Using Synopsys Design Compiler & FPGA Compiler to Synthesize Designs for MAX+PLUS II Software

SYNOPSYS®

Synopsys
700 East Middlefield Road
Mountain View, CA 94043
(650) 962-5000
http://www.synopsys.com

Introduction

The Altera® MAX+PLUS® II software interacts easily with third-party EDA tools such as the Synopsys Design Compiler or FPGA Compiler. With the MAX+PLUS II software, you can use the Synopsys Design Compiler or FPGA Compiler as your front-end tool to target Altera programmable logic devices (PLDs).

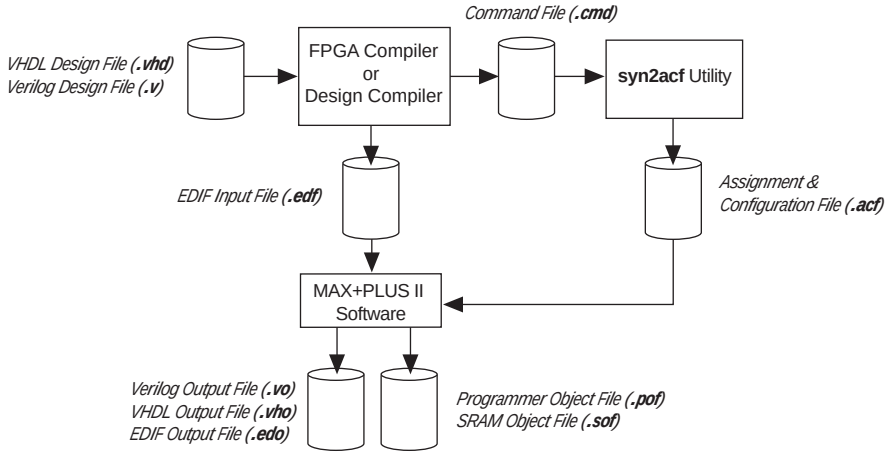
The Altera / Synopsys interface is included with the MAX+PLUS II software for UNIX workstations. The MAX+PLUS II software provides all the necessary files and libraries needed to support the Altera / Synopsys interface. The Altera / Synopsys tools let you quickly synthesize and implement designs for Altera devices, and perform multiple design iterations in a single day.

This technical brief describes how to:

- Set up the Synopsys design environment
- Generate an EDIF netlist file for Altera devices
- Pass timing constraints from the Synopsys design environment to the MAX+PLUS II software

Figure 1 shows a flow diagram of the Altera / Synopsys interface for UNIX workstations.

Figure 1. Altera / Synopsys Interface for UNIX Workstations



For complete instructions on how to synthesize and simulate a design using the Synopsys Design Compiler or FPGA Compiler and then import it into the MAX+PLUS II software for compilation, you should refer to the MAX+PLUS II Altera Commitment to Cooperative Engineering Solutions (ACCESS℠) Key Guidelines, which are available in the Altera Technical Support (Atlas℠) section of the Altera web site at **http://www.altera.com**. The guidelines are also available in HTML format in the **\lit\html\maxkey** directory on the *MAX+PLUS II CD-ROM* (version 8.2 or higher).

Setting Up the Synopsys Design Environment

To set up the Synopsys design environment, you must edit the **.synopsys_dc.setup** configuration file, which sets the Design Compiler or FPGA Compiler variables to target a specific Altera device family. For example, to target a FLEX® 10K device, you should add the following lines to the **.synopsys_dc.setup** file.

```
.
.
.
search_path={./<path to maxplus2 directory>
     synopsys/library/alt_syn/flex10k/lib}
target_library = {flex10k_fpga.db}
symbol_library = {altera.sdb}
synthetic_library = {flex10k_fpga.sldb}
link_library = {flex10k_fpga.sldb flex10k_fpga.db}
define_design_lib DW_FLEX10k_FPGA -path/
     <path to maxplus2 directory> /synopsys/library/alt_syn/flex10k/lib
     /dw_flex10k_fpga
edifout_netlist_only = "true"
edifout_write_attributes="false"
edifout_write_properties_list = LUT_FUNCTION
edifout_power_and_ground_representation = "net"
edifout_no_array = "false"
edifout_power_net_name = "VDD"
edifout_ground_net_name = "GND"
edifin_power_net_name = "VDD"
compile_fix_multiple_port_nets = "true"
bus_naming_style = "%s<%d>"
bus_dimension_separator_style = "><"
bus_interface_style = "%s<%d>"
.
.
.
```

For complete instructions on editing the **.synopsys_dc.setup** file for other Altera device families, see the MAX+PLUS II ACCESS Key Guidelines.

Table 1 describes some of the **.synopsys_dc.setup** file parameters.

Table 1. .synopsys_dc.setup File Parameters

Parameter	Description
`search_path`	Specifies the location of library files in the MAX+PLUS II software. When editing the **.synopsys_dc.setup** file, you should set the parameter to an Altera device library.
`target_library`	Specifies the technology library for design mapping.
`link_library`	Specifies the technology library for interpreting input descriptions.
`symbol_library`	Used to create symbols in the Synopsys Design Compiler or FPGA Compiler graphical user interface (GUI).
`synthetic_library`	Specifies the target device architecture for the Synopsys DesignWare interface.
`define_design_lib`	

The `synthetic_library` and `define_design_lib` parameters optimize functionality in specific device families by mapping arithmetic and comparison operators to architecture-specific implementations and to carry chains (FLEX devices only). These parameters also allow you to control carry and cascade chains via MAX+PLUS II synthesis options.

Generating an EDIF Netlist File

Table 2 provides a quick reference for the commands needed to generate an EDIF netlist file using the Synopsys dc_shell or fpga_shell commands. For complete instructions on generating an EDIF netlist file, refer to the MAX+PLUS II ACCESS Key Guidelines or MAX+PLUS II Help.

Table 2. Commands for Generating an EDIF Netlist File

Command	Description
`read -f vhdl/verilog` <*design name*>	Analyzes and elaborates the design from the original source files.
`current_design=`<*top-level design name*>	Specifies the top-level design file.
`link`	Checks to make sure all descriptions of current designs are available in the Design Compiler or FPGA Compiler memory.
`ungroup -flatten -all`	Flattens the design.
`uniquify`	Removes a multi-instantiated hierarchy in the current design by creating a unique design for each cell instance.
`compile -m` <*effort level*>	Synthesizes the elaborated hardware description language (HDL) into look-up tables (LUTs).
`replace_fpga`	Converts FPGA cells to gate primitives for mapping in the MAX+PLUS II software. This command is not necessary if the `edifout_write_properties_list = LUT_FUNCTION` parameter is included in the **.synopsys_dc.setup** file.
`write -f edif -o` <*design name*>`.edf -h`	Generates an EDIF Input File (**.edf**) for MAX+PLUS II place-and-route.

After generating the EDIF netlist file, you can import it into the MAX+PLUS II software. In the MAX+PLUS II Compiler, remember to designate a target device in the **Device** dialog box (Assign menu). To designate Synopsys as the tool vendor used to create the EDIF Input File, choose **EDIF Netlist Reader Settings** (Interfaces menu) and select *Synopsys* in the *Vendor* drop-down list box. These selections ensure that the appropriate Library Mapping File (**.lmf**) is used during compilation.

Passing Timing Constraints from the Synopsys Design Environment to the MAX+PLUS II Software

Although Altera recommends that you make all timing assignments directly in the MAX+PLUS II software, you can use the **syn2acf** utility to pass timing constraints from a Synopsys Command File (**.cmd**) to a MAX+PLUS II Assignment & Configuration File (**.acf**). Table 3 provides sample commands that pass timing constraints from the Synopsys design environment to the MAX+PLUS II software.

For complete instructions on passing timing constraints, see the MAX+PLUS II ACCESS Key Guidelines.

Table 3. Commands for Passing Timing Constraints

Command	Description
`read -f vhdl/verilog` <*design name*>	Reads the design file.
`include timing.cmd`	Contains the Synopsys timing assignments.
`compile`	Compiles the design.
`current_design=`<*design name*>	Defines the current design.
`include /`<*path*>`/syn2acf.cmd`	Generates the files needed to pass timing constraints.
`sh /`<*path*>`/syn2acf`<*design name*>	Invokes the **syn2acf** utility.
`quit`	Instructs the Synopsys Design Complier or FPGA Compiler to quit.

When passing timing constraints from the Synopsys design environment to the MAX+PLUS II software, data is converted from the Synopsys CMD format to the MAX+PLUS II ACF format. Figure 2 shows sample timing constraints.

Figure 2. Timing Constraints Converted from Synopsys CMD Format to Altera ACF Format

Synopsys CMD Format

```
create_clock -period 50 -waveform{0 25} CLK
set_clock_skew -delay 2 CLK
set_input_delay 10 IN2
set_input_delay 5 -clock CLK IN1
set_output_delay 20 OUT2
set_output_delay 5 -clock CLK OUT1
set_max_delay 25 -to OUT1
set_max_delay 35 -to OUT2
set_multicycle_path 2 -to n20_reg
```

Altera ACF Format

```
TIMING POINT
BEGIN
"|OUT2": TCO= 15.00ns {synopsys};
"|IN1": TPD= 10.00ns {synopsys};
"|IN2": TPD= 5.00ns {synopsys};
"|OUT1": TCO= 20.00ns {synopsys};
"|IN1": TSU= 20.00ns {synopsys};
"|IN2": TSU= 117.00ns {synopsys};
"|CLK": FREQUENCY= 50.00ns {synopsys};
"|n10_reg": FREQUENCY= 100.00ns
{synopsys};
```

More Information

Altera provides extensive support documentation to help you successfully interface the MAX+PLUS II software with third-party EDA synthesis tools. For technical support, contact Altera Applications at (800) 800-EPLD. You can also e-mail your technical questions to Altera at **sos@altera.com**. For answers to common questions regarding the Altera / Synopsys interface, search Atlas Solutions on the Altera web site.

The following documents also provide more detailed information:

- MAX+PLUS II ACCESS Key Guidelines
- *MAX+PLUS II Programmable Logic Development System & Software Data Sheet*
- *EDA Software Support*
- *Technical Brief 48 (Passing Hierarchical Timing Constraints from Synopsys Tools to MAX+PLUS II Version 9.0)*

The documents are available by contacting Literature Services at (888) 3-ALTERA; you can also download them from the Altera web site at **http://www.altera.com**.

101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
http://www.altera.com