

S-engine™: A System-Level Visualization and Editing Engine for EDA Tools

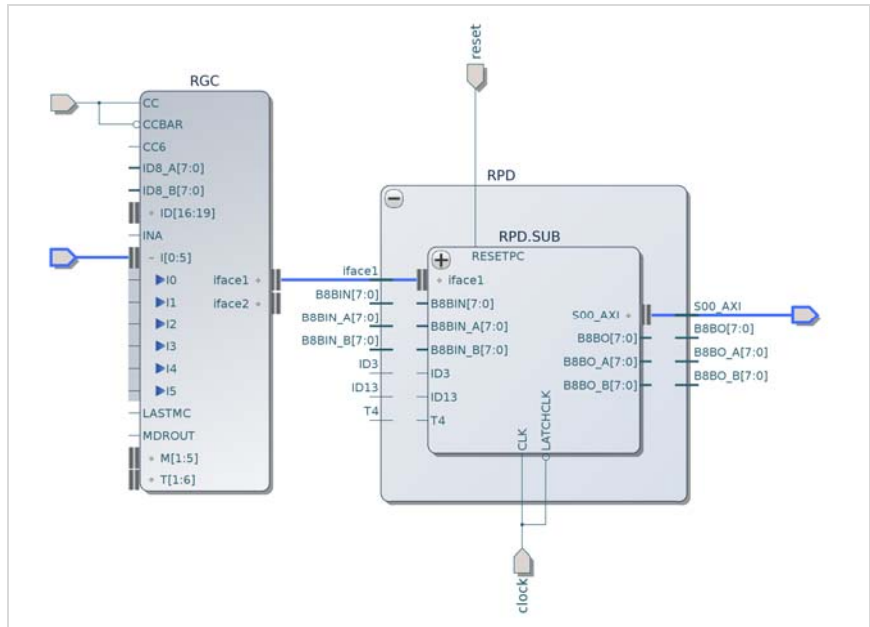
With S-engine Concept Engineering introduces a radically new concept of visualization and smart editing technology. EDA tool manufacturers working with high-level system structures can now easily create modern system-level and IP-centric design and visualization tools with built-in editing capabilities.

Automatic schematic generation has been enhanced to allow visualization of higher levels of abstraction, such as interface connection and intelligent IP-on-the-fly management to easily handle configurable IP blocks.

Integrated with EDA tools, S-engine helps designers of system-level, SoC, NoC and IP products to assemble, visualize and modify behaviour and architecture of their systems interactively.

Place-and-route techniques developed by Concept Engineering produce clean

system-level schematics for complex designs. When integrated into high-level synthesis tools S-engine provides excellent control and visibility over the synthesis process.



- Automatic system-level schematic generation
- System visualization engine with built-in smart editing capabilities
- IP-on-the-fly management for easy handling of customizable IP blocks
- Schematics generated quickly, easy-to-read and extendable
- Two way communication with host application for cross probing etc.
- Easy integration into EDA tools through simple APIs

The new schematic engine supports a mix of system-level, RTL and gate-level technologies in a single schematic diagram and helps to understand complex systems consisting of building blocks (IP) from many different sources.

At a Glance

FEATURES	BENEFITS
Simple and robust API	Ensures easy integration and reliable applications
Smart editing capabilities	Ideal engine for IP-centric system editors
Production-proven software components	Performance and quality of application is very high
Highly customizable component	Widget and application fit together
Qt, PyQt, Tcl/Tk, Windows, Java, JavaScript, wxWidget	Easily fits into your existing software development flow
Proprietary algorithms	Result in easy-to-read schematics and short response times
On-the-fly schematic creation	Results in very high speed and capacity
Bi-directional communication between widget and application	Allows interaction with the application (e.g. cross-probing, highlighting, attribute display, ballooning)
Incremental schematic viewing	Allows interactive modification of schematic fragments
Windows, Linux and UNIX platform support	Application will work on almost any hardware platform
Built-in system- and IP-level symbols	Application works without specific symbol libraries

Widest Platform Availability

GUI Platform	NlviewQT	NlviewJA	NlviewJS	NlviewTK	NlviewMFC	NlviewWX
Supported GUI environment	Digia's Qt Framework 3, 4 and 5	Java SDK	Web Browser (HTML5 / ECMAScript 6)	Tcl/Tk 8.1 or later	Microsoft Foundation Classes	wxWidgets 2.42 or later
Available as	Class derived from QWidget	Component (AWT) JComponent (Swing)	JavaScript Library	Tk Widget	Class derived from CWnd	Class derived from wxWindow
Deliverable	Sources + Core Lib	Java Bean	JavaScript Sources	Tcl package Loadable extension	MFC Extension DLL and Sources + Core Lib	Sources + Core Lib
Customizable by	Qt Properties	Java Bean Properties Property Command	Property Command	Configure Options Property Command	Class Attributes Property Command	Property Command
API Interface	Class Methods and Signals / Slots	Component Methods and Event-Listeners	Object Methods and Callbacks	Tcl Commands and Callbacks	Class Methods and Notification Messages	Class Methods and Notification Messages
Printing	PostScript, PDF, SVG and Native Qt	PostScript, PDF, SVG and Native Java	PostScript, PDF, SVG	PostScript, PDF, SVG and Native Windows	PostScript, PDF, SVG and Native Windows	PostScript, PDF, SVG and Native wxWidgets

Company Contact

Concept Engineering GmbH · Bötzingen Str. 29 · 79111 Freiburg · Germany
 Tel: +49-761- 47094-0 · Fax: +49-761- 47094-29 · Email: info@concept.de · <http://www.concept.de>