Using Magma PALACE™ Physical Synthesis to Improve ProASIC Performance

Introduction

Use Magma PALACE (Physical And Logical Automatic Compilation Engine) Actel Edition (AE) Physical Synthesis to optimize a netlist generated by other synthesis tools, such as Synopsys® Design Compiler®, Synplicity® Synplify®, and Mentor Graphics® LeonardoSpectrum and Precision®. PALACE optimizes the gate count (silicon area) and performance with respect to device resources and design constraints. Currently, PALACE AE is only supported for ProASIC^{PLUS}. However, this application note addresses how you can use PALACE AE as a logical synthesis tool for ProASIC designs.

Physical Flow vs. Logical Flow

PALACE AE supports two flows: Logical Effort and Physical Effort. ProASIC users can use only logical flow for design optimization. Refer to Table 1 for Logic Effort choices.

Table 1 • **Logic Effort Levels**

<value>	Explanation	Usage for ProASIC Designs
0	No optimization, no change in area	Not useful.
1	Combinatorial / sequential optimization with area focus	Used for area minimization
2	Combinatorial optimization mode	Used for area / timing improvement, where a flip-flop in the design must remain unchanged
3	Both combinatorial / sequential optimization performed, register balancing, and replicating	Used for timing optimization with timing constraints, or critical path optimization
4	Extensive optimization, exhausting all possible algorithms to achieve timing closure	Used for extensive timing optimization with timing constraints or critical path optimization

In Logical Effort, PALACE AE performs logic optimization, register balancing, retiming, and fanout control. In Physical flow, PALACE AE performs complete placement of the design.

ProASIC users can take advantage of logical optimization in PALACE AE for design improvement. Furthermore, timing constraints in SDC or GCF format can be used to guide PALACE AE for optimization.

Running PALACE AE

PALACE AE works primarily on netlists generated by a synthesis tool, such as Synplify or LeonardoSpectrum. PALACE AE uses constraint files to guide the optimization effort. ProASIC users must pass only timing constraints to PALACE AE. Figure 1 on page 2 shows a typical PALACE AE flow.

The PALACE AE inputs are:

- Input netlist – Netlist generated by any synthesis tool
- SDC timing constraint files

The PALACE AE outputs are:

- Optimized netlist – The optimized netlist generated by PALACE AE
- Timing constraint file – A forward-annotated constraint file for Designer place-and-route. This file consists of constraints you entered and some additional PALACE AE generated timing constraints.
- Report file – PALACE AE report file with warning info, any errors, etc.

In versions before Libero™ Integrated Design Environment (IDE) v6.0 and when using with Designer, running PALACE AE requires a command prompt on a PC or a UNIX shell. Refer to the *PALACE User's Guide* or PALACE AE online help for more information.

This is the command-line option for running a ProASIC design in PALACE AE:

PALACE AE_actel -family apa -in_design <netlist file> -in_constraint <constraint file> -logic_effort <value>

Because ProASIC and ProASIC^{PLUS} have similar resource information, you can use the ProASIC^{PLUS} library in PALACE AE to modify your ProASIC netlist. Use the "apa" family option and load the ProASIC^{PLUS} library in PALACE AE.

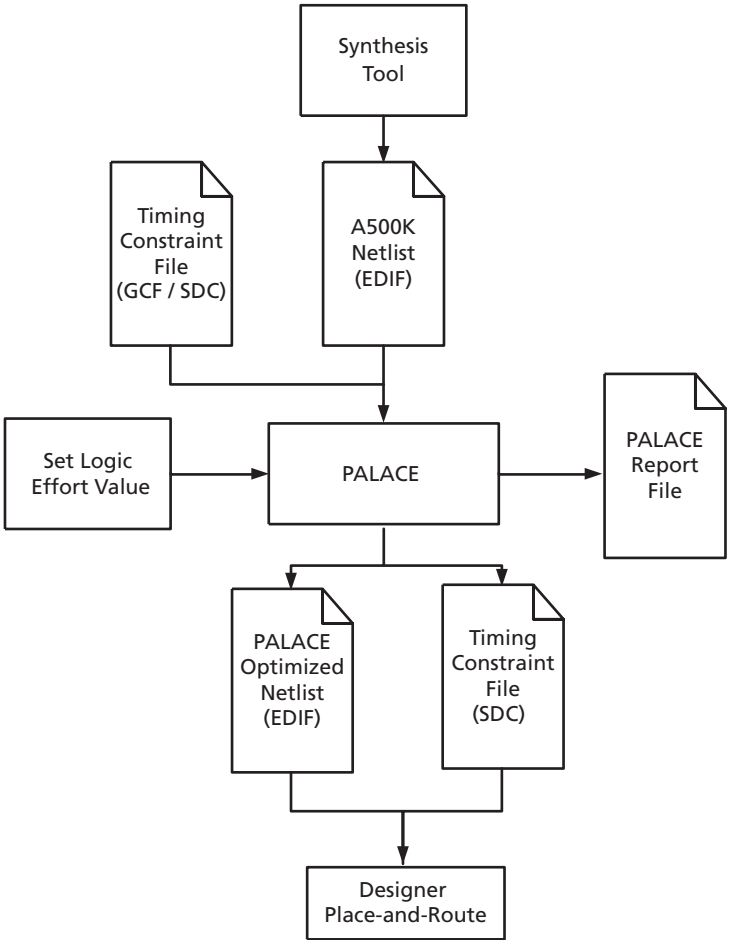

Figure 1 • **ProASIC PALACE AE Use Flow**

Example

The following example employs two clocks. Five post-layout runs (1-5 below) were performed to achieve the best performance.

1. No PALACE AE run
2. PALACE AE run with Logic Effort = 1
3. PALACE AE run with Logic Effort = 2
4. PALACE AE run with Logic Effort = 3 (with 100 MHz timing constraints)
5. PALACE AE run with Logic Effort = 4 (with 100 MHz timing constraints)

As the results indicate, running PALACE AE in logic_effort 1 can save in area (Table 1 on page 1). Running in performance mode, i.e., logic_effort mode, will improve performance at the expense of area (Table 2).

Table 2 • **ProASIC Results**

Name	No PALACE AE		LE = 1		LE = 2		LE = 3		LE = 4		Result
	f_{MAX}	Area Used	f_{MAX}	Area Used	f_{MAX}	Area Used	f_{MAX}	Area Used	f_{MAX}	Area Used	
design - clk1	55.96	27.4%	58.40	26.5%	70.9	29.2	77.86	33.9	86.9	35.4	clk1 improved 55%
design - clk2	70.09		58.56		70.36		77.71		74.8		clk2 improved 11%

Tuning Performance for the Right Clocks

Designs with multiple clocks could be optimized with realistic timing constraints on each clock. Actel strongly recommends that you do not overconstrain all clocks.

If there are critical clocks, set tight timing constraints on these clocks, and loosen the constraints on some of the noncritical clocks. To perform iterative optimization, tighten the constraints on the critical clocks by 10% more than the results. Perform this operation until you meet your constraints or the clock performance starts to decrease.

Optimization in Designer

Designer supports a limited set of SDC constraints for the ProASIC family. The constraints generated in PALACE AE must be imported as an auxiliary file. Designer translates most of the SDC constraints into the internal GCF format before performing place-and-route. Actel recommends that you tune the performance of the design within Designer by entering additional timing constraints in Timer. Iterating through the place-and-route with incremental changes to timing constraints may produce better results.

Related Documents

User's Guides

PALACE User's Guide

http://www.actel.com/documents/palace_UG.pdf

http://www.actel.com

Actel Corporation
2061 Stierlin Court
Mountain View, CA
94043-4655 USA
Phone 650.318.4200
Fax 650.318.4600

Actel Europe Ltd.
Dunlop House, Riverside Way
Camberley, Surrey GU15 3YL
United Kingdom
Phone +44 (0) 1276 401 450
Fax +44 (0) 1276 401 490

Actel Japan
EXOS Ebisu Bldg. 4F
1-24-14 Ebisu Shibuya-ku
Tokyo 150 Japan
Phone +81.03.3445.7671
Fax +81.03.3445.7668

Actel Hong Kong
39th Floor, One Pacific Place
88 Queensway, Admiralty
Hong Kong
Phone +852.227.35712
Fax +852.227.35999

51900073-0/06.04