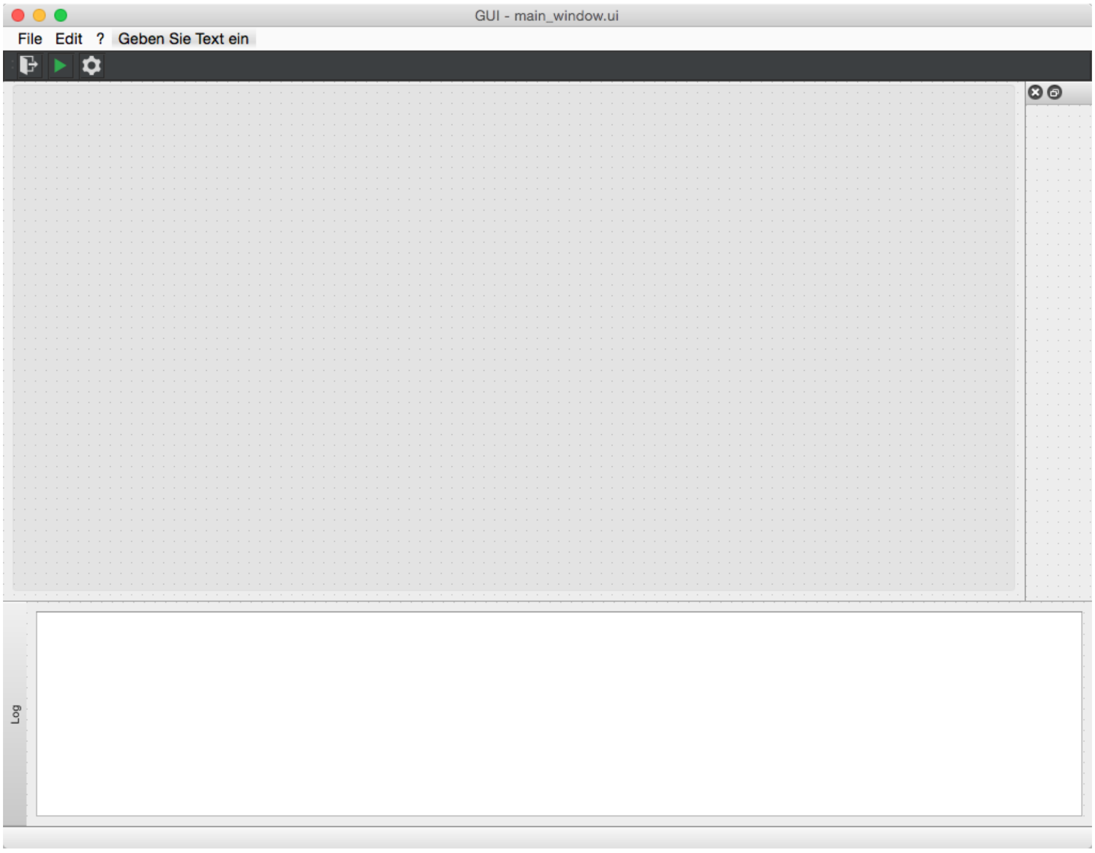


Figure 4.2: Main Window UI

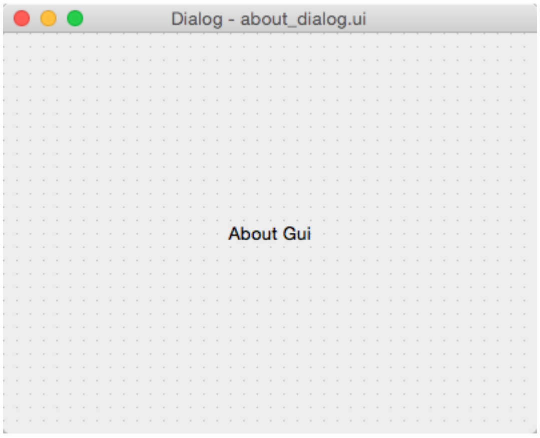


Figure 4.3: About Dialog

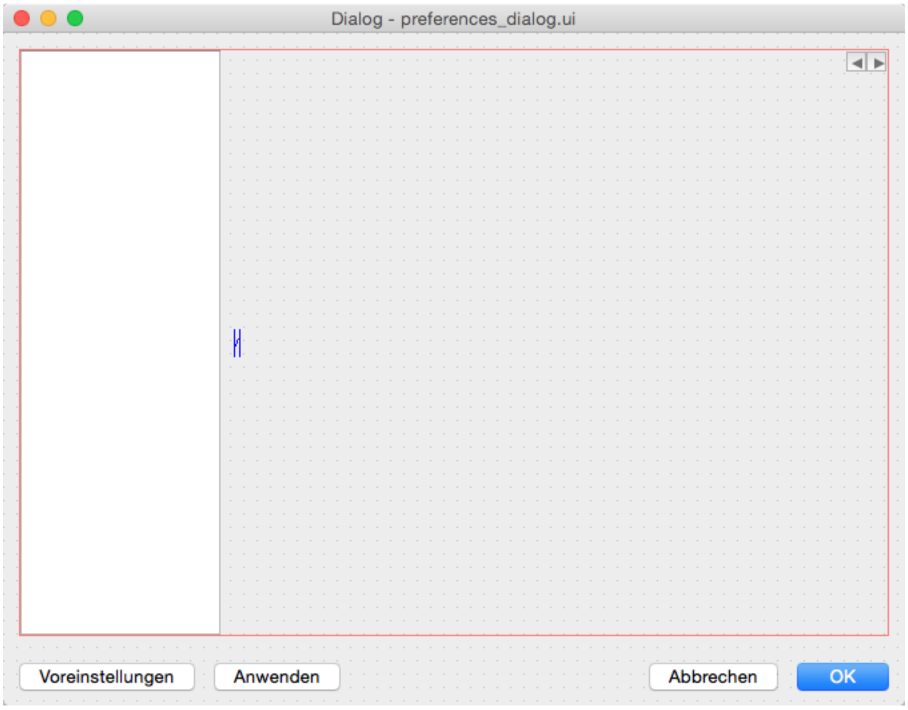
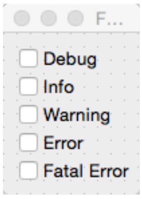


Figure 4.4: Preferences Dialog



□ Debug
□ Info
□ Warning
□ Error
□ Fatal Error

Figure 4.5: Logger preferences widget

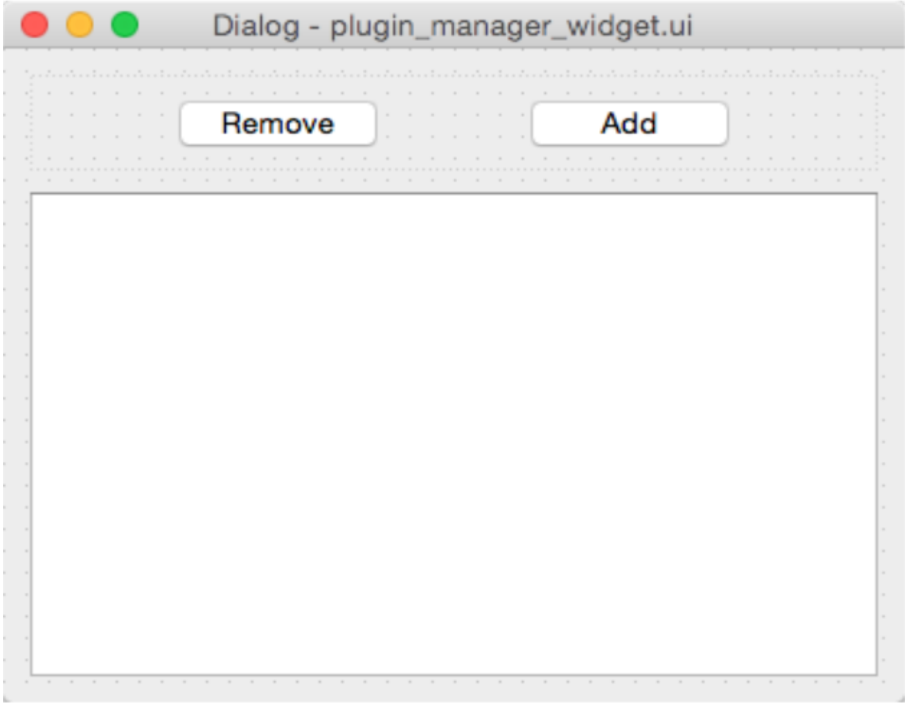


Figure 4.6: Plugin manager widget

5 Data

This chapter should described the pieces of information that are needed to be saved persistently.

- Json file(s) - internal data model
- Input file
- Synthezisable output file

List of Figures