

ALTERA®

Quartus® II Software Design Series: Foundation



Objectives

- Create a new Quartus® II project
- Choose supported design entry methods
- Compile a design into an FPGA
- Locate resulting compilation information
- Create design constraints (assignments & settings)
- Manage I/O assignments
- Perform timing analysis & obtain results

Class Agenda

- Intro to Altera & Devices
- Quartus II Feature Overview
- Design Methodology
- Projects
 - Exercise 1
- Design Entry
 - Exercise 2
- Compilation
 - Exercise 3
- Settings & Assignments
 - Exercise 4
- I/O Planning
 - Exercise 5
- Timing Analysis
 - Exercise 6



Quartus II Software Design Series: Foundation

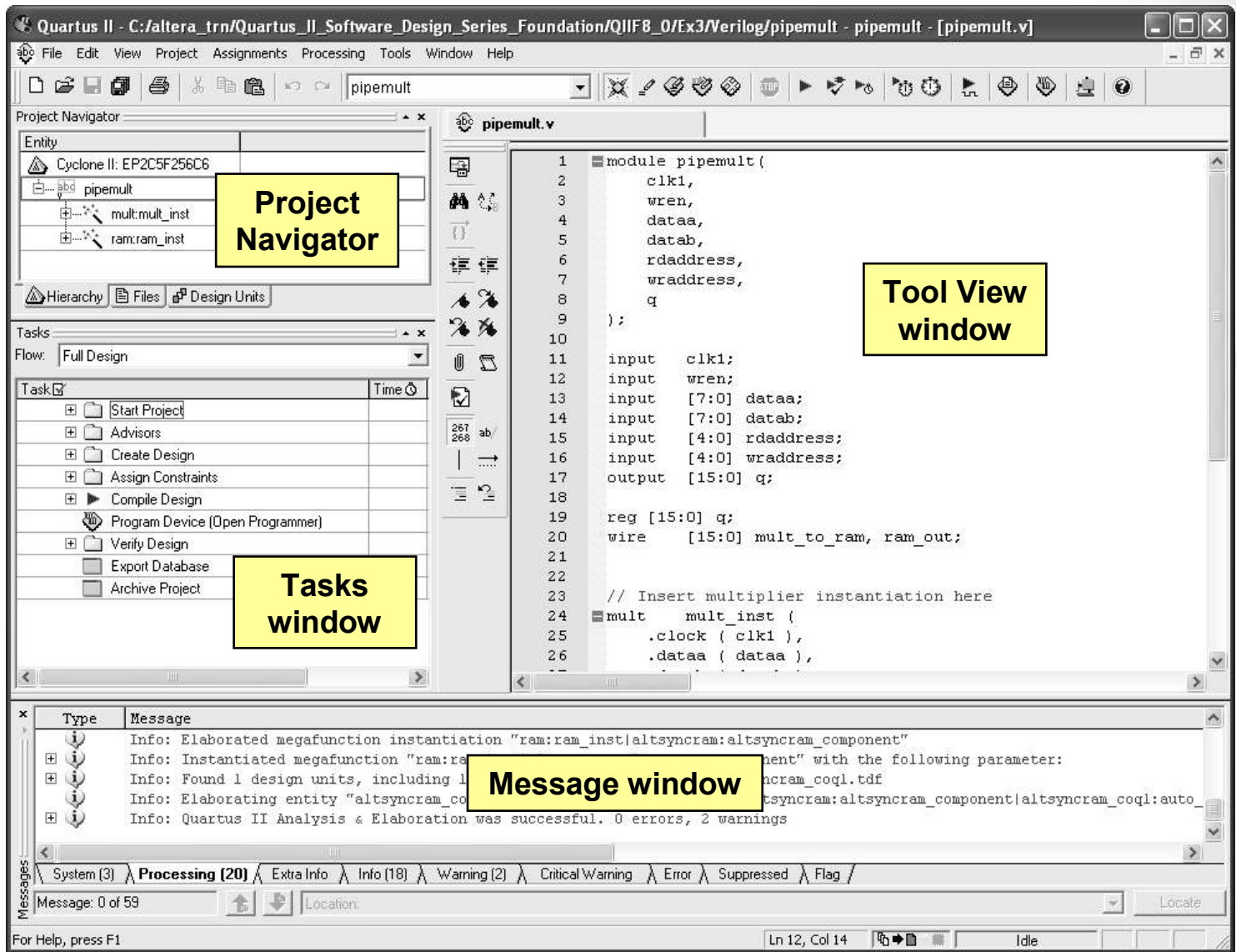
Quartus II Design Software Feature Overview



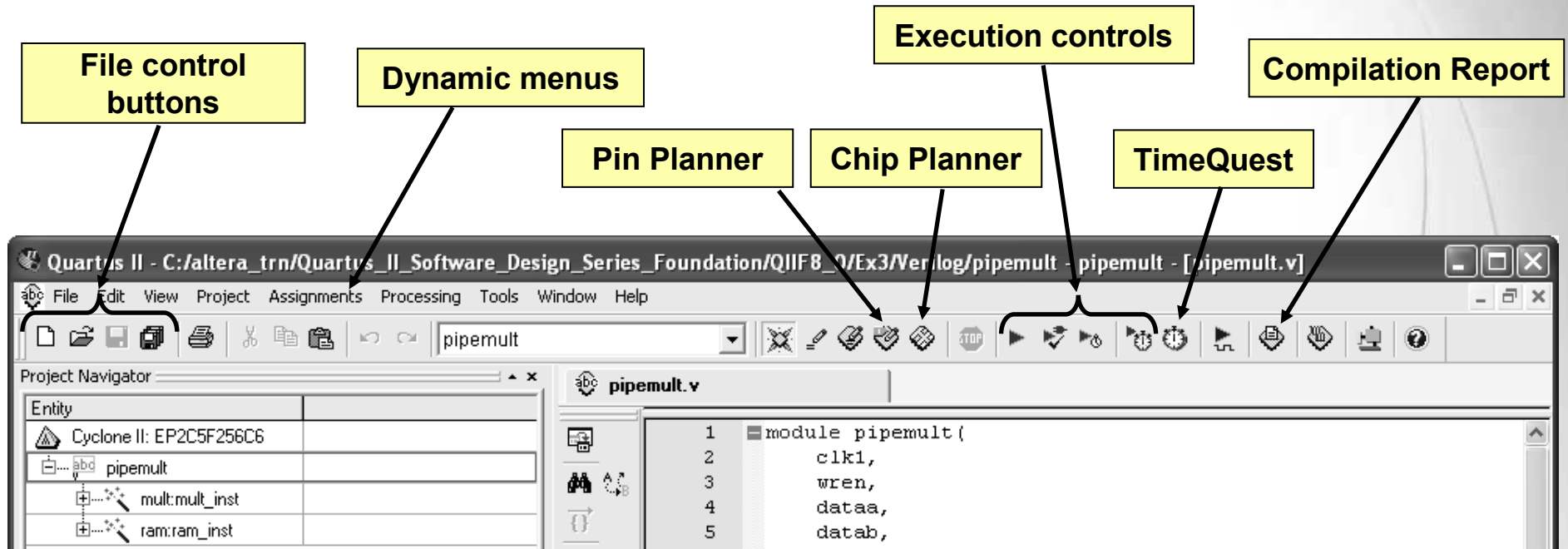
Quartus II Design Software

- Fully-integrated development tool
 - Multiple design entry methods
 - Logic synthesis
 - Place & route
 - Simulation
 - Timing & power analysis
 - Device programming

Quartus II Default Operating Environment



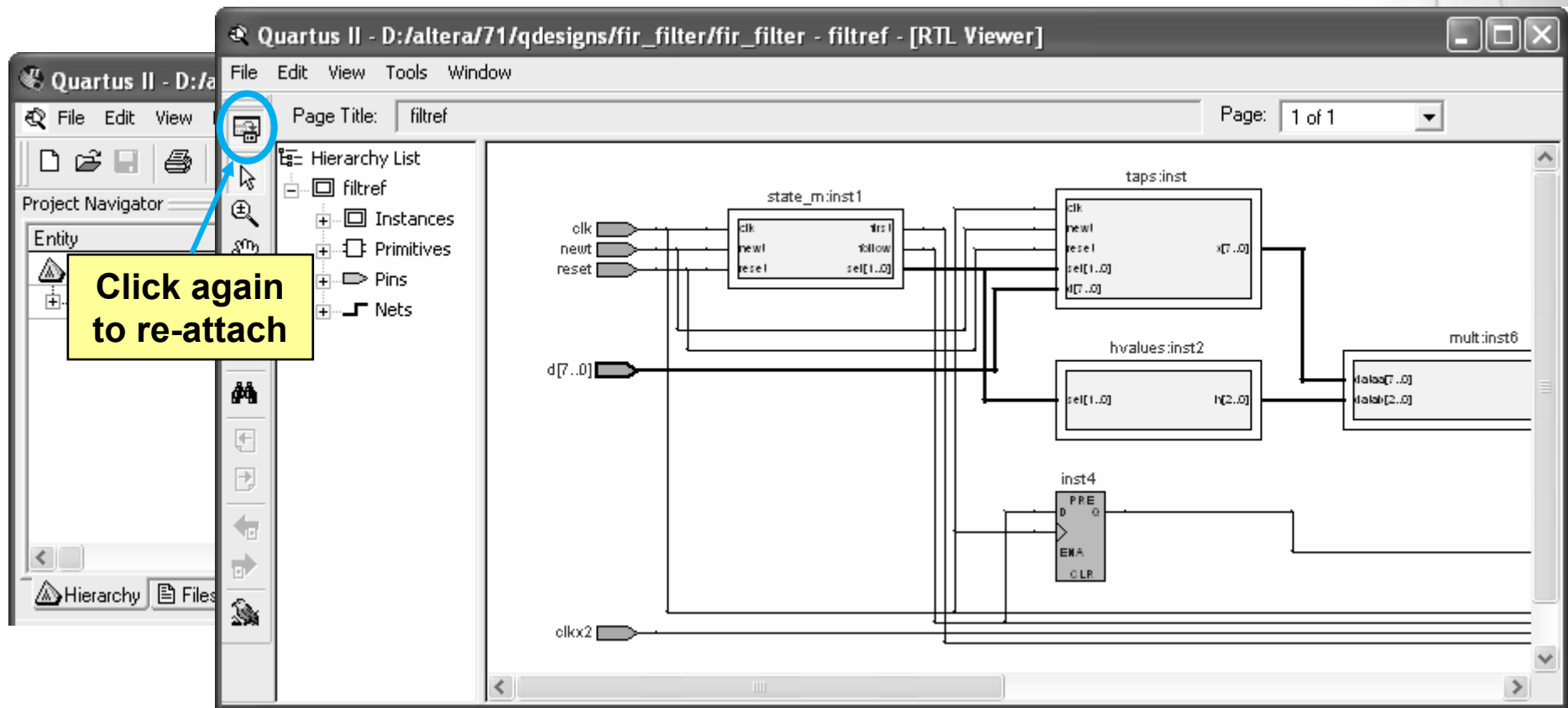
Main Toolbar



To reset views:
1. Tools ⇒ Customize ⇒ Toolbars ⇒ Reset All
2. Restart Quartus II

Detachable Windows

- Separate child windows from the Quartus II GUI frame (**Window** menu ⇒ **Detach/Attach Window**)



Tasks Window

- Easy access to most Quartus II functions
- Organized into related tasks within two task flows

Full Design Flow
Perform all project tasks

Compilation Flow
Focus on compilation tasks

Tasks: Flow: Full Design

Task	Time
Start Project	
Open New Project Wizard	
Open Existing Project	
Create Revision	
Specify Project Libraries	
Import Database	
Advisors	
Create Design	
Assign Constraints	
Compile Design	
Program Device (Open Programmer)	
Verify Design	
Export Database	
Archive Project	

Tasks: Flow: Compilation

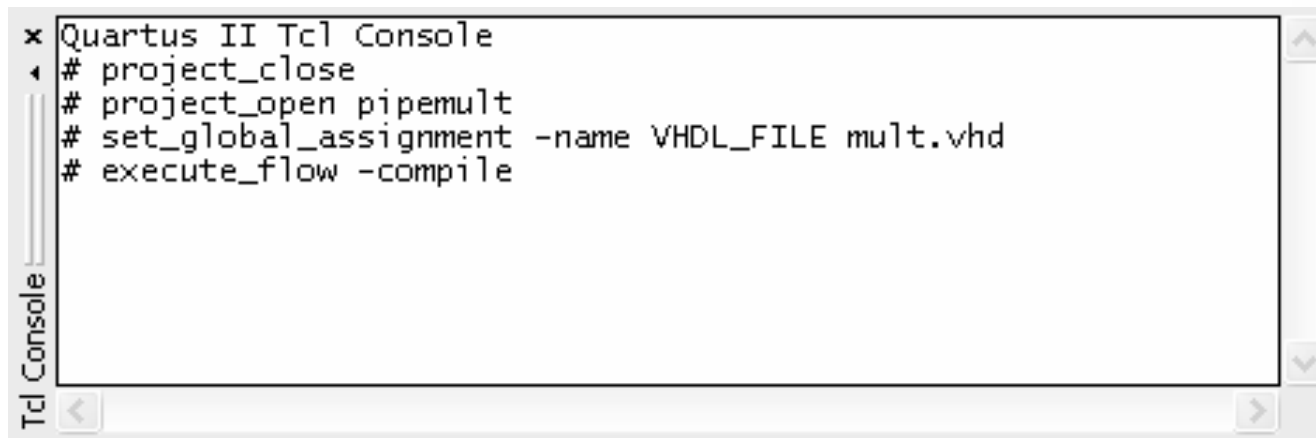
Task	Time
Compile Design	
Analysis & Synthesis	
Fitter (Place & Route)	
Assembler (Generate programming files)	
Classic Timing Analysis	
EDA Netlist Writer	
Program Device (Open Programmer)	

Double-click any task to run

Tcl Console Window

- Enter and execute Tcl commands directly in the GUI

View menu ⇒ Utility Windows ⇒ Tcl Console



```
Quartus II Tcl Console
# project_close
# project_open pipemult
# set_global_assignment -name VHDL_FILE mult.vhd
# execute_flow -compile
```

- Execute from command-line using Tcl shell

- quartus_sh --shell

Tips & Tricks Advisor

Help menu ⇒ Tips & Tricks

Get an Early Timing Estimate	
Recommendation	You can get an early timing estimate without running a full compilation.
Description	You can use the Start Early Timing Estimate command on the Processing menu to get a full timing report based on estimated delays for the design. This command can run the Fitter up to ten times faster than a full fit and produces estimated delays within 20% of what a full compilation can achieve.
Action	Use the Start Early Timing Estimate command on the Processing menu to run an early timing estimate. You can specify settings for the early timing estimate in the Settings dialog box when a project is open. Open Settings dialog box - Early Timing Estimate page

Provides useful instructions on using the Quartus II software & links to settings. Available sections include:

- New features in current release
- Helpful features and project settings available to designers

For Help, press F1

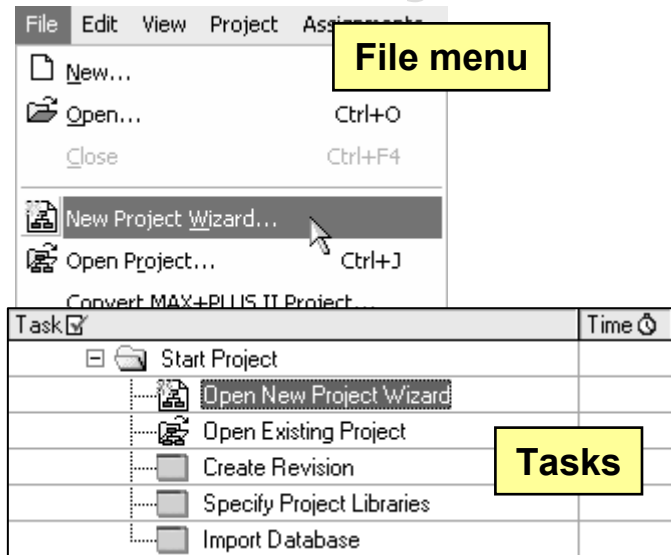


Quartus II Software Design Series: Foundation

Quartus II Projects



New Project Wizard



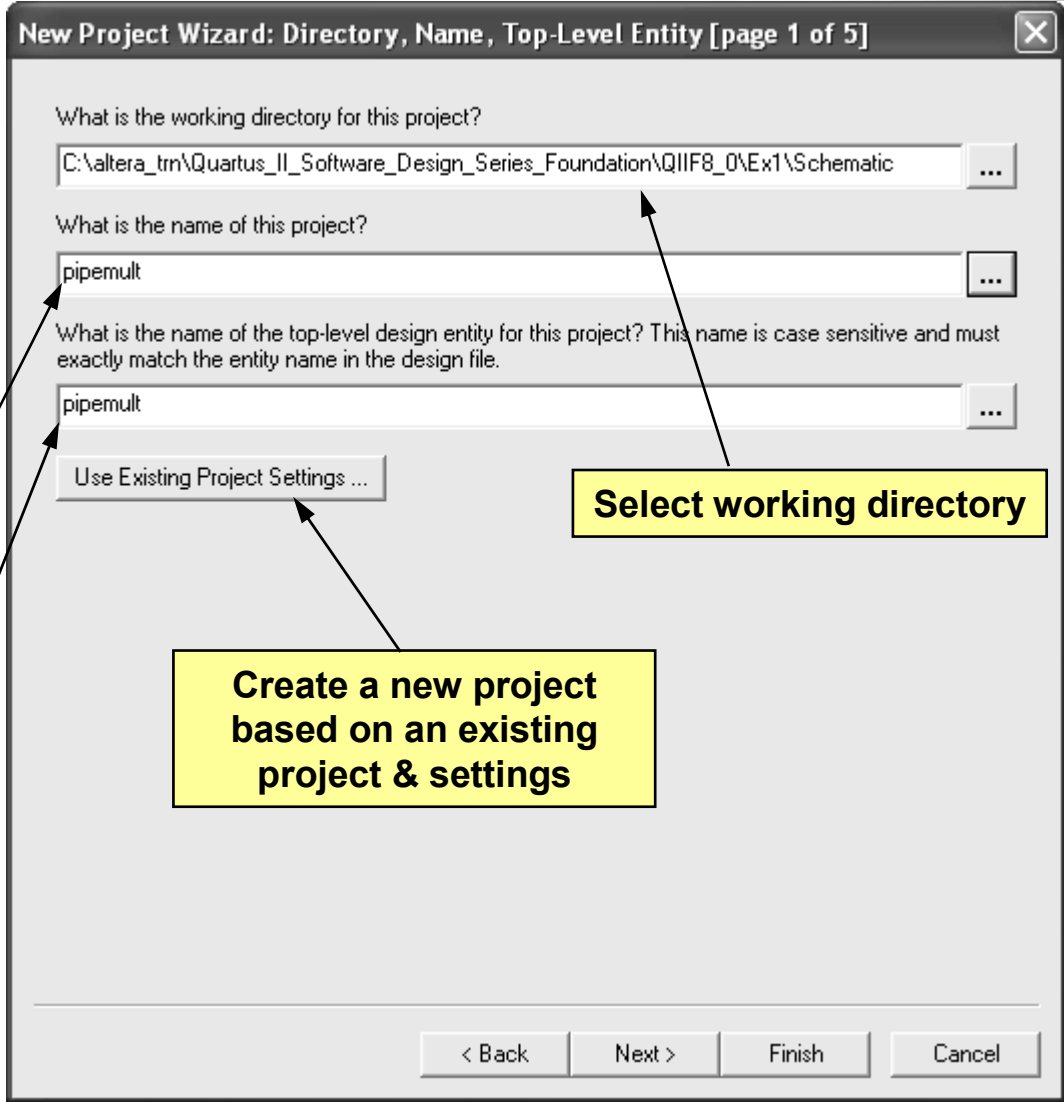
File menu

Tasks

Name of project can be any name; recommend using top-level file name

Top-level entity does not need to be the same name as top-level file name

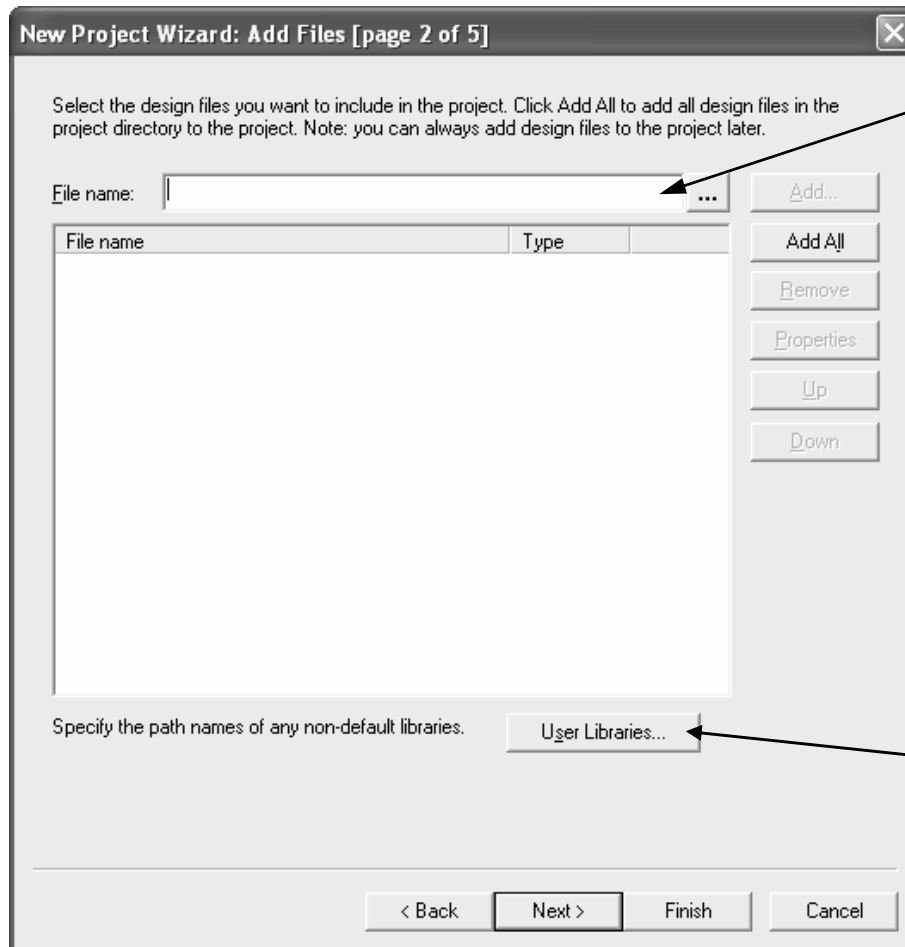
Tcl: `project_new <project_name>`



Select working directory

Create a new project based on an existing project & settings

Add Files



Add design files

- Graphic (.BDF)
- AHDL
- VHDL
- Verilog
- EDIF

Notes:

- *Files in project directory do not need to be added*
- *Add top-level file if filename & entity name are not the same*
- *Absolute & relative paths are supported*

Add user library pathnames

- User libraries (any directory containing files)
- MegaCore®/AMPPSM libraries
- Pre-compiled VHDL packages

```
Tcl: set_global_assignment -name VHDL_FILE* <filename.vhd>  
Tcl: set_global_assignment -name USER_LIBRARIES <library_path_name>  
* Replace with VERILOG_FILE, EDIF_FILE, AHDL_FILE or BDF_FILE
```

Device Selection

New Project Wizard: Family & Device Settings [page 3 of 5]

Select the family and device you want to target for compilation.

Device family:
 Family: Cyclone III
 Devices: All

Target device:
 Auto device selected by the Fitter
 Specific device selected in 'Available devices' list

Show in 'Available device' list:
 Package: FBGA
 Pin count: 256
 Speed grade: Any

Show advanced devices
 HardCopy compatible only

Available devices:

Name	Core v...	LEs	User I/...	Memor...	Embed...	PLL
EP3C5F256A7	1.2V	5136	183	423936	46	2
EP3C5F256C6	1.2V	5136	183	423936	46	2
EP3C5F256C7	1.2V	5136	183	423936	46	2
EP3C5F256C8	1.2V	5136	183	423936	46	2
EP3C5F256I7	1.2V	5136	183	423936	46	2
EP3C10F256A7	1.2V	10320	183	423936	46	2
EP3C10F256C6	1.2V	10320	183	423936	46	2
EP3C10F256C7	1.2V	10320	183	423936	46	2
EP3C10F256C8	1.2V	10320	183	423936	46	2

Companion device:
 HardCopy:
 Limit DSP & RAM to HardCopy device resources

< Back Next > Finish Cancel

Choose device family and filter results

Advanced information on future devices

Choose specific part number from list or let Fitter choose smallest, fastest device based on filter criteria

```
Tcl: set_global_assignment -name FAMILY "device family name"
Tcl: set_global_assignment -name DEVICE <part_number>
```



EDA Tool Settings

**Choose EDA tools
& file formats**

**Add or change
settings later**

New Project Wizard: EDA Tool Settings [page 4 of 5]

Specify the other EDA tools -- in addition to the Quartus II software -- used with the project.

Design Entry/Synthesis

Tool name: Synplify Pro

Format: VQM

Run this tool automatically to synthesize the current design

Simulation

Tool name: ModelSim-Altera

Format: VHDL

Run gate level simulation

VHDL
Verilog

Timing Analysis

Tool name: PrimeTime

Format: Verilog

Run this tool automatically after compilation

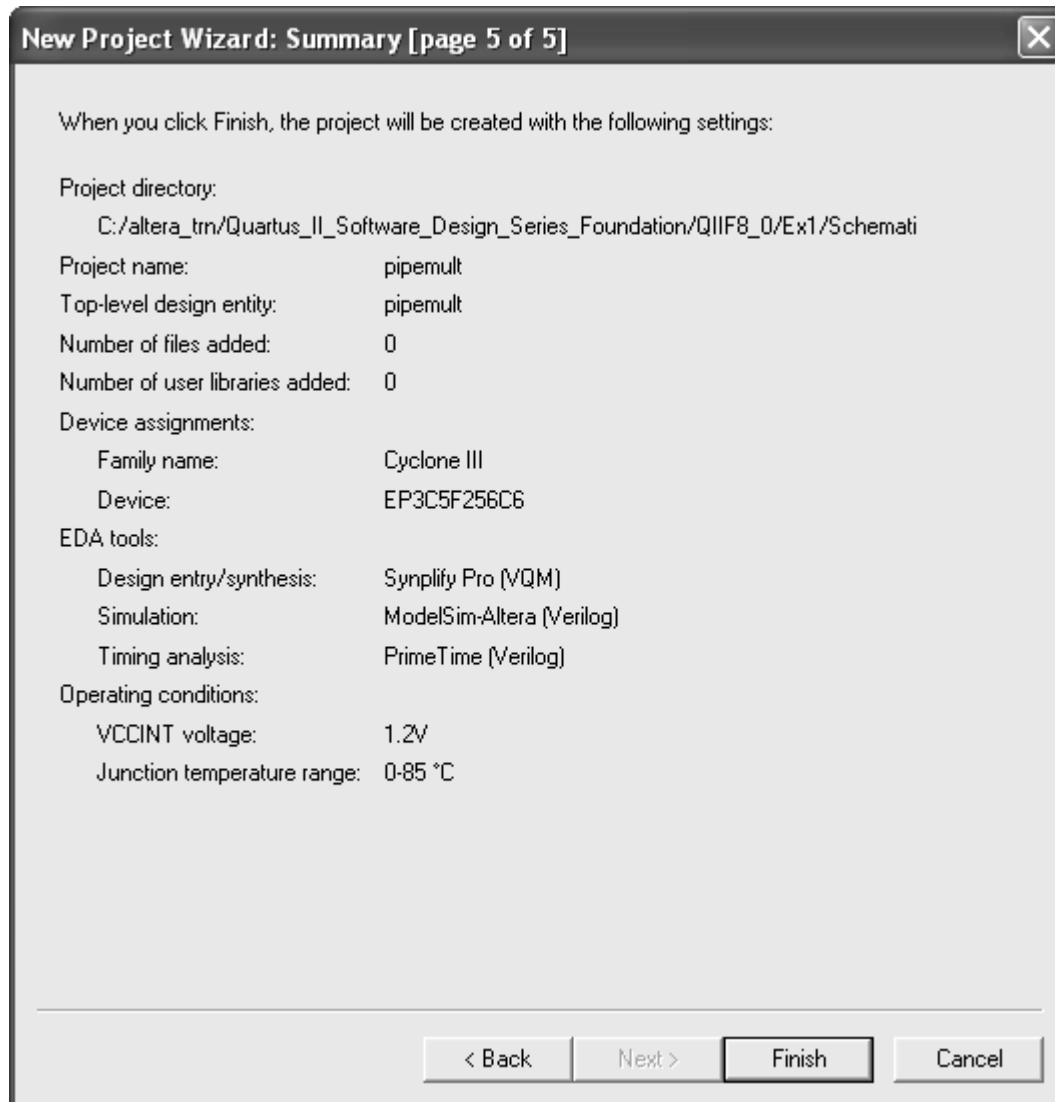
< Back Next > Finish Cancel

See handbook for Tcl command format

© 2008 Altera Corporation—Confidential

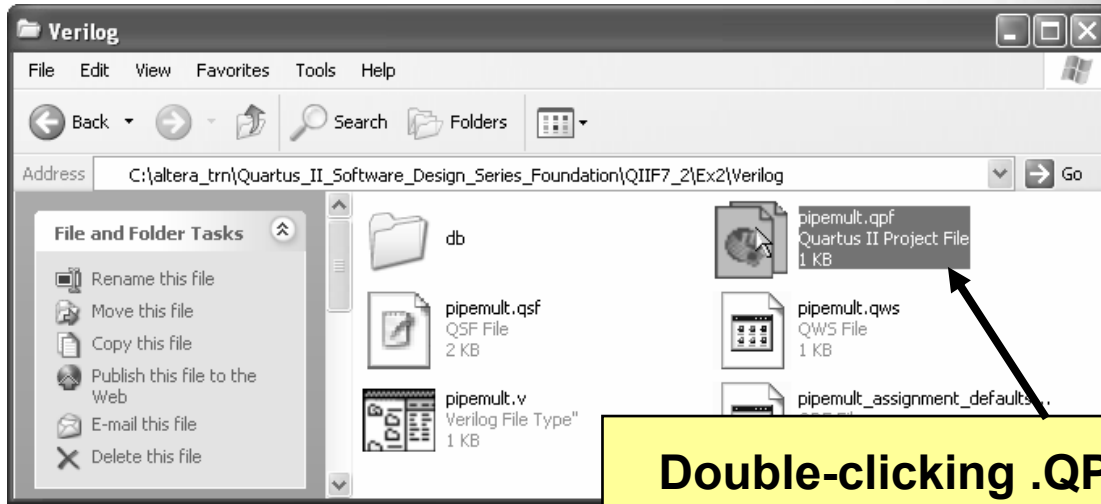
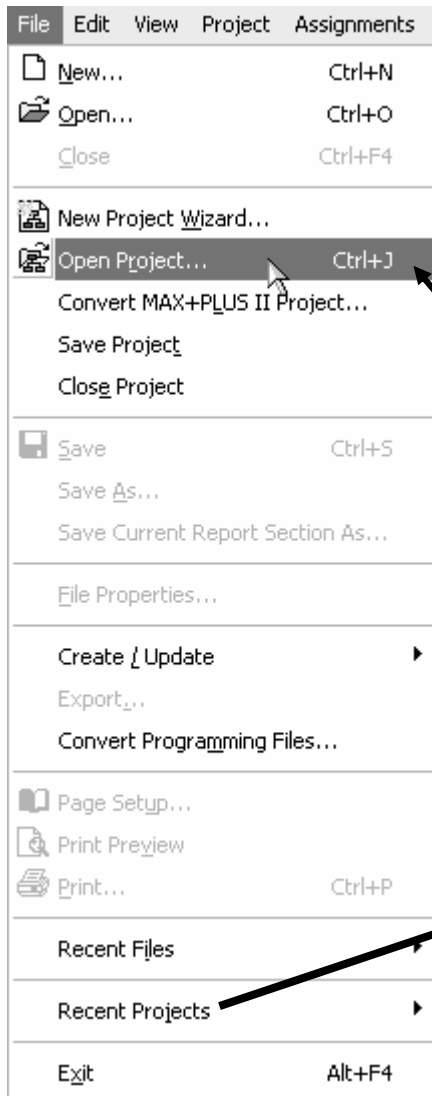
Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation

Done!



**Review results &
click Finish**

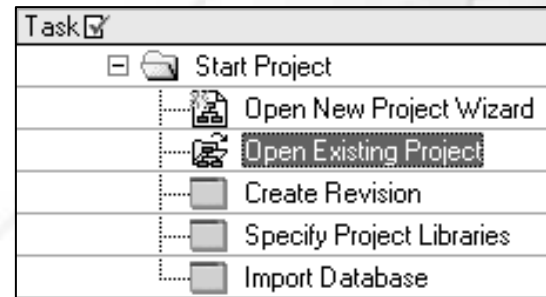
Opening an Existing Project



File => Open Project... OR

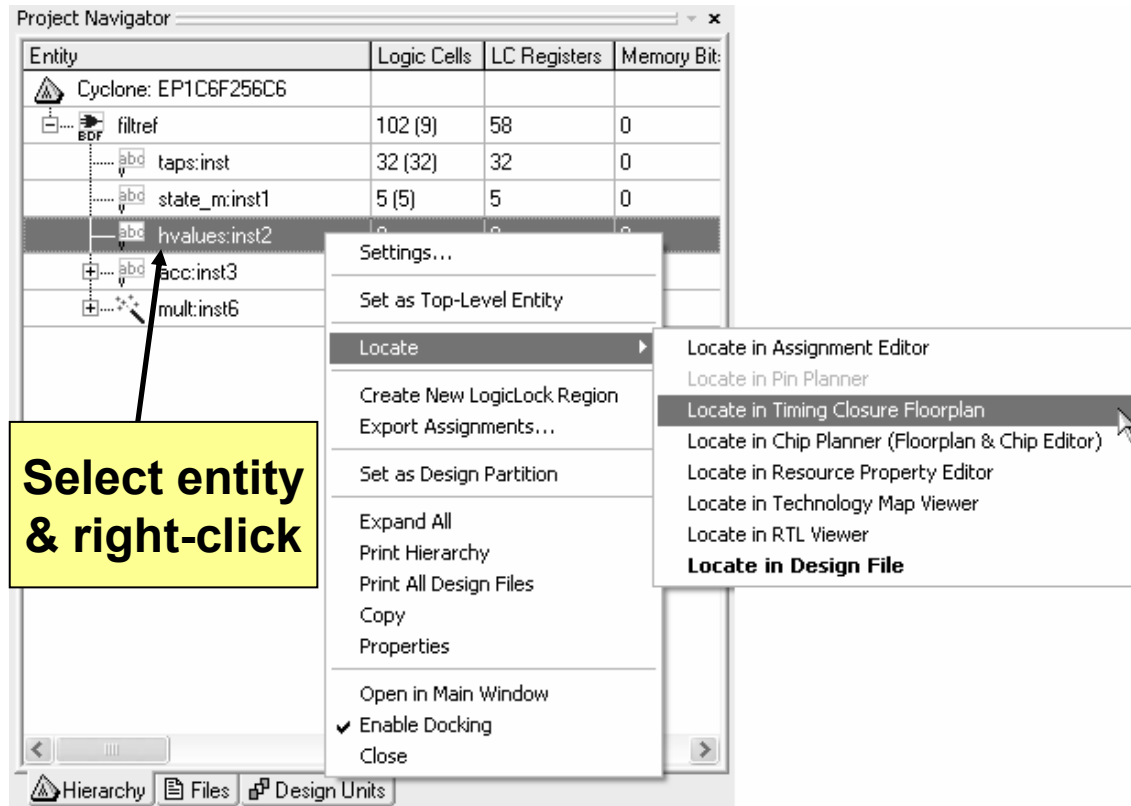
Select from most recent projects list OR Tasks window

- 1 D:\altera\71\qdesigns\QIIF7_1\Ex7\couner_dsp\top_counter.qpf
- 2 D:\altera\71\qdesigns\QIIF7_1\Ex7\counter_nios\top_counter.qpf
- 3 D:\altera\71\qdesigns\fir_filter\fir_filter.qpf
- 4 D:\altera\71\qdesigns\my_new_project\my_project.qpf
- 5 D:\altera\71\qdesigns\QIIF7_1\Ex2\WHDL\pipemult.qpf



```
Tcl: project_open <project_name>
```

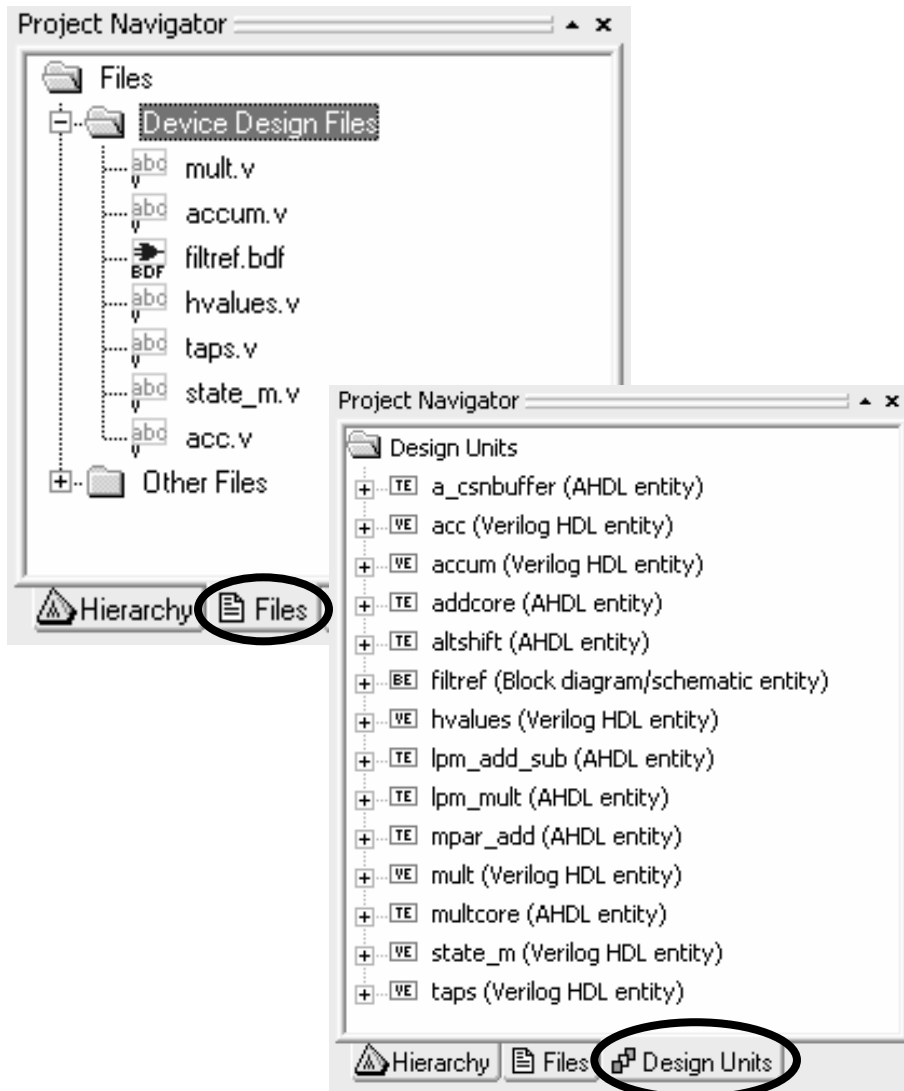
Project Navigator – Hierarchy Tab



- Displays project hierarchy after project is analyzed
- Uses
 - Set top-level entity
 - Set incremental design partition
 - Make entity-level assignments
 - Locate in design file or viewers/floorplans
 - View resource usage

Full compilation or Processing menu ⇒ Start ⇒ Start Analysis & Elaboration

Files & Design Units Tabs



■ Files tab

- Shows files explicitly added to project
- Uses
 - Open files
 - Remove files from project
 - Set new top-level entity
 - Specify VHDL library
 - Select file-specific synthesis tool
- Can also use **Project ⇒ Add/Remove Files in Project...**

■ Design Units tab

- Displays design unit & type
 - VHDL entity
 - VHDL architecture
 - Verilog module
 - AHDL subdesign
 - Block diagram filename
- Expanded unit displays file which instantiates design unit

Quartus II Project Files & Folders

- Quartus II Project File (.QPF)
- Quartus II Defaults File (.QDF)
- Quartus II Settings File (.QSF)
- db folder
 - Contains compiled design information

- Synopsys Design Constraints (.SDC)
 - Holds timing constraints
 - Discussed later

Project & Default Files

■ Quartus II Project File (QPF)

- Quartus II version
- Time stamp
- Active revision(s)

fir_filter.QPF

```
QUARTUS_VERSION = "8.0"  
DATE = "08:37:10  June 19, 2008"  
  
# Active Revisions  
  
PROJECT_REVISION = "filtref"  
PROJECT_REVISION = "filtref_new"
```

■ Quartus II Defaults Files (QDF)

- Stores Quartus II project setting & assignment defaults
- Example names: *assignment_defaults.qdf* or *<revision_name>_assignment_defaults.qdf*
- Found in local project or *altera\<version>\quartus\bin* directory
 - Copy stored in local project directory read before original version in bin

Quartus II Settings File (QSF)

- Stores all settings & assignments (constraints)
- Uses Tcl syntax
- Can be edited manually by user

See “Quartus II Settings File Reference Manual” for more details on QSF assignments & syntax

Add user comments (#) & white space

Reorganize QSF based on categories (Project menu)

Source other TCL/QSF files to organize assignments

NEW assignments added to end of file

The screenshot shows the Quartus II IDE with a QSF file open. The file content is as follows:

```
21 # Altera recommends that you do not modify this file. This
22 # file is updated automatically by the Quartus II software
23 # and any changes you make may be lost or overwritten.
24
25
26
27 # Project-Wide Assignments
28 # =====
29 set_global_assignment -name ORIGINAL_QUARTUS_VERSION 7.1
30 set_global_assignment -name PROJECT_CREATION_TIME_DATE "08:37:10 APRIL 10, 2007"
31 set_global_assignment -name LAST_QUARTUS_VERSION 7.1
32 set_global_assignment -name VERILOG_FILE mult.v
33 set_global_assignment -name VERILOG_FILE accum.v
34 set_global_assignment -name BDF_FILE filtref.bdf
35 set_global_assignment -name VERILOG_FILE hvalues.v
36 set_global_assignment -name VERILOG_FILE taps.v
37 set_global_assignment -name VERILOG_FILE state_m.v
38 set_global_assignment -name VERILOG_FILE acc.v
39 set_global_assignment -name SMART_RECOMPILE ON
40 set_global_assignment -name VECTOR_WAVEFORM_FILE fir.vwf
41
42 # This is where my pin assignments are located
43 # =====
44 set_location_assignment PIN_G1 -to clk
45 source "location_assignments.tcl"
46
47 # Classic Timing Assignments
48 # =====
49 set_global_assignment -name FMAX_REQUIREMENT "85 MHz"
50
51 # Analysis & Synthesis Assignments
52 # =====
53 set_global_assignment -name FAMILY Cyclone
54 set_global_assignment -name TOP_LEVEL_ENTITY filtref
55 set_global_assignment -name DEVICE_FILTER_PACKAGE FBGA
```

Annotations in the image include:

- A yellow box pointing to the header comment lines (lines 21-23): "Add user comments (#) & white space".
- A yellow box pointing to the "Project-Wide Assignments" section (lines 27-40): "Reorganize QSF based on categories (Project menu)".
- A yellow box pointing to the "source" command (line 45): "Source other TCL/QSF files to organize assignments".
- A yellow box pointing to the "Classic Timing Assignments" section (lines 47-50): "NEW assignments added to end of file".

Note: See Appendix for more notes on using QSF file.



Project Archive

- Creates 2 files
 - Compressed Quartus II Archive File (.QAR)
 - Includes design files, QPF file, & QSF file(s)
 - Option to include databases (db folder in project directory)
 - Recompile necessary if databases not included
 - Creates local QDF file for archive
 - Archive activity log (.QARLOG)
- Example uses
 - File storage (e.g. version control)
 - Project handoff
 - Useful for sending to Altera support
- Design files referenced from user libraries are included in archive

```
Tcl: project_archive <project_name>
```

Project Archive (cont.)

Project Menu or Tasks window

- Task
- Start Project
- Advisors
- Create Design
- Assign Constraints
- Compile Design
- Program Device (Open Programmer)
- Verify Design
- Export Database
- Archive Project**

Archive Project

Specify a Quartus II Archive File for the current project. The Quartus II software automatically archives your source design and project files; the options below allow you to include other files. The Including version-compatible database files option takes additional time to archive the project.

Archive file name:
filtref

Archive current active revision only

Include the following optional database files

- No database files included (Recommended)
- Compilation and simulation database files (For current versions of the Quartus II software)**
- Version-compatible database files (For future versions of the Quartus II software)
- Include both kinds of database files

Include functions from system libraries

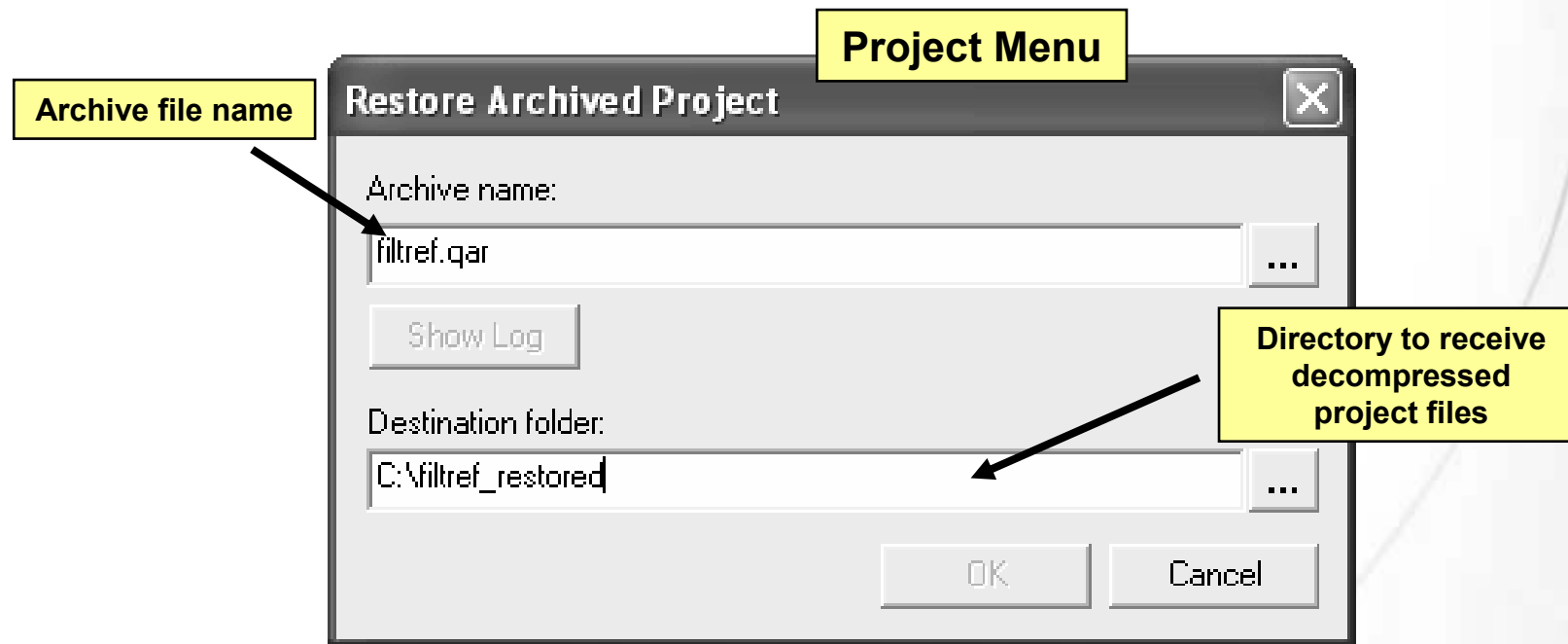
Add/Remove Files... OK Cancel

Database inclusion

View files to be included in archive and select files to add to or remove from archive

Project Restore

- Decompresses .QAR into specified directory



```
Tcl: project_restore <archive_file>
```



Quartus II Software Design Series: Foundation

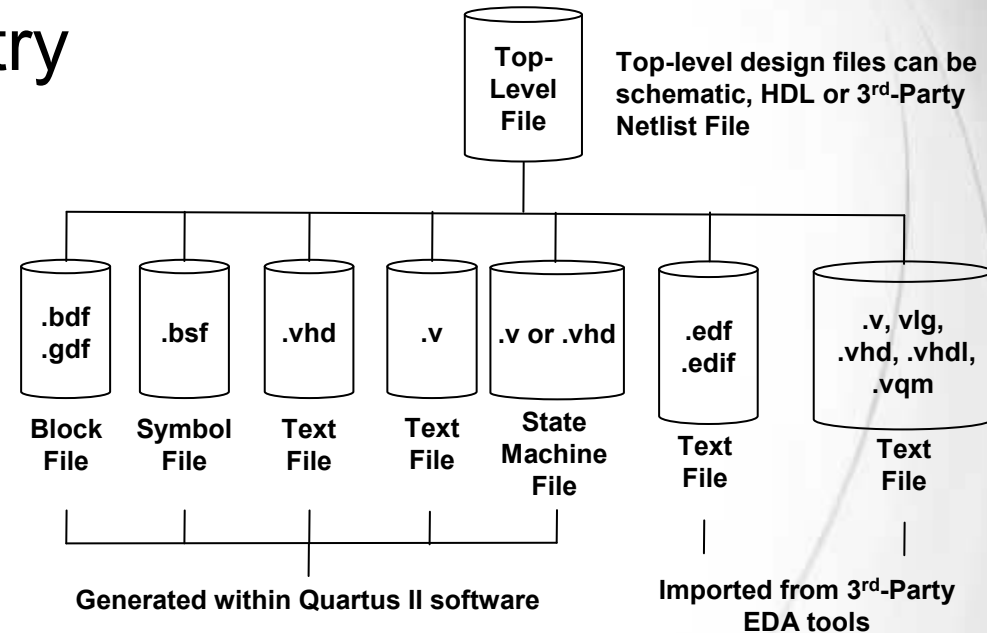
Design Entry



Design Entry Methods

■ Quartus II design entry

- Text editor
 - VHDL
 - Verilog
- Schematic editor
 - Block Diagram File
 - Graphic Design File
- State machine editor
 - HDL from state machine file
- Memory editor
 - HEX
 - MIF




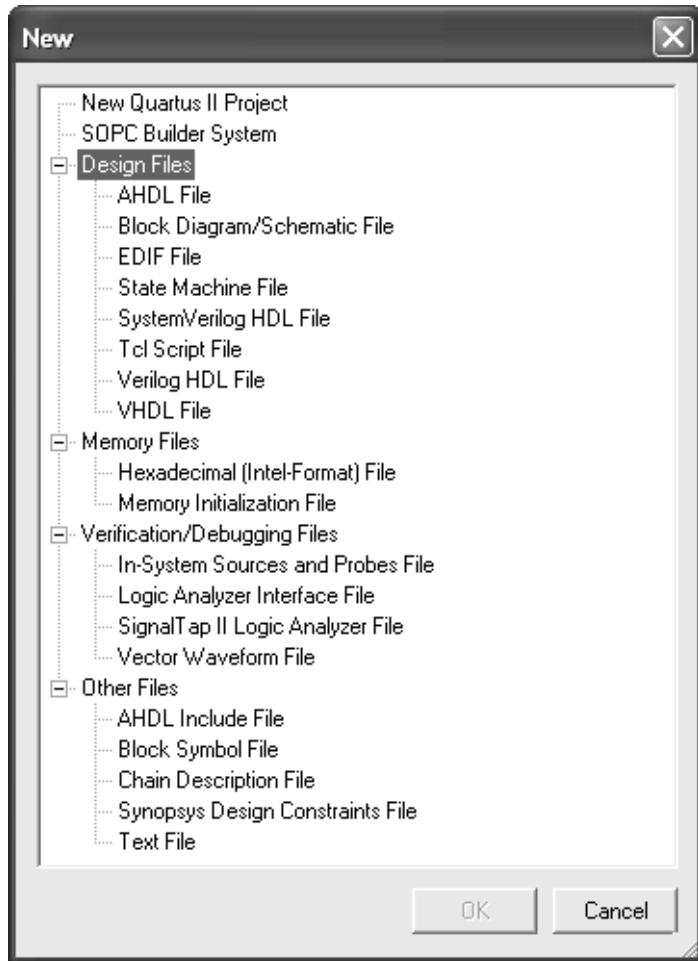
■ 3rd-party EDA tools

- EDIF 2 0 0
- Verilog Quartus Mapping (.VQM)

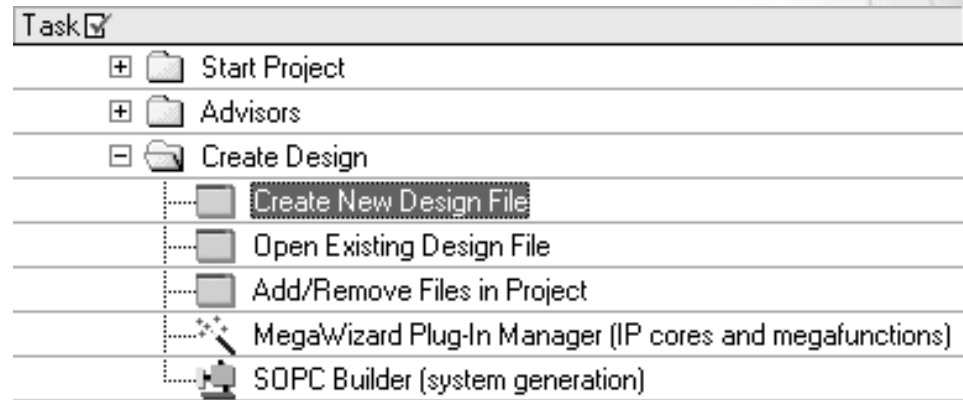
■ Mixing & matching design files allowed

Creating New Design Files (& Others)

File ⇒ New or  in Toolbar



Tasks window



**Create new files or start
New Project Wizard**

Verilog & VHDL

- **VHDL**- VHSIC hardware description language
 - IEEE Std 1076 (1987 & 1993) supported
 - IEEE Std 1076.3 (1997) synthesis packages supported
- **Verilog**
 - IEEE Std 1364 (1995 & 2001) & 1800 (SystemVerilog) supported
- Create in the Quartus II editor or any standard text editor
 - Select different text editor app with **Tools** ⇒ **Options for Text Editor**
- Use Quartus II integrated synthesis to synthesize
- View supported commands in on-line help

***Learn more about HDL in Altera HDL
customer training classes***

Text Editor Features

The image shows the Quartus II text editor interface with several callout boxes highlighting specific features:

- Find/highlight matching delimiters:** A yellow box with an arrow pointing to a search bar in the editor.
- Collapse/expand functions:** A yellow box with an arrow pointing to a tree view on the left side of the editor.
- Insert Template (Edit menu):** A yellow box with an arrow pointing to the 'Insert Template' dialog box.
- Bookmarks (on/off/jump to):** A yellow box with an arrow pointing to a bookmark icon in the left margin.
- Preview window: edit before inserting & save as user template:** A yellow box with an arrow pointing to the 'Preview' window of the 'Insert Template' dialog.

The 'Insert Template' dialog box shows a tree view of language templates and a preview window. The 'Save User Template' dialog box is also visible, showing a text field for the user template name.

Schematic Design Entry

- Full-featured schematic design capability
- Schematic Editor uses
 - Create simple test designs to understand the functionality of an Altera megafunction
 - PLL, LVDS I/O, memory, etc...
 - Create top-level schematic for easy viewing & connection
 - Convert Block Diagram File (.BDF) to HDL file (VHDL/Verilog) or image file (.JPG or .BMP) (**File** ⇒ **Create/Update**)
 - Convert HDL file to schematic block symbol file (.BSF; black box) (**File** ⇒ **Create/Update**)

*Note: Please see the Appendix for a more detailed discussion of the Block Diagram Editor and schematic entry.
Online training available: [Using the Quartus II Software: Schematic Design](#)*

© 2008 Altera Corporation—Confidential

Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation

32

The Altera logo is located in the bottom right corner of the slide. It consists of the word "ALTERA" in a bold, uppercase, sans-serif font. The letters are white with a black outline, and the entire logo is set against a dark, semi-transparent background.

Altera Megafunctions

- Pre-made design blocks
- Benefits
 - Configurable settings add flexibility
 - “Drop-in” support to accelerate design entry
 - Pre-optimized for Altera architecture
- Two versions
 - Quartus II megafunctions
 - Intellectual Property (IP) megafunctions

Quartus II Megafunctions

- Free & installed with Quartus II software
 - Non-encrypted functions written in AHDL
 - HDL simulation models installed in Quartus II libraries
- Two types
 - Altera-specific megafunctions (begin with “ALT”)
 - Library of parameterized modules (LPMs)
 - Industry standard logic functions
 - See www.edif.org/lpmweb (EDIF.org archive) for more info
- Examples
 - Multiply-accumulate (ALTMULT_ACCUM)
 - On-chip RAM/ROM (ALTSYNCRAM)
 - PLL (ALTPLL)
 - DDR/QDR memory interface (ALTMEMPHY)
 - Counter (LPM_COUNTER)
 - Comparator (LPM_COMPARE)

IP Megafunctions

- Must purchase license to use in finished design
 - Logic for IP function is encrypted
- Two types
 - MegaCore® IP
 - Developed by Altera
 - Install with Quartus II software or download/install individually from www.altera.com
 - Altera Megafunctions Partner Program (AMPP™) IP
 - Developed by 3rd-Party IP vendors & certified by Altera
 - Contact vendor for evaluating and licensing function
- All MegaCore functions & some AMPP functions support OpenCore® Plus feature
 - Develop design using free version of core
 - HDL simulation models provided with IP
 - Generate time-limited configuration/programming files
 - See [AN320: OpenCore Plus Evaluation of Megafunctions](#)

Example MegaCore IP

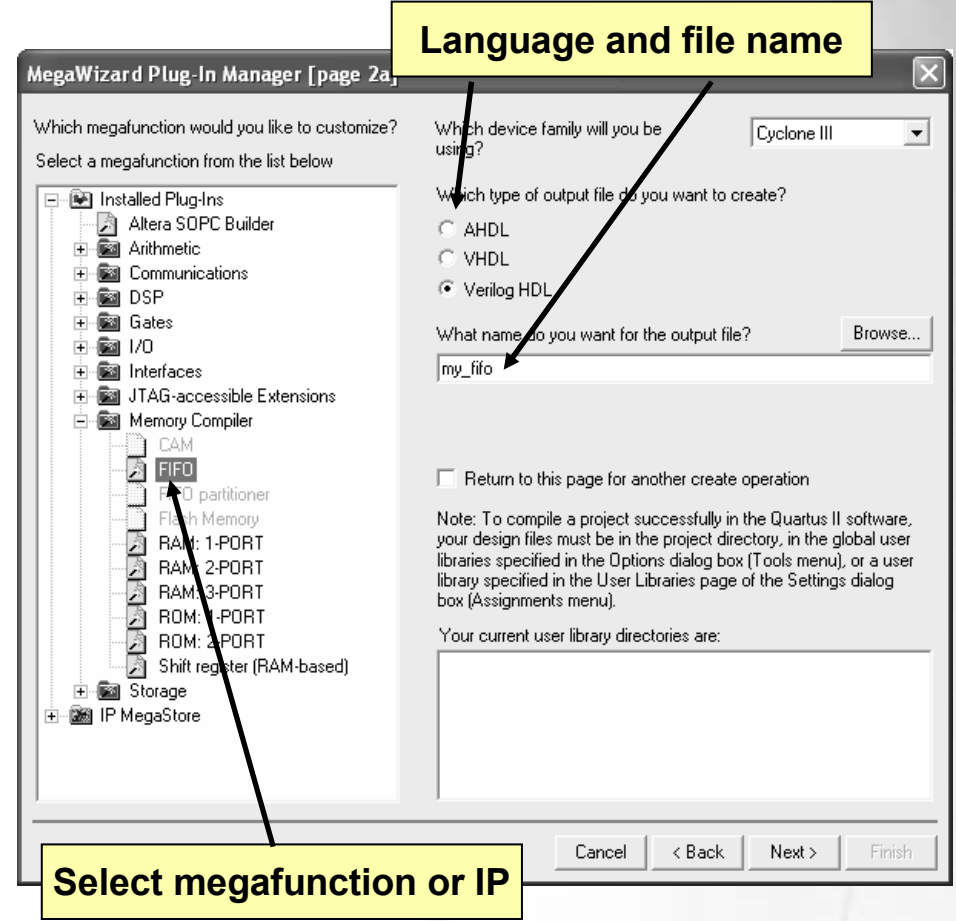
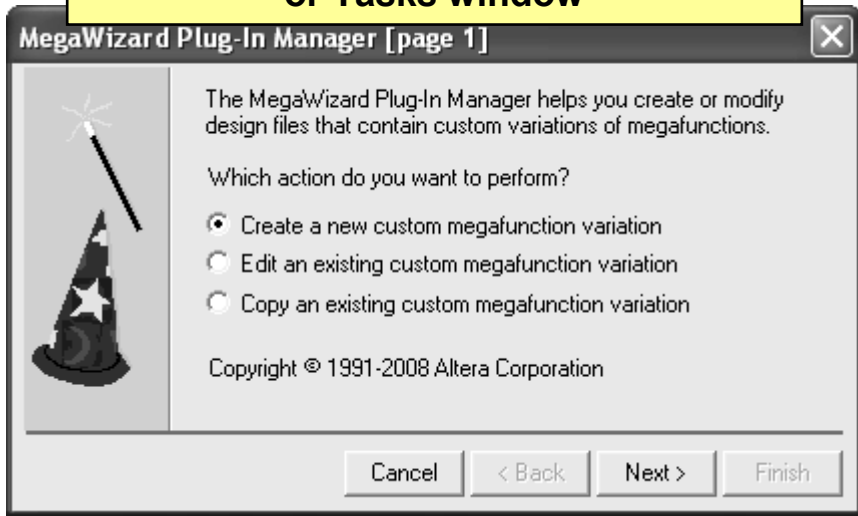
- Triple-Speed Ethernet MAC
- FIR Compiler
- Fast Fourier Transform
- DDR/DDR2 High Performance Memory Controller
- CRC Compiler
- PCI Compiler

MegaWizard Plug-in Manager

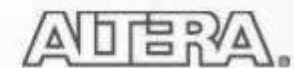


- Eases implementation and configuration of megafunctions & IP
- GUI, command line or both

Tools ⇒ MegaWizard Plug-In Manager or Tasks window



Command line: `qmegawiz <-silent> <module | wizard>=<mf_name> <ports & parameters options> file_name`



MegaWizard Example

Locate documentation in Quartus II Help or the web

ALTMULT_ADD

1 Parameter Settings | 2 EDA | 3 Summary

General | Extra Modes | Saturation | Rounding | Multipliers

Currently selected device family: Cyclone II

Match project/default

General

What is the number of multipliers? 2 multipliers

All multipliers have similar configurations

How wide should the A input buses be? 16 bits

How wide should the B input buses be? 16 bits

How wide should the 'result' output bus be? 33 bits

Create a 4th asynchronous clear input option
This forces all registers to have an associated asynchronous clear input

Create an associated clock enable for each clock

Input Representation

What is the representation format for A inputs? Unsigned

What is the representation format for B inputs? Unsigned

Resource Usage: 4 dsp_9bit + 33 lut + 33 reg

Buttons: Cancel, < Back, Next >, Finish

Multiply-Add megafunction

Three step process to configure megafunction

Updating graphical representation

Resource usage

Customization options



MegaWizard Output File Selection

MegaWizard Plug-In Manager [page 9 of 9] -- Summary

ALTMULT_ADD

1 Parameter Settings 2 EDA 3 Summary

About Documentation

Turn on the files you wish to generate. A gray checkmark indicates a file that is automatically generated, and a red checkmark indicates an optional file. Click Finish to generate the selected files. The state of each checkbox is maintained in subsequent MegaWizard Plug-In Manager sessions.

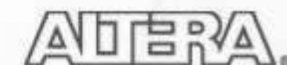
The MegaWizard Plug-In Manager creates the selected files in the following directory:

C:\altera_trn\Quartus_II_Software_Design_Series_Foundation\QIIF7_2\Solutions\Final_projects\Schematic\

File	Description
<input checked="" type="checkbox"/> mult_add.v	Variation file
<input type="checkbox"/> mult_add.inc	AHDL Include file
<input type="checkbox"/> mult_add.cmp	VHDL component declaration file
<input checked="" type="checkbox"/> mult_add.bsf	Quartus II symbol file
<input checked="" type="checkbox"/> mult_add_inst.v	Instantiation template file
<input checked="" type="checkbox"/> mult_add_bb.v	Verilog HDL black-box file
<input checked="" type="checkbox"/> mult_add_waveforms.html	Sample waveforms in summary
...mult_add_wave*.jpg	Sample waveform file(s)









Resource Usage
4 dsp_9bit + 33 lut + 33 reg

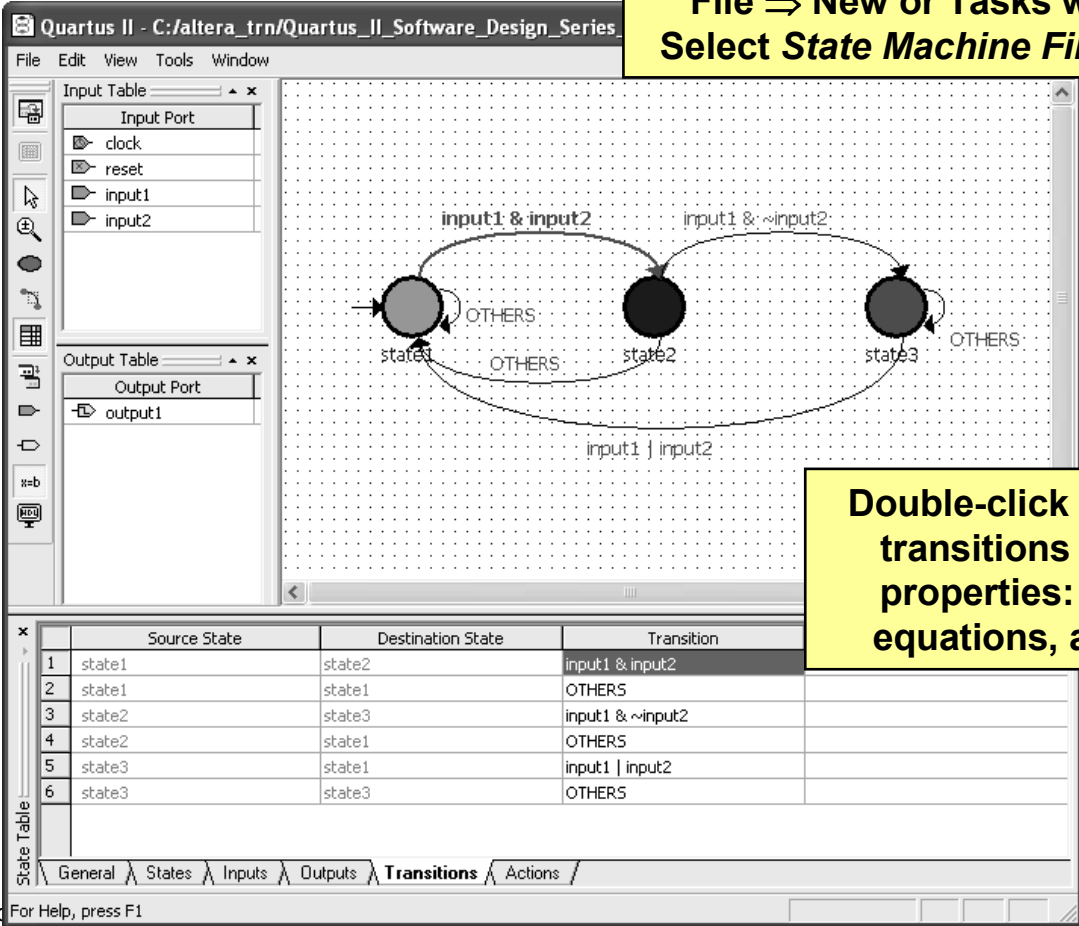
- **Default**
 - HDL wrapper file
- **Selectable**
 - HDL instantiation template
 - VHDL component declaration (CMP)
 - Quartus II symbol (BSF)
 - Verilog black box
 - Behavioral waveform (.html)



State Machine Editor

- Create state machines in GUI
 - Manually by adding individual states and transitions
 - Automatically with State Machine Wizard (**Tools** menu)
- Generate state machine HDL code (required)

- New state 
- New transition 
- State table 
- State machine wizard 
- Input port 
- Output port 
- View equations 
- Generate HDL 



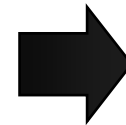
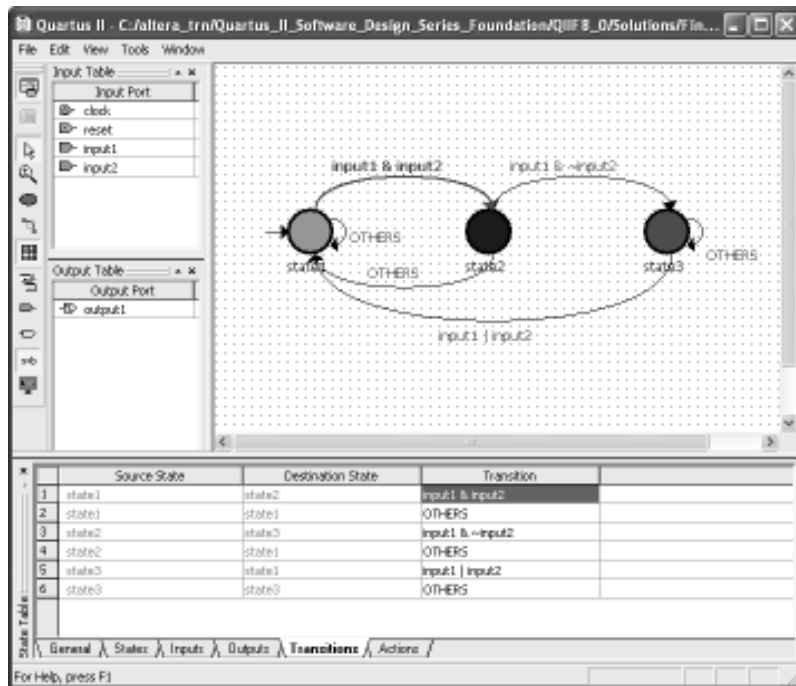
File ⇒ New or Tasks window
Select *State Machine File* (.SMF)

Double-click states & transitions to edit properties: name, equations, actions

	Source State	Destination State	Transition
1	state1	state2	input1 & input2
2	state1	state1	OTHERS
3	state2	state3	input1 & ~input2
4	state2	state1	OTHERS
5	state3	state1	input1 input2
6	state3	state3	OTHERS

From .SMF to HDL

- Generate optimized code (Verilog or VHDL)
- Add to project
- Required for use



```

22 module SM1 (
23     clock, reset, input1, input2,
24     output1);
25
26     input clock;
27     input reset;
28     input input1;
29     input input2;
30     tri0 reset;
31     tri0 input1;
32     tri0 input2;
33     output output1;
34     reg output1;
35     reg reg_output1;
36     reg [2:0] fstate;
37     reg [2:0] reg_fstate;
38     parameter state1=0, state2=1, state3=2;
39
40     initial
41     begin
42         reg_output1 <= 1'b0;
43     end
44
45     always @(posedge clock)
46     begin
47         if (clock) begin
48             fstate <= reg_fstate;
49             output1 <= reg_output1;
50         end
51     end
52
53     always @(fstate or reset or input1 or input2)
54     begin
55         if (reset) begin
56             reg_fstate <= state1;
57             reg_output1 <= 1'b0;
58         end
59         else begin
60             case (fstate)
61             state1: begin
62                 if ((input1 & input2))
63                     reg_fstate <= state2;
64                 else
65                     reg_fstate <= state1;
66             end
67             state2: begin
68                 reg_output1 <= 1'b1;
69             end
69         end
70     end

```

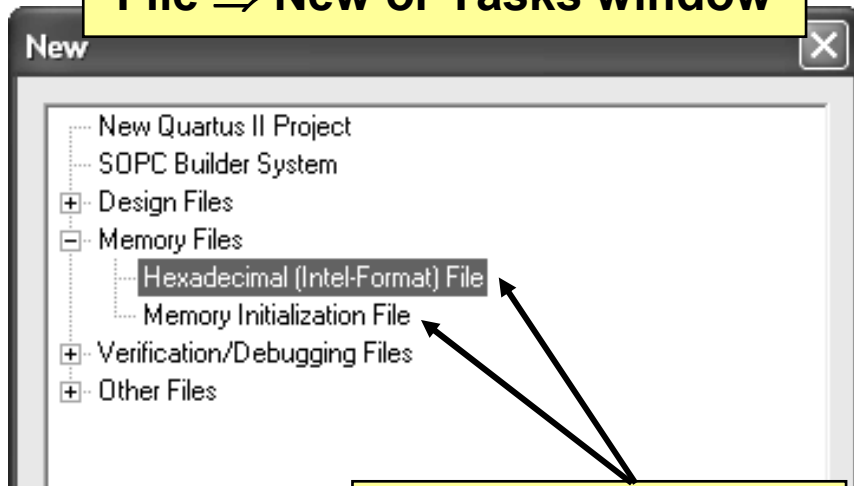


Memory Editor

- Create or edit memory initialization files in Intel HEX (.HEX) or Altera-specific (.MIF) format
- Design entry
 - Use to initialize memory blocks (ex. RAM, ROM) during power-up
 - Initialization file data sent to device during device programming
- Simulation
 - Use to initialize memory blocks before simulation or after breakpoints

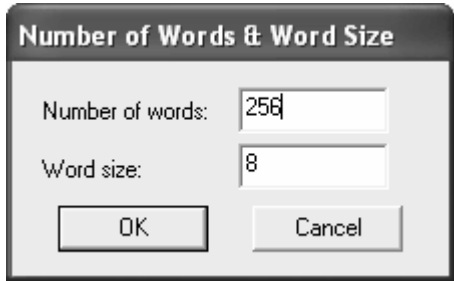
Create Memory Initialization File

File ⇒ New or Tasks window

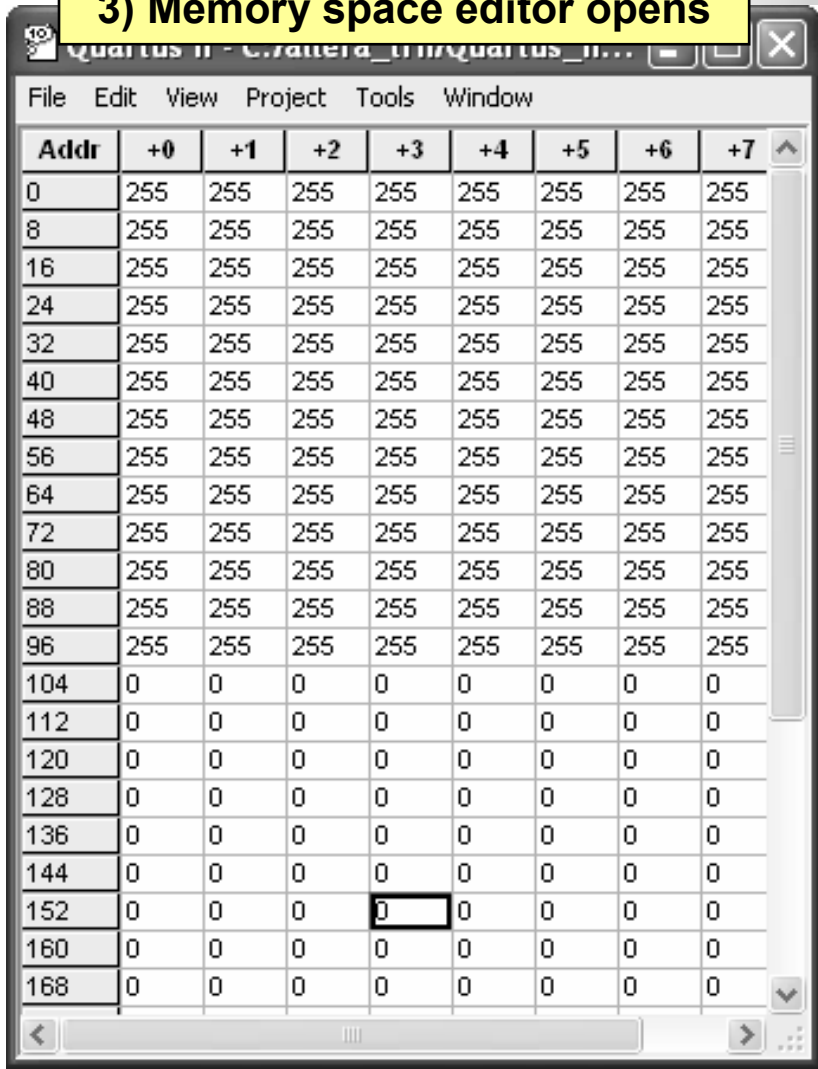


1) HEX or MIF format

2) Select memory size



3) Memory space editor opens



Using Memory File In Design

MegaWizard Plug-In Manager - RAM: 2-PORT [page 8 of 10]

RAM: 2-PORT

1 Parameter Settings | 2 EDA | 3 Summary

General > Widths/Blk Type > Clks/Rd, Byte En > Regs/Clkens/Aclrs > Output1 > Mem Init >

new_ram

data[7..0]
wraddress[4..0]
wren
rdaddress[4..0]
clock

32 Word(s) RAM

q[7..0]

Block Type: AUTO

Resource Usage
256 ram_bits

Do you want to specify the initial content of the memory?

No, leave it blank

Initialize memory content data to XX..X on power-up in simulation

Yes, use this file for the memory content data
(You can use a Hexadecimal (Intel-format) File [.hex] or a Memory Initialization File [.mif])

Browse...

File name: pipemult.hex

The initial content file should conform to which port's dimensions? PORT_B

Cancel < Back Next > Finish

Specify MIF or HEX file in MegaWizard

May also specify MIF or HEX file in HDL using the ram_init_file attribute



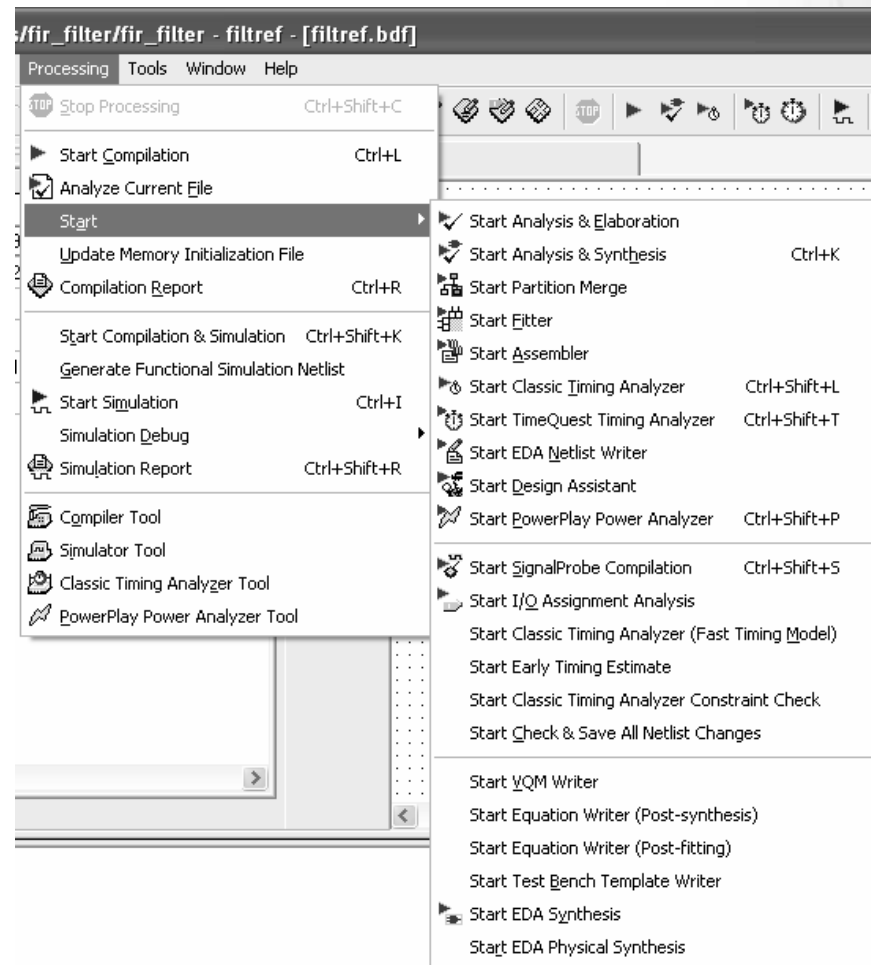
Quartus II Software Design Series: Foundation

Quartus II Compilation



Processing Options

- Start Compilation
 - Performs full compilation
- Start Analysis & Elaboration
 - Checks syntax & builds database only
 - Performs initial synthesis
- Start Analysis & Synthesis
 - Synthesizes & optimizes code
- Start Fitter
 - Places & routes design
 - Generates output netlists
- Start Assembler
 - Generate programming files
- Start TimeQuest Timing Analyzer
- Start I/O Assignment Analysis
- Start Design Assistant



Status & Message Windows



View menu ⇒ Utility Windows

Module	Progress %	Time ⌚
Full Compilation	49 %	00:00:20
Analysis & Synthesis	100 %	00:00:09
Fitter	96 %	00:00:11
Assembler	0 %	00:00:00
TimeQuest Timing Analyzer	0 %	00:00:00

Status bars indicate compilation progress

Task %	Task	Time ⌚
49%	Compile Design	00:00:57
✓	Analysis & Synthesis	00:00:35
96%	Fitter (Place & Route)	00:00:22
0%	Assembler (Generate programming files)	
0%	TimeQuest Timing Analysis	
	EDA Netlist Writer	
	Program Device (Open Programmer)	

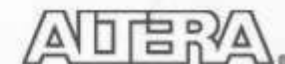
Message window displays informational, warning, & error messages

Type	Flag	Message
i		Info: Fitter routing operations ending: elapsed time is 00:00:00
i		Info: The Fitter performed an Auto Fit compilation. No optimizations were skipped because the design's timing a
i		Info: Started post-fitting delay annotation
+	⚠	Warning: Found 16 output pins without output pin load capacitance assign
i		Info: Delay annotation completed successfully
i		Info: Design uses memory blocks. Violating setup or hold times or memory
+	⚠	Warning: The Reserve All Unused Pins setting has not been specified, and
i		Info: Generated suppressed messages file C:/altera_trn/Quartus_II_Softwa
+	i	Info: Quartus II Fitter was successful. 0 errors, 4 warnings
i		Info: *****

Manually flag selected messages for later review

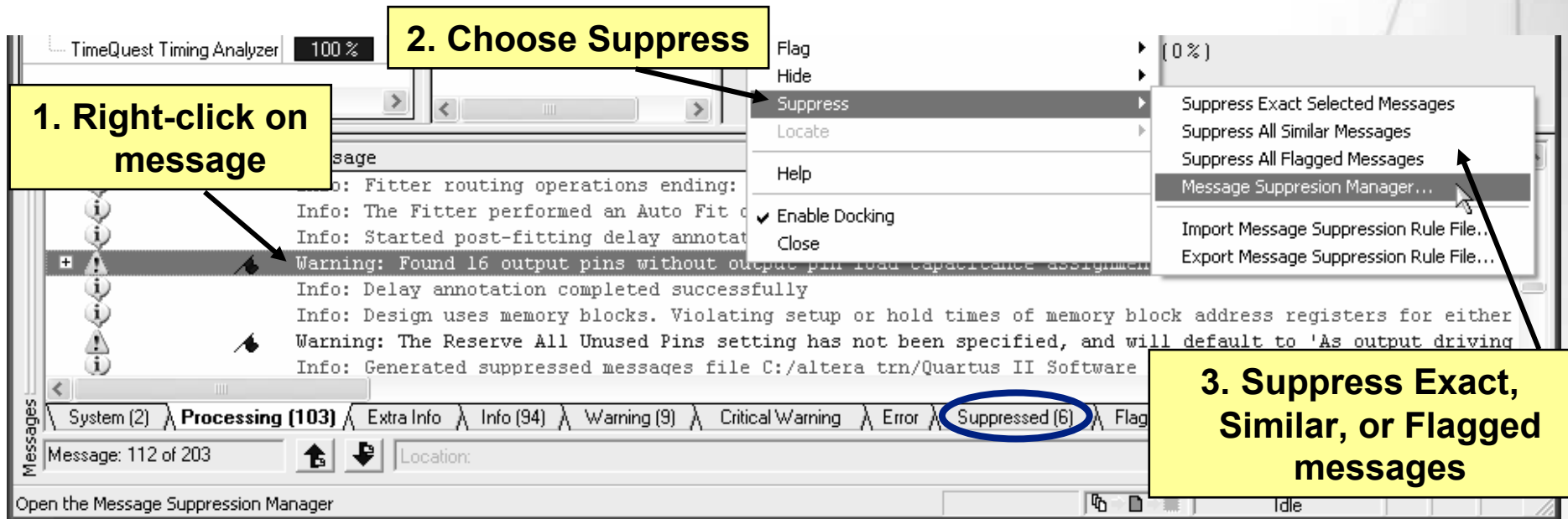
System (2) Processing (103) Extra Info Info (94) Warning (9) Critical Warning Error Suppressed (6) Flag (2)

Message: 109 of 203 Location: Locate

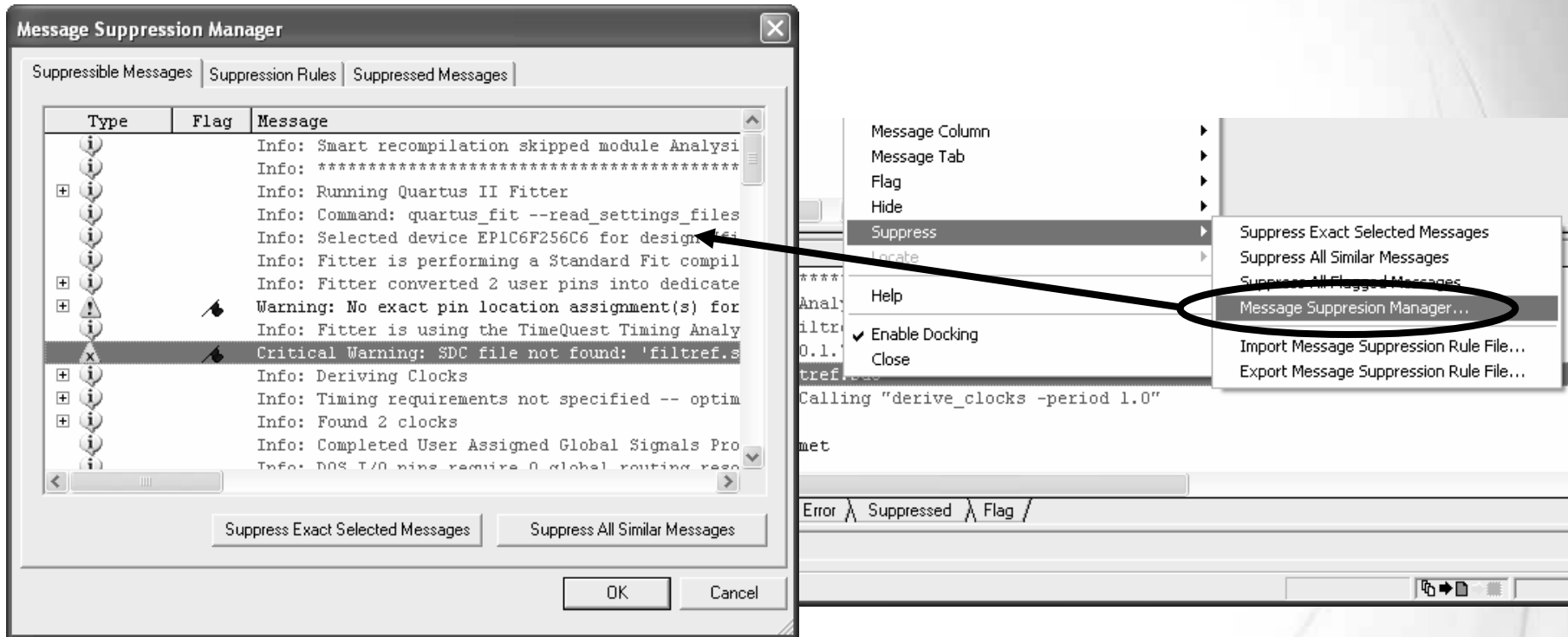


Message Suppression

- Hides messages from current & future compiles
 - Ex. Known synthesis warning message already investigated
- Displays suppressed messages on different tab in message window
- Stores suppression rules in <revision_name>.SRF file



Message Suppression Manager Tool



■ Use to

- View all suppressible messages
- View/add/remove suppression rules
- View messages suppressed for current & future compiles

Compilation Report

- Graphical window containing all compilation processing information
 - Resource usage
 - Device pin-out
 - Settings and constraints applied
 - Messages
- Opens automatically when processing begins
- Recommendation: Go through report for a design to get sense of information being provided
- Information also available as text files in project directory
 - Ex. `<project_name>.fit.rpt` & `<project_name>.map.rpt`

Compilation Report



Flow Summary	
Flow Status	Successful - Thu May 03 17:30:51 2007
Quartus II Version	7.1 Build 156 04/30/2007 SJ Full Version
Revision Name	filtref
Top-level Entity Name	filtref
Family	Cyclone
Device	EP1C6F256C6
Timing Models	Final
Met timing requirements	N/A
Total logic elements	162 / 5,980 (3 %)
Total pins	22 / 185 (12 %)
Total virtual pins	0
Total memory bits	0 / 92,160 (0 %)
Total PLLs	0 / 2 (0 %)

Example: Resource Usage

Quartus II - D:/altera/71/qdesigns/fir_filter/fir_filter - filtref - [Compilation Report - Fitter Resou...]

File Edit View Tools Window

Compilation Report

- Legal Notice
- Flow Summary
- Flow Settings
- Flow Non-Default Global Settings
- Flow Elapsed Time
- Flow Log
- Analysis & Synthesis
- Fitter
 - Summary
 - Settings
 - Netlist Optimizations
 - Pin-Out File
 - Resource Section
 - Resource Usage Summary
 - Input Pins
 - Output Pins
 - I/O Bank Usage
 - All Package Pins
 - Output Pin Default Load For Reported
 - Resource Utilization by Entity
 - Delay Chain Summary
 - Pad To Core Delay Chain Fanout
 - Control Signals
 - Global & Other Fast Signals
 - Non-Global High Fan-Out Signals
 - Logic and Routing Section
 - Device Options
- Advanced Fitter Data
- Messages
- Suppressed Messages
- Assembler
- TimeQuest Timing Analyzer

Several tables in Resource Section detail how much of FPGA resources available and used

Fitter Resource Usage Summary		Usage
Resource		
1	Total logic elements	162 / 5,980 (3 %)
2	-- Combinational with no register	77
3	-- Register only	57
4	-- Combinational with a register	28
5		
6	Logic element usage by number of inputs	
7	-- 4 input functions	
8	-- 3 input functions	
9	-- 2 input functions	
10	-- 1 input functions	
11	-- 0 input functions	43
12		
13	Logic elements by mode	
14	-- normal mode	135
15	-- arithmetic mode	27
16	-- qfbk mode	9
17	-- register cascade mode	0
18	-- synchronous clear/load mode	52
19	-- asynchronous clear/load mode	39
20		
21	Total registers	85 / 6,523 (1 %)
22	Total LABs	26 / 598 (4 %)
23	Logic elements in carry chains	30
24	User inserted logic elements	0
25	Virtual pins	0
26	I/O pins	22 / 185 (12 %)
27	-- Clock pins	2 / 2 (100 %)
28	Global signals	3

Netlist Viewers

■ RTL Viewer

- Graphically represents results of synthesis
- Visually check initial HDL synthesis results
 - Before any Quartus II optimizations
- Locate synthesized nodes for assigning constraints
- Debug verification issues

■ Technology Map Viewers (Post-Mapping & regular)

- Graphically represents results of mapping (post-synthesis) & fitting
- Analyze critical timing paths graphically
 - Delay values displayed if timing information available
- Locate nodes & node names after optimizations (cross-probing)
 - Assigning constraints
 - Debugging

RTL Viewer

Tools Menu ⇒ Netlist Viewers or Tasks window “Compile Design” tasks

The screenshot shows the Quartus II software interface. On the left is the Hierarchy List, which is a tree view of the design components. It includes a 'Hierarchy List' section with a 'filtref' folder containing 'Instances' (acc:inst3, hvalues:inst2, mult:inst6, state_m:inst1, taps:inst), 'Primitives' (Pins, Nets), and 'Logic Clouds' (sel[1..0]_DATAIN_). On the right is the Schematic View, which displays a complex circuit diagram with various logic blocks and interconnecting lines. A yellow box labeled 'Schematic View' is overlaid on the bottom right of the schematic area.

Schematic View

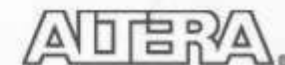
Hierarchy List

Note:

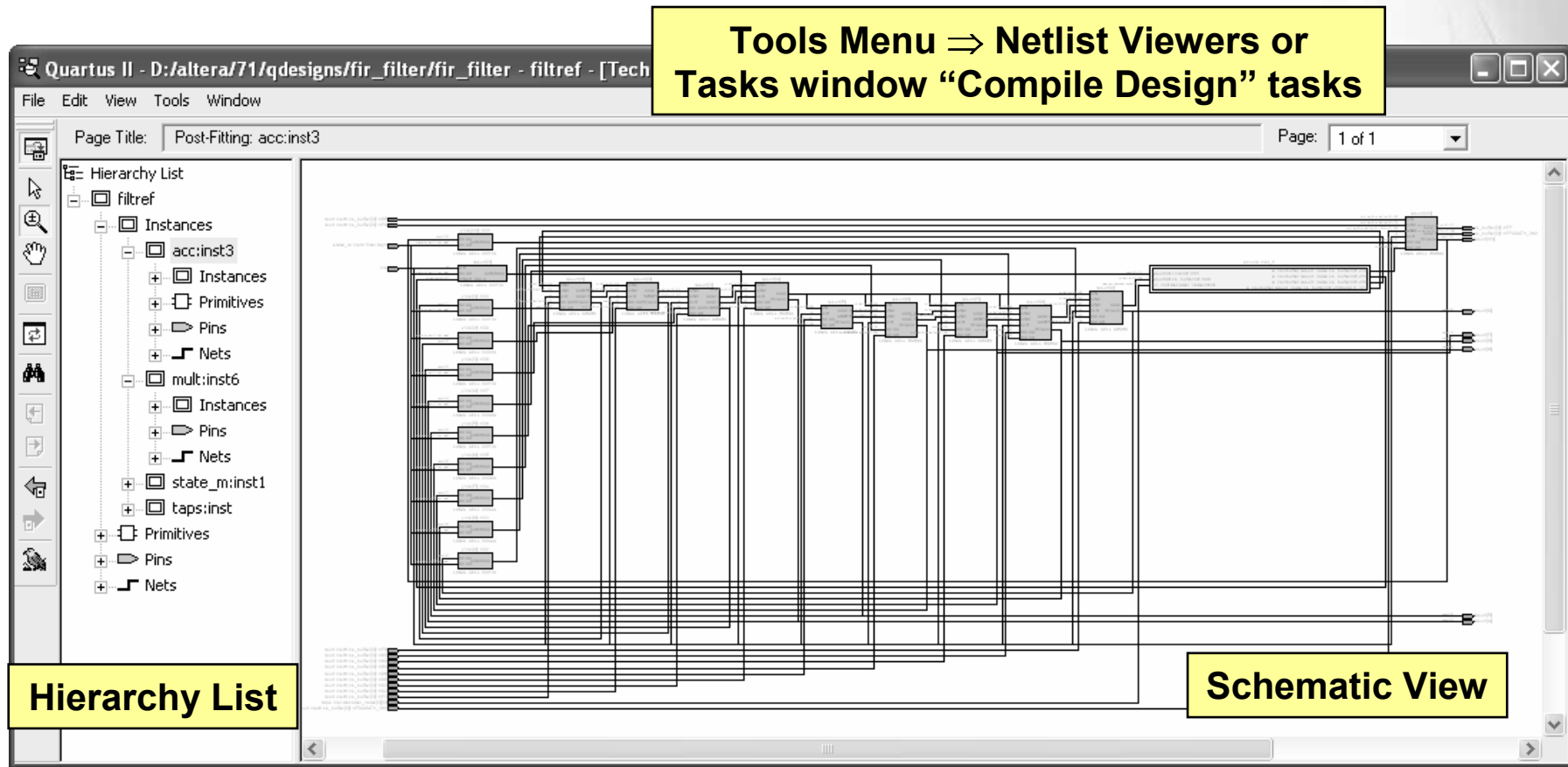
1) Must perform elaboration first (e.g. Analysis & Elaboration OR Analysis & Synthesis)

© 2008 Altera Corporation—Confidential

Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation



Technology Map Viewers



Note:

1) **Must run synthesis and/or fitting first**

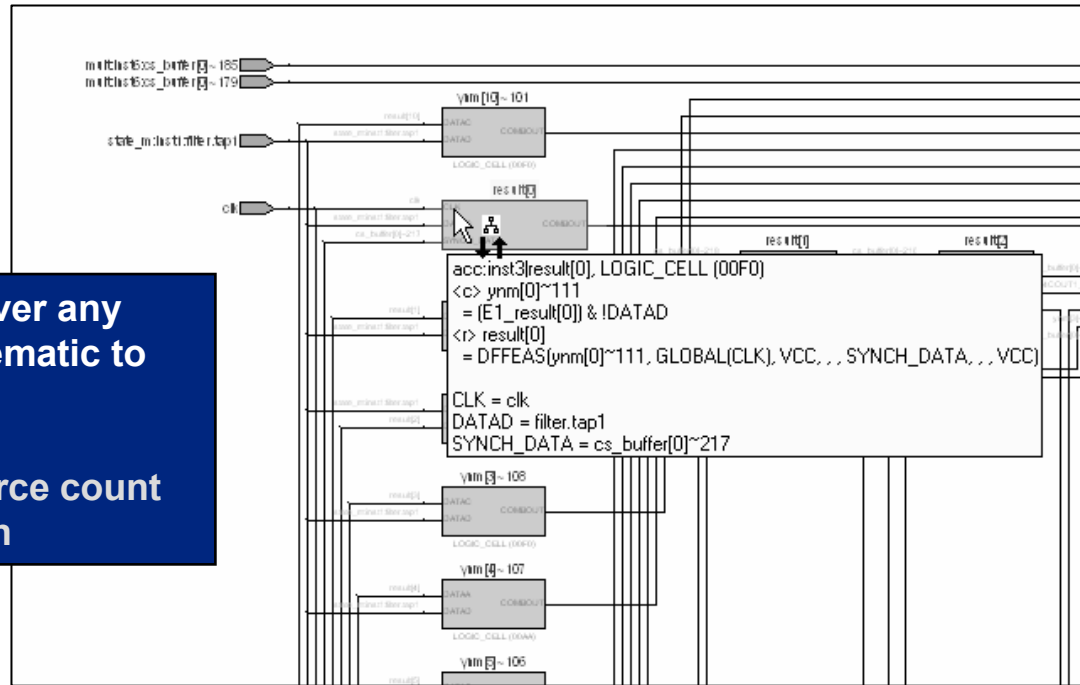
© 2008 Altera Corporation—Confidential

Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation

Schematic View (Technology Viewer)

Place pointer over any element in schematic to see details

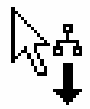
- Name
- Internal resource count
- Logic equation



- Represents design using atoms
 - I/O pins & cells
 - Lcells
 - Memory blocks
 - MAC (DSP blocks)

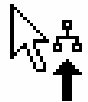
Schematic Hierarchy Navigation

- Mouse pointer indicates action



- ← Descending hierarchy

- Double-click on instance
- Right-click & select **hierarchy down**

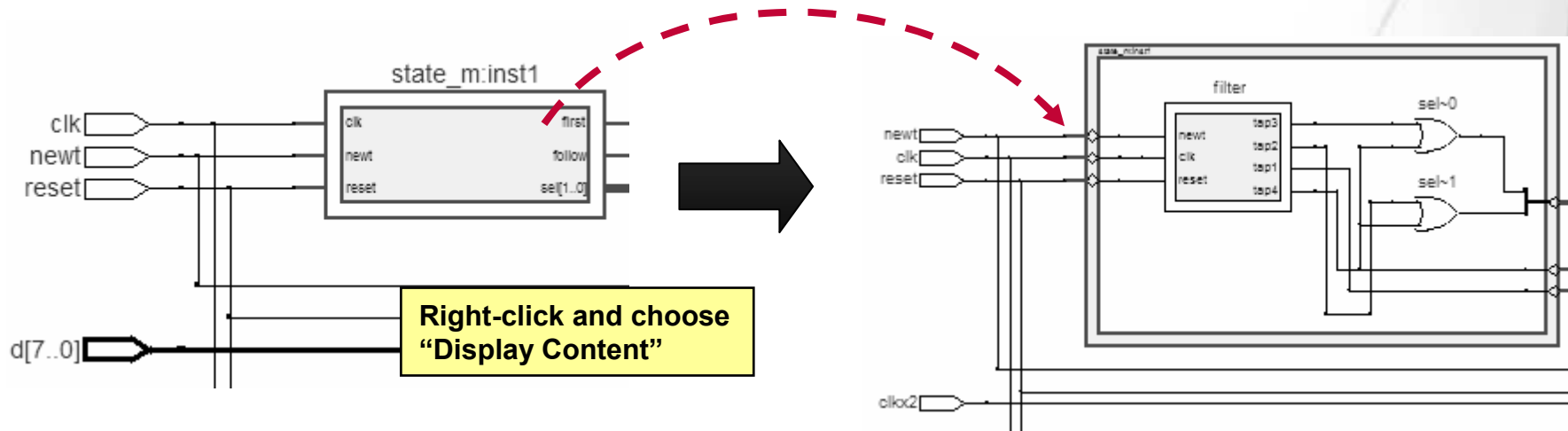


- ← Ascending hierarchy

- Double-click in empty space
- Right-click & select **hierarchy up**

- Expand instances within current level of hierarchy

- Tip: click instances (blocks) to highlight in red first before performing actions



Filter Schematic

Unfiltered: All components & paths shown

Right-click for filter menu

Filtered: Only selected components & related paths displayed

Filter options

- Find... (Ctrl+F)
- Go To... (Ctrl+G)
- Refresh
- Zoom
- Hierarchy Up (Ctrl+U)
- Hierarchy Down (Ctrl+D)
- Filter**
 - Sources
 - Destinations
 - Sources & Destinations
 - Selected Nodes & Nets**
 - Between Selected Nodes
 - Bus Index
- Reduce
- Locate
- Display Content
- Copy Image
- Copy Tooltip
- Group Related Nodes
- Ungroup Selected Nodes
- Group Source Logic
- Ungroup Source Logic
- Viewer Options
- Properties



State Machine Viewer

Tools Menu ⇒ Netlist Viewers or Tasks window “Compile Design” tasks

State Machine: |filter|state_m:inst1|filter

State Flow Diagram

Diagram showing states: idle, tap1, tap4, tap2, tap3. Transitions include: idle to idle (reset), idle to tap1, tap1 to tap4, tap4 to tap1, tap4 to tap2, tap2 to tap3, tap3 to tap2, tap3 to tap4, tap2 to tap1, tap1 to tap3, tap4 to tap3.

State	Condition
1	idle
2	idle
3	tap4
4	tap4
5	tap3
6	tap2
7	tap1

State Transition/Encoding Table

Transitions / Encoding /

Highlighting state in state transition table highlights corresponding state in state flow diagram

Use drop-down to select state machine

Chip Planner

- Editable graphical view of target device
- Displays
 - Graphical layout of device resources
 - Routing channels between device resources
 - Internal routing channels within LABs
 - Global clock regions
- Uses
 - View placement of design logic
 - View connectivity between resources used in design
 - Make placement assignments
 - Debugging placement related issues

Chip Planner



Device: EP1C6F256C6 Task: Floorplan Editing (Assignn...)

Full chip view displays logic placement

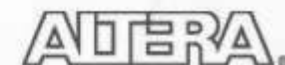
Tools Menu or Tasks window "Compile Design" tasks

Device: EP1C6F256C6 Task: Post-Compilation Editing (...)

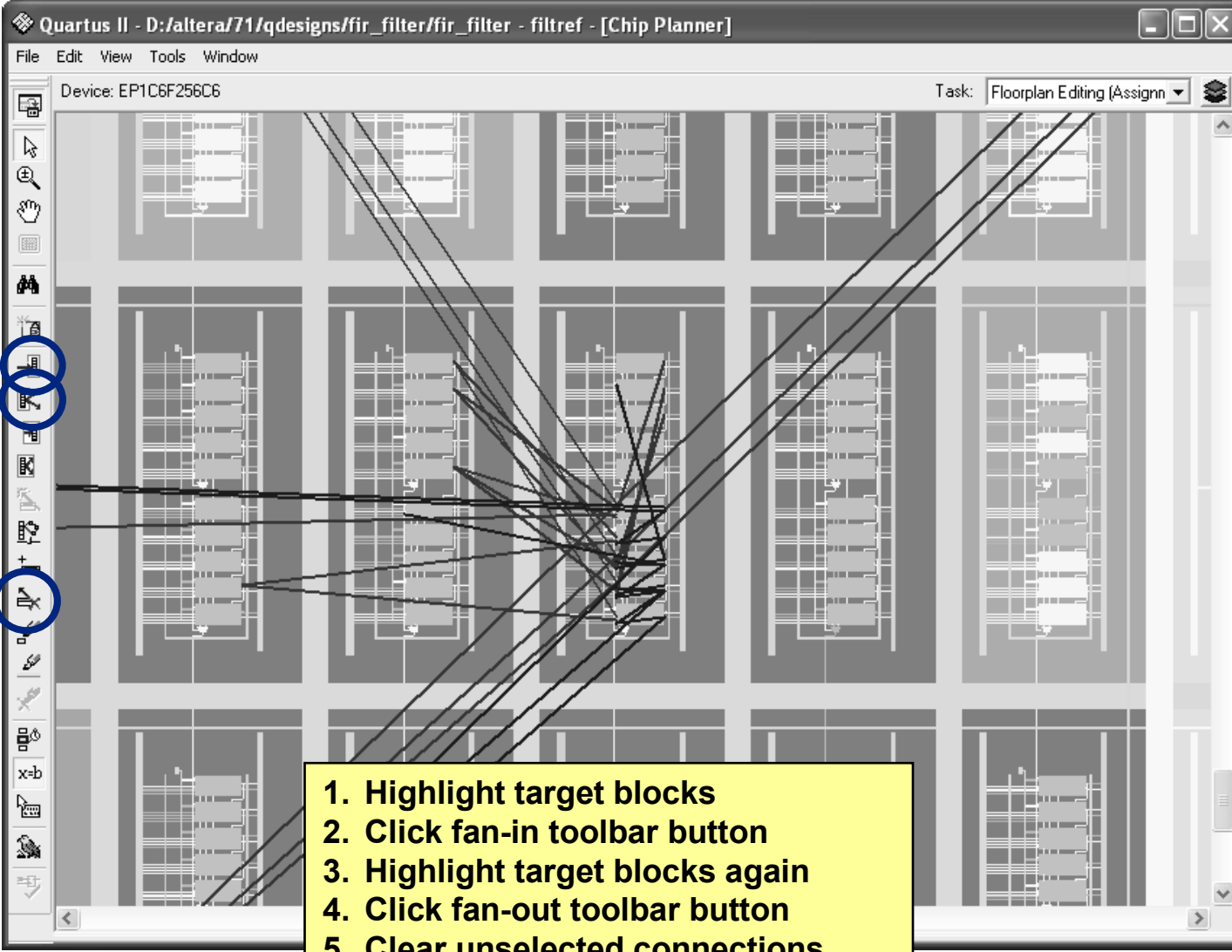
Colored elements indicate used resources

Block utilization: 10 of 10
Resource: LC_X11_Y14_N0
Node [S]: state_m:inst1|filter.tap3

Zoom in for detailed logic implementation & routing usage



Displaying Fan-In & Fan-Out



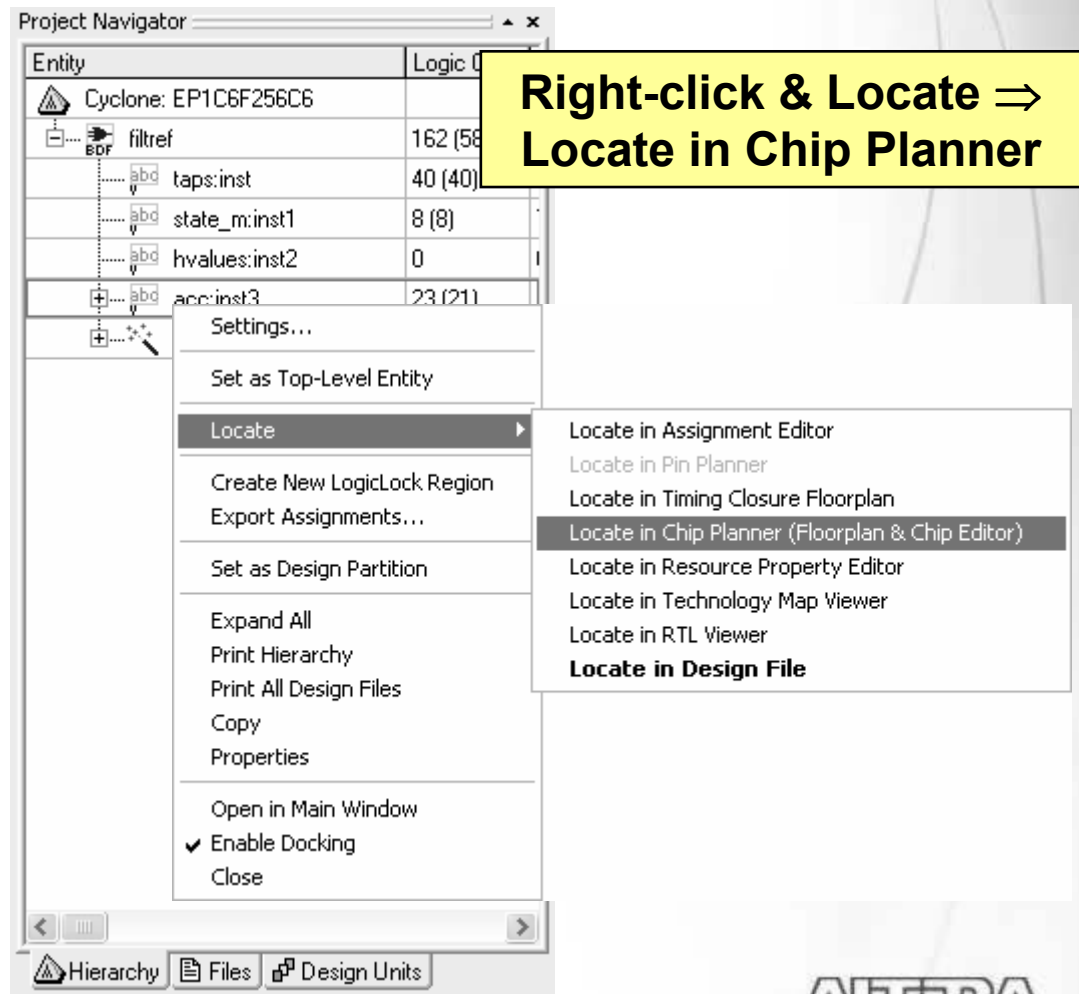
The screenshot shows the Quartus II Floorplan Editor interface. The title bar reads "Quartus II - D:/altera/71/qdesigns/fir_filter/fir_filter - filtref - [Chip Planner]". The menu bar includes "File", "Edit", "View", "Tools", and "Window". The status bar shows "Device: EP1C6F256C6" and "Task: Floorplan Editing (Assignn)". The main workspace displays a grid of logic blocks with a complex network of connections. Two toolbar buttons on the left are circled in blue: the fan-in button (top) and the fan-out button (bottom). A yellow text box at the bottom of the workspace contains the following instructions:

1. Highlight target blocks
2. Click fan-in toolbar button
3. Highlight target blocks again
4. Click fan-out toolbar button
5. Clear unselected connections

Cross-Probing from/to Chip Planner

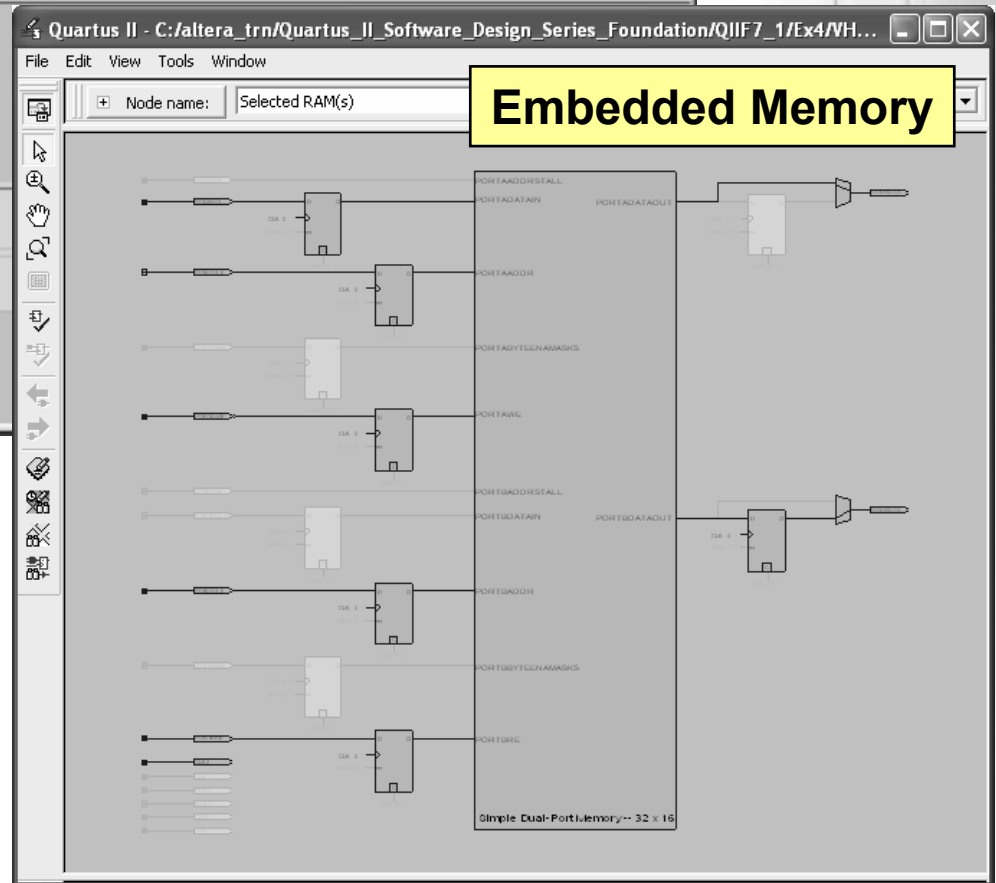
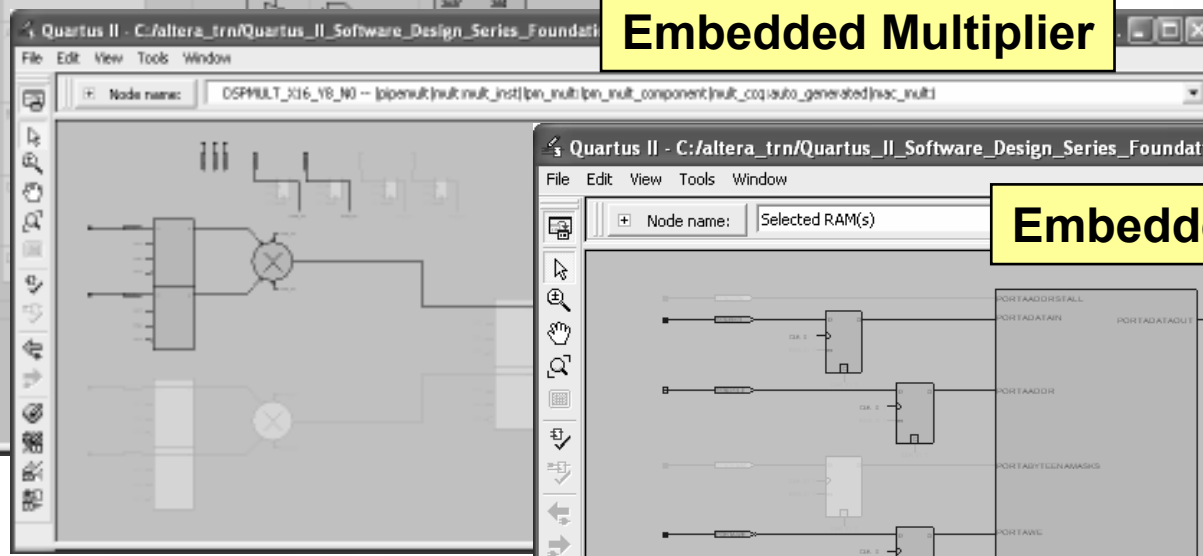
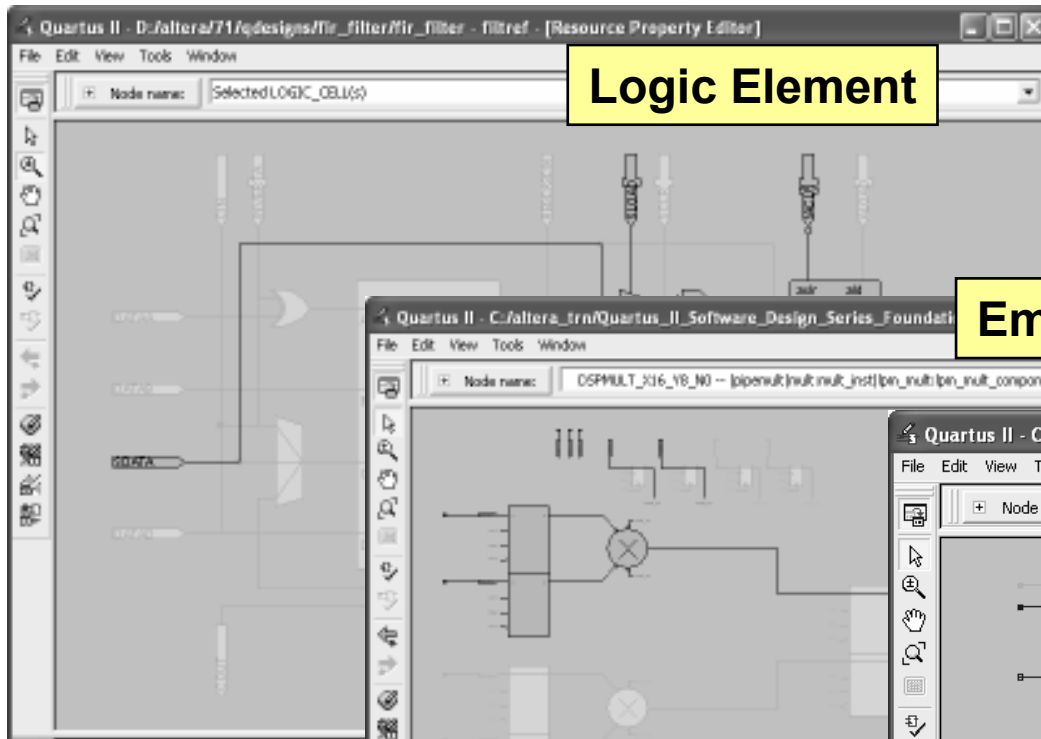
- Locate hierarchy blocks or specific logic from other Quartus II windows

- Project Navigator
- Compilation Report
- Design files
- RTL Viewer
- Technology Viewer
- Message window
- Pin Planner
- TimeQuest reports



Resource Property Editor

- Use to view detailed logic implementation & connections
 - No direct access from menus
 - Cross-probe from other Quartus II tools
- Advanced feature: make ECO changes post-fit without recompiling
- Views
 - Logic cells (look-up tables & registers)
 - Embedded memory
 - Embedded multipliers
 - I/O cells
 - PLLs





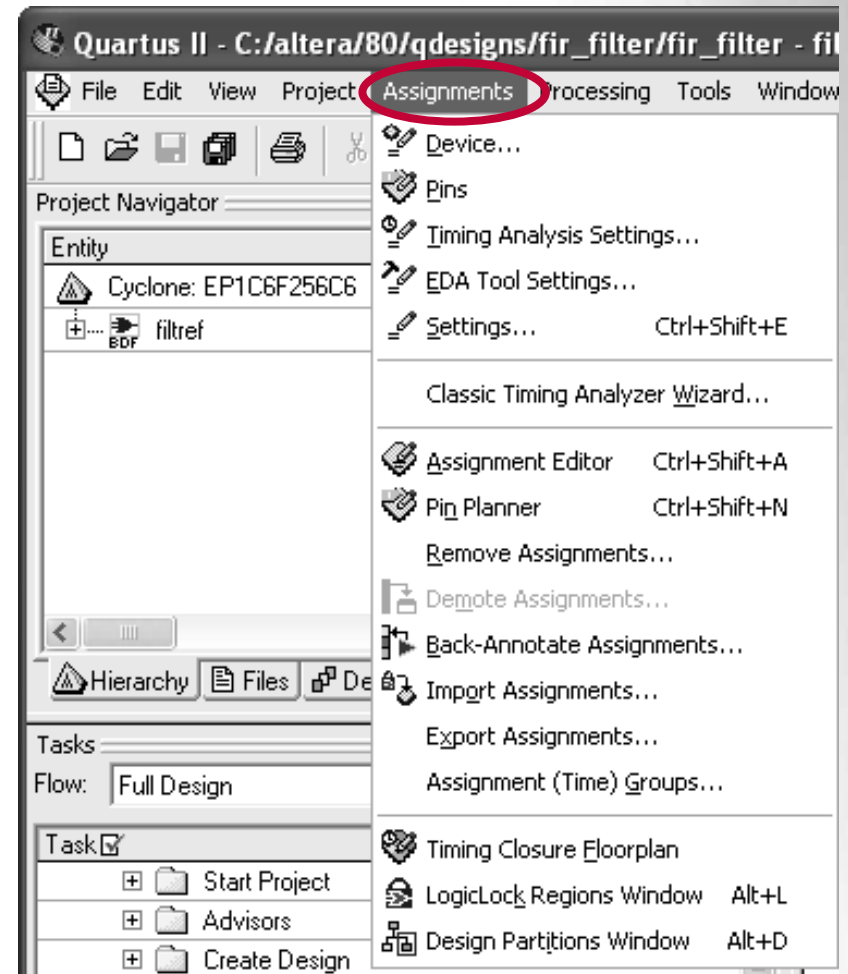
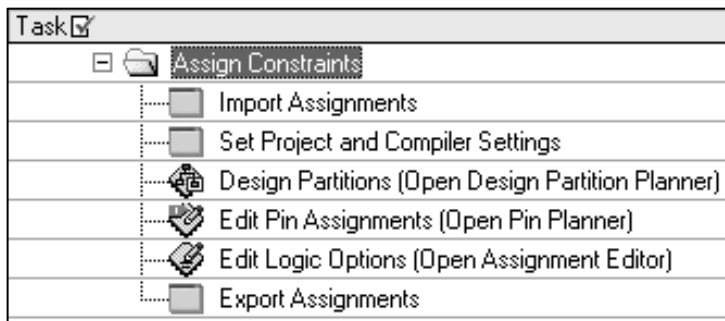
Quartus II Software Design Series: Foundation

Settings & Assignments

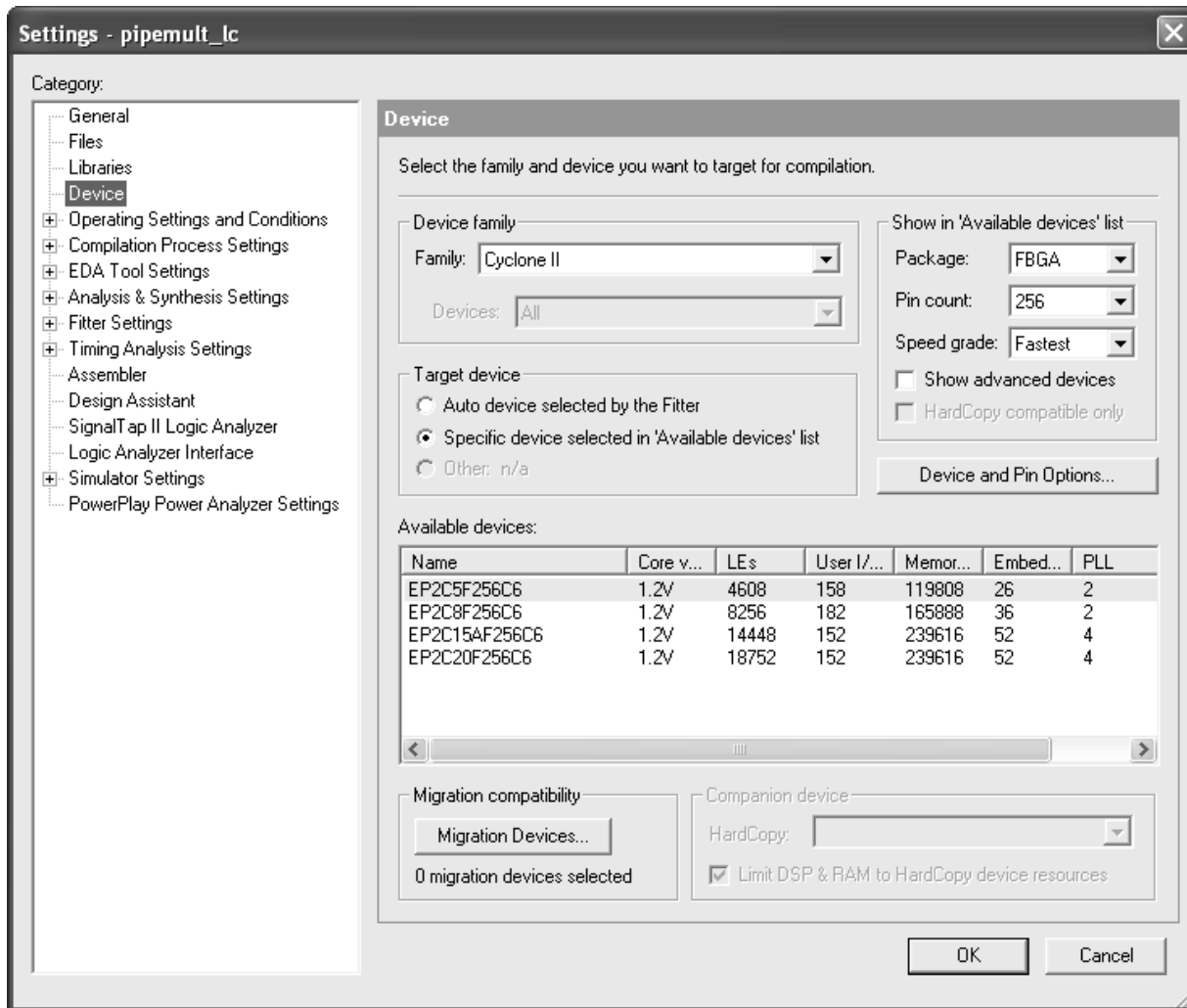


Synthesis & Fitting Control

- Controlled using two methods
 - **Settings**
 - Project-wide switches
 - **Assignments** (aka logic options; constraints)
 - Individual entity/node controls
- Both accessed in **Assignments** menu or Tasks window
- Stored in QSF file for project/revision



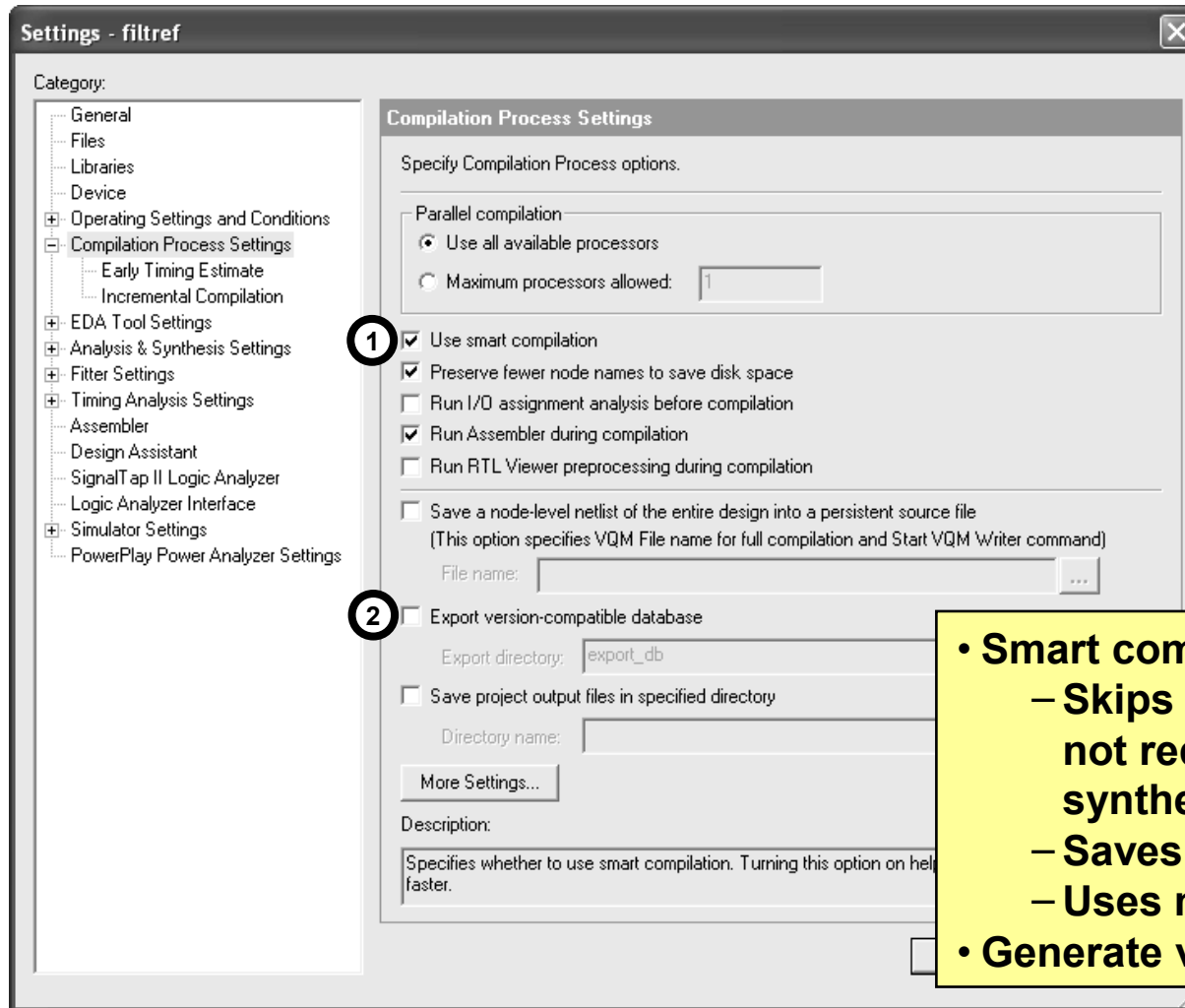
Settings Dialog Box



- Change settings**
- Top-level entity
 - Target device
 - Add/remove files
 - Libraries
 - VHDL '87 or '93?
 - Verilog '95, '01 or SystemVerilog?
 - EDA tool settings
 - Timing settings
 - Compiler settings
 - Synthesis settings
 - Fitter settings
 - Simulator settings
 - Power analysis settings

```
Tcl: set_global_assignment -name <assignment_name*> <value>
```


Compilation Process

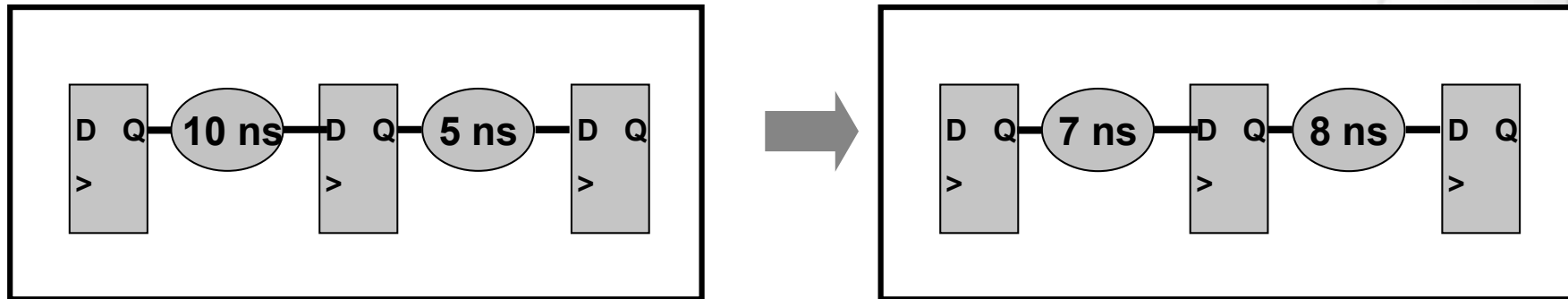
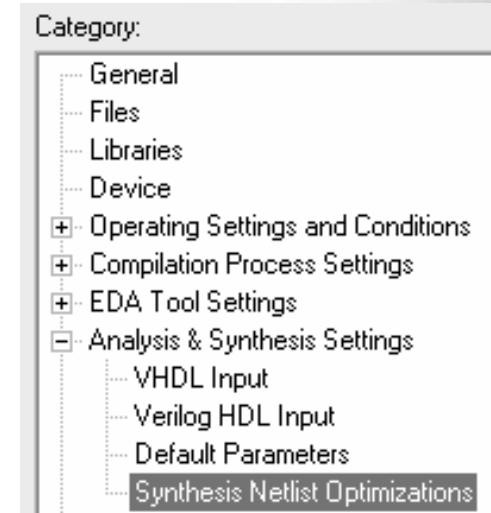


- **Smart compilation⁽¹⁾**
 - Skips entire compiler modules when not required (i.e. elaboration, synthesis, etc.)
 - Saves compiler time
 - Uses more disk space
- **Generate version-compatible database⁽²⁾**

```
Tcl: set_global_assignment -name SMART_RECOMPILE ON
```

Synthesis Netlist Optimizations: Gate-Level Register Retiming

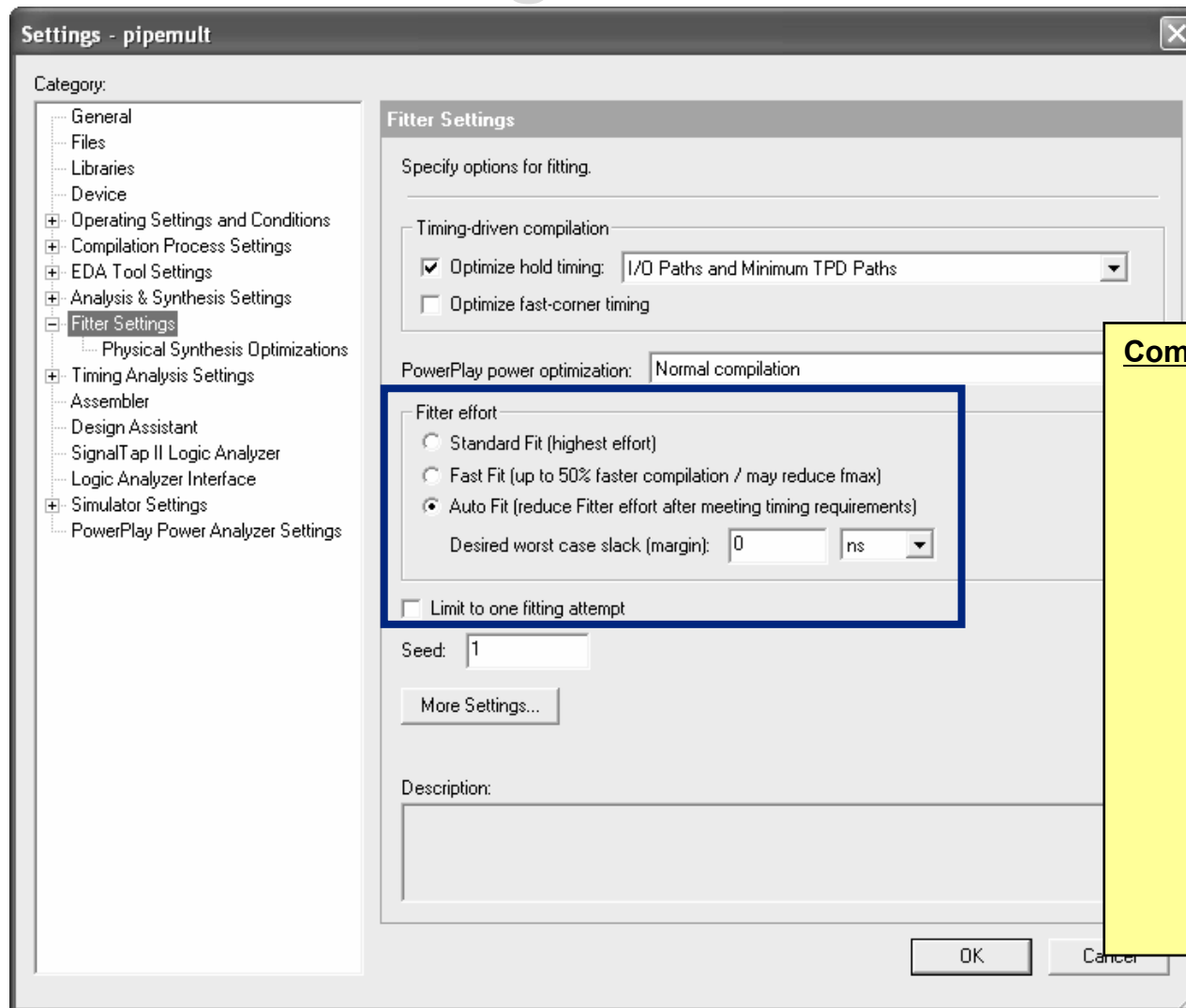
- Optimize netlist during Synthesis
- Moves registers across combinatorial logic to balance timing
- Trades between critical & non-critical paths
- Makes changes at gate level
- Created/modified nodes noted in Compilation Report



```
Tcl: set_global_assignment -name ADV_NETLIST_OPT_SYNTH_GATE_RETIME ON
```



Fitter Settings

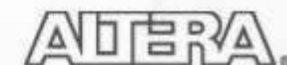


- Compilation speed/fitter effort**
- **Standard fit**
 - Highest effort
 - Longest compile time
 - **Fast fit**
 - Faster compile but possibly lesser design performance
 - **Auto fit**
 - Compile stops after meeting timing
 - Conserves CPU time
 - Will mimic standard fit for hard-to-fit designs
 - Default for new designs
 - **One fitting attempt**

```
Tcl: set_global_assignment -name FITTER_EFFORT "<Effort Level>"
```

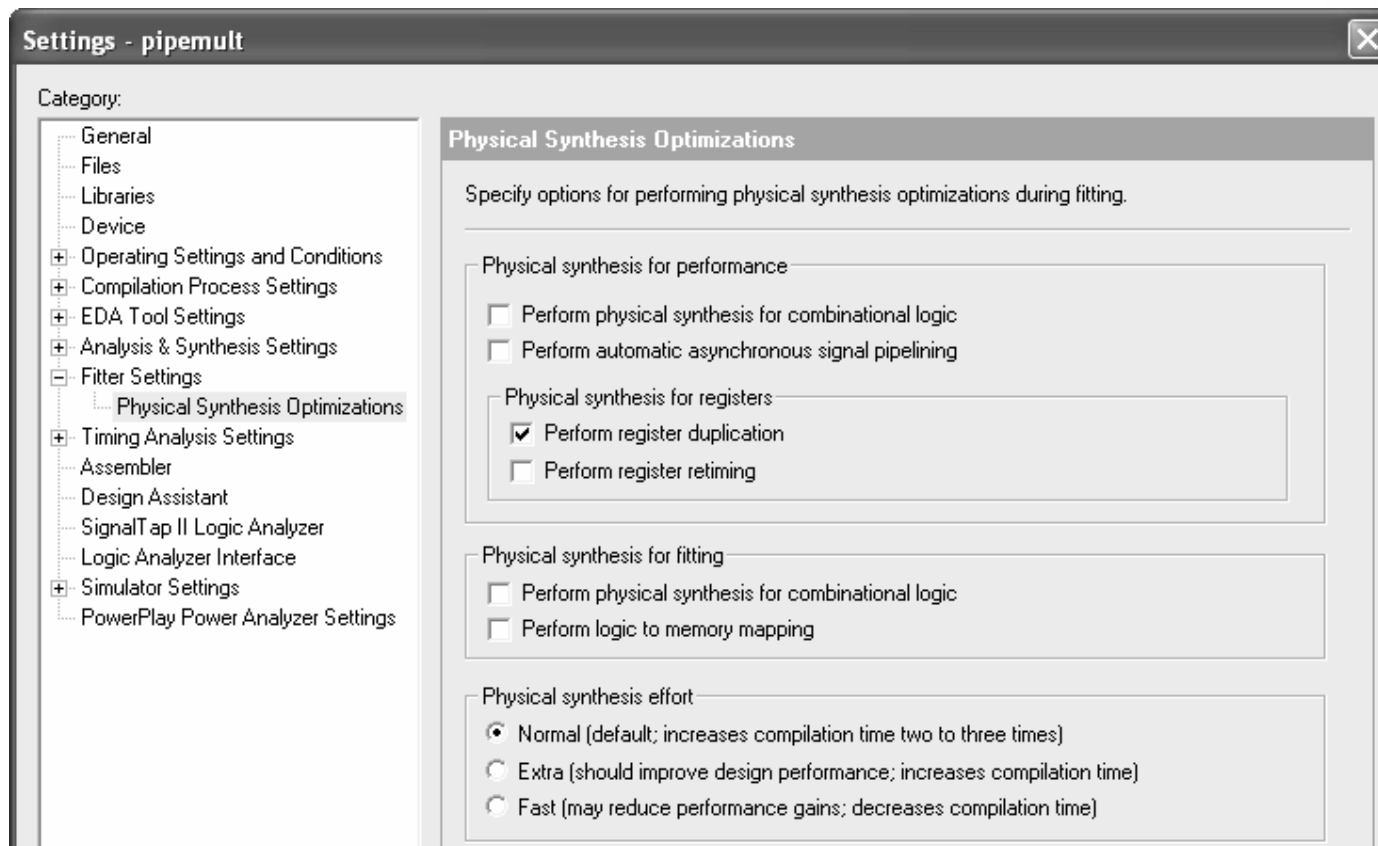
© 2008 Altera Corporation—Confidential

Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation



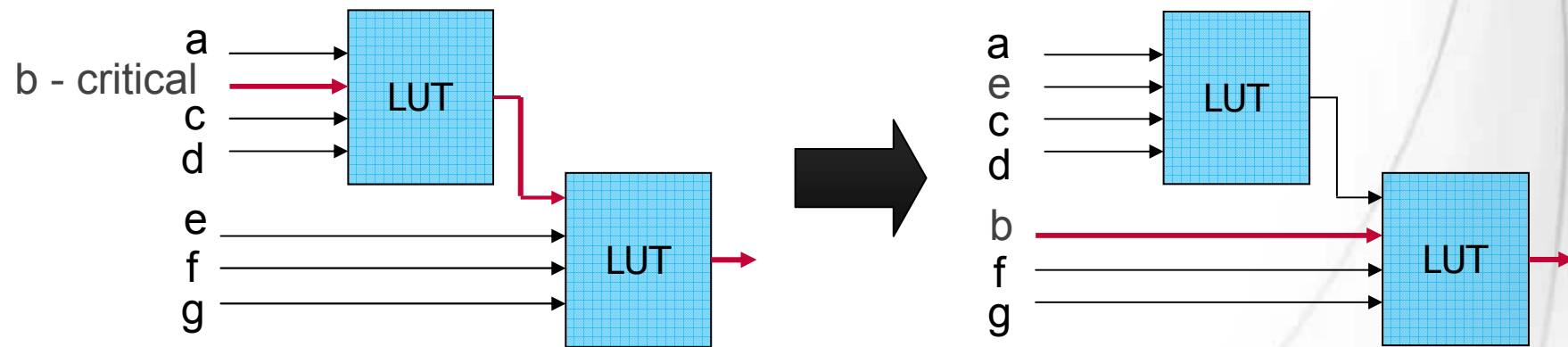
Physical Synthesis

- Re-synthesize based on fitter output
 - Makes incremental changes that improve results for a given placement
 - Compensates for routing delays from Fitter



Combinational Logic

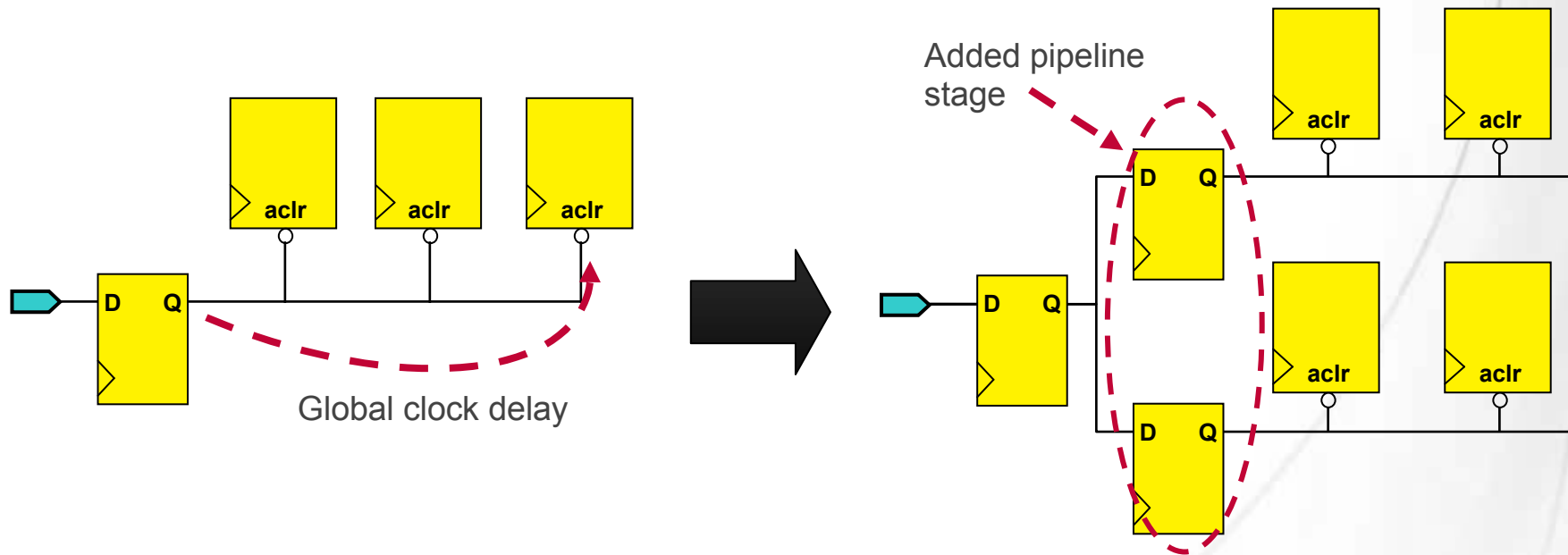
- Swaps look-up table (LUT) ports within LEs to reduce critical path LUTs



```
Tcl: set_global_assignment -name PHYSICAL_SYNTHESIS_COMBO_LOGIC ON
```

Asynchronous Signal Pipelining

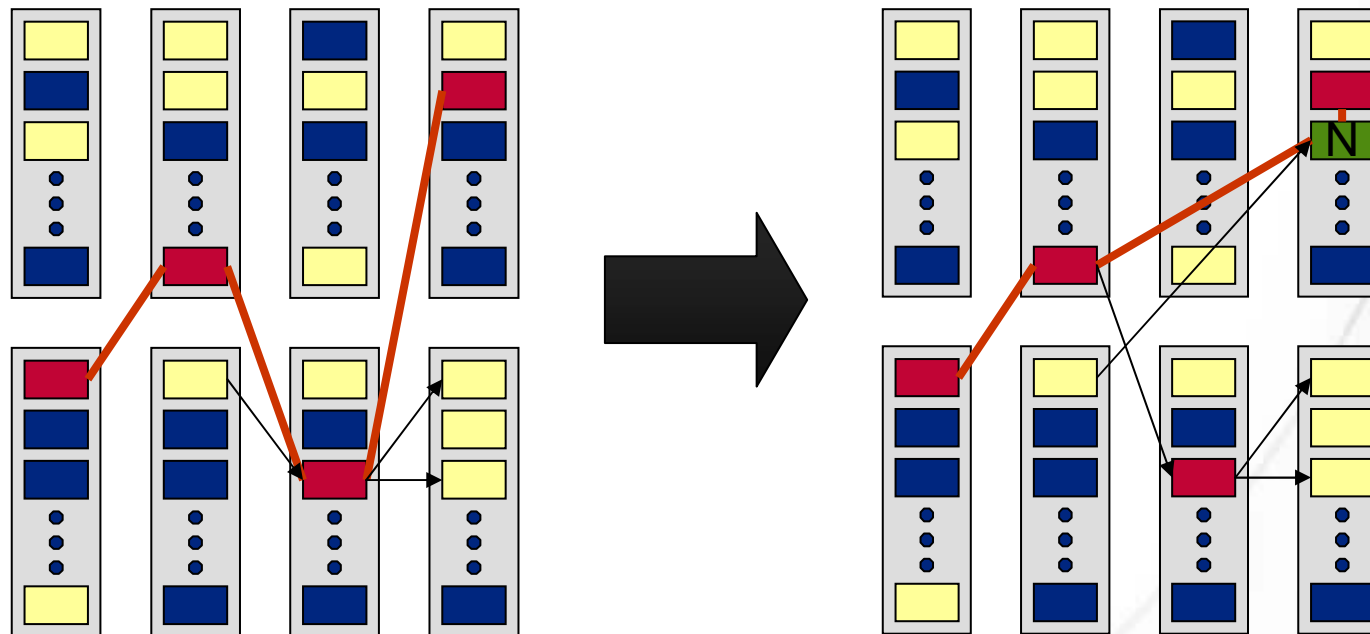
- Adds pipeline registers to asynchronous clear or load signals in very fast clock domains



```
Tcl: set_global_assignment -name PHYSICAL_SYNTHESIS_ASYNCHRONOUS_SIGNAL_PIPELINING ON
```

Duplication

- High fan-out registers or combinatorial logic duplicated & placed to reduce delay



```
Tcl: set_global_assignment -name PHYSICAL_SYNTHESIS_REGISTER_DUPLICATION ON
```

Assignment Editor (AE)

- Provides spreadsheet assignment entry & display
 - Can copy & paste from clipboard

The screenshot shows the Quartus II Assignment Editor window. The window title is "Quartus II - D:/altera/71/qdesigns/fir_filter/fir_filter - filtref - [Assignment Editor]". The menu bar includes File, Edit, View, Tools, and Window. Below the menu bar, there are several control elements: a Category dropdown set to "All", a Node Filter button, and an Information field. The main area is a spreadsheet with columns: From, To, Assignment Name, Value, and Enabled. The spreadsheet contains 16 rows of assignments. A toolbar is visible on the left side of the spreadsheet area. Callouts point to various features: "Sort on columns" points to the toolbar, "Assignment Editor toolbar" points to the left toolbar, "Assignments Menu or Tasks window" points to the top right window controls, "Enable/disable individual assignments" points to the Enabled column, and "Customizable columns" points to the column headers.

	From	To	Assignment Name	Value	Enabled
1		clk	Location	PIN_G1	Yes
2		clk	Clock Settings	clocka	Yes
3		clkx2	Clock Settings	clockb	Yes
4	clk	clkx2	Multicycle	2	Yes
5		d	Location	IOBANK_1	Yes
6		d	I/O Standard	SSTL-2 Class II	Yes
7		reset	I/O Standard	3.3-V LVTTTL	Yes
8		newt	I/O Standard	3.3-V LVCMOS	Yes
9		d[6]	I/O Standard	2.5 V	Yes
10		reset	Location	IOBANK_4	Yes
11		yn_out	Location	IOBANK_4	Yes
12		me	Reserve Pin	As input tri-stated	Yes
13		yvalid	Location	PIN_E14	Yes
14		clkx2	Location	PIN_C13	Yes
15		newt	Location	PIN_C15	Yes
16	<<new>>	<<new>>	<<new>>	<<new>>	<<new>>

Editing Multiple Assignments

- Use Edit bar, auto-fill, copy & paste

Quartus II - D:/altera/71/qdesigns/fir_filter/fir_filter - filtref - [Assignment Editor*]

File Edit View Tools Window

Category: Pin All Timing Logic Options

Node Filter: Click the Node Filter button to view more options

Information: Specifies the I/O standard of a pin. Different device families support different I/O stand with diffe

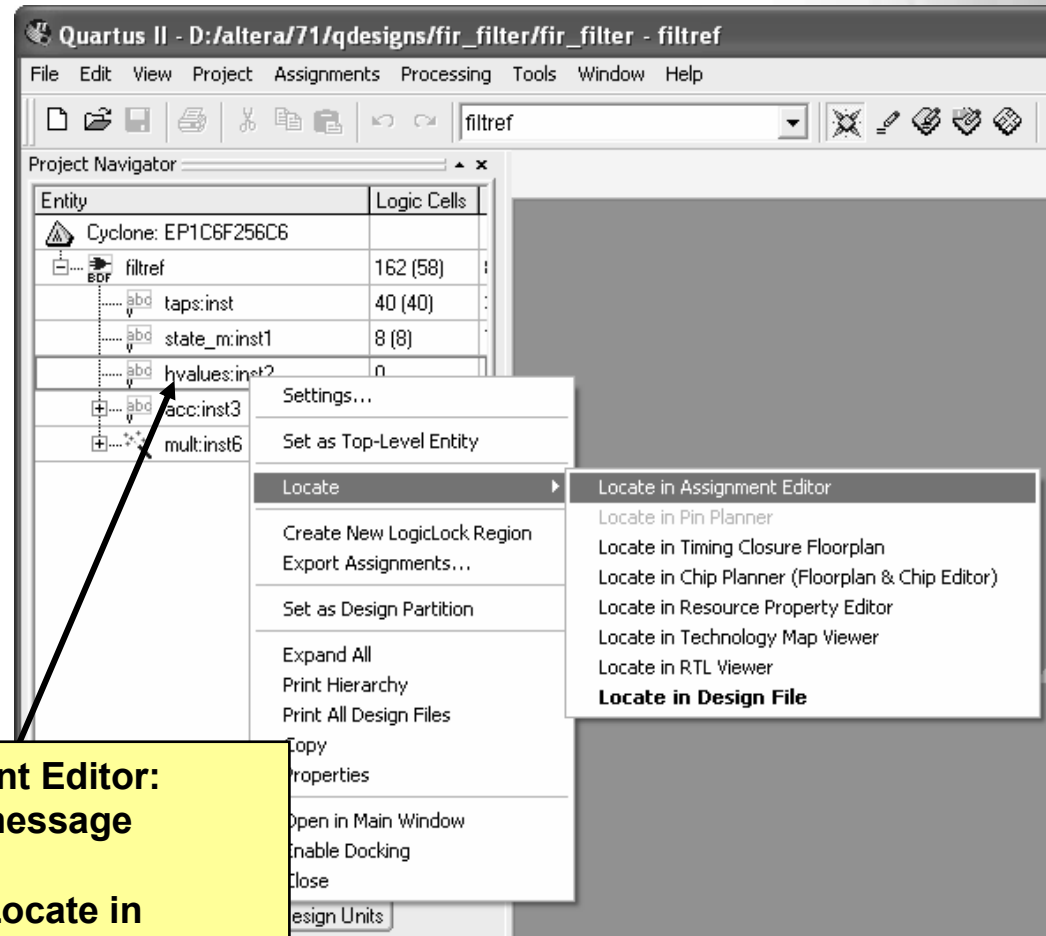
Edit: SSTL-2 Class II

	To	Location	I/O Bank	I/O Standard	General Function	Special Function	Re
1	clk			3.3-V LVTTTL			
2	d	IOBANK_1	1	SSTL-2 Class II			
3	reset	IOBANK_4	4	3.3-V LVTTTL			
4	yn_out	IOBANK_4	4	3.3-V LVTTTL			
5	yvalid	PIN_E14	3	3.3-V LVTTTL	Row I/O	LVD538p/DQ1R3	
6	clkx2	PIN_C13	2	3.3-V LVTTTL	Column I/O	LVD533p	
7	newt	PIN_C15	3	3.3-V LVCMOS	Row I/O	LVD536p	
8	<<new>>	<<new>>					

Auto-fill multiple adjacent cells

Creating Assignments: Cross-Probing

- Virtually all windows & tools cross-probe (locate) to Assignment Editor
- Examples
 - Project Navigator
 - Message window
 - Compilation Report
 - Design files

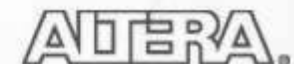


To invoke Assignment Editor:
1) Highlight object/message
2) Right-click
3) Select Locate ⇒ Locate in Assignment Editor*

***Note:** Assignment Editor pre-filled with target node/pin name

© 2008 Altera Corporation—Confidential

Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation



Node Finder



Search by name using wildcards (? or *)

Use filter to select the type of nodes to be displayed

Displays nodes meeting search criteria

Locate nodes in a certain level of hierarchy

List of found nodes in selected entity & lower levels of hierarchy

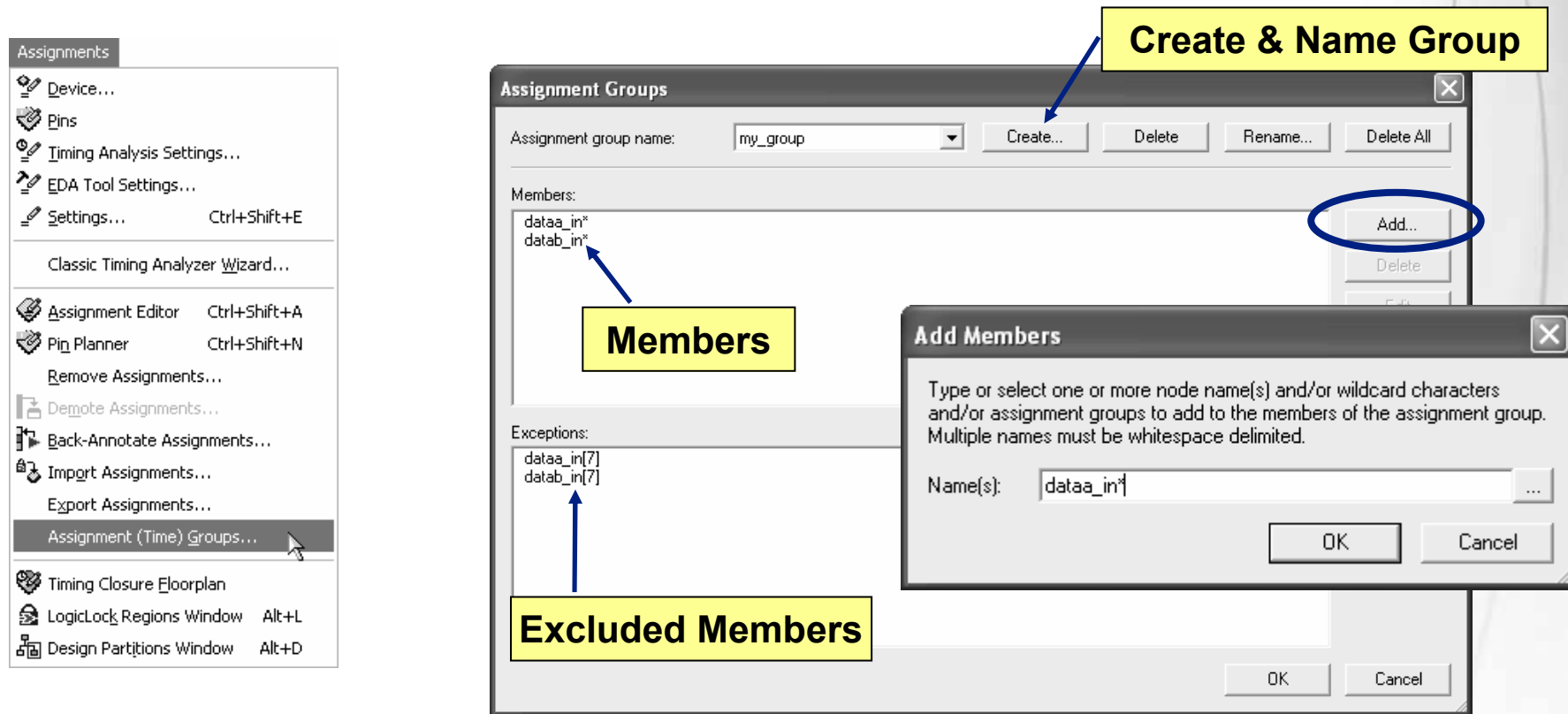
Select nodes on left & use arrows to move to the right

Name	Assignm
clk	PIN_G1
clkx2	PIN_H1
d	Unassign
d[0]	PIN_C7
d[1]	PIN_B7
d[2]	PIN_G2
d[3]	PIN_H5
d[4]	PIN_A8
d[5]	PIN_D7
d[6]	PIN_N7
d[7]	PIN_D8
d[8]	PIN_A6
d[9]	PIN_B8
d[10]	PIN_G16
d[11]	Unassigned
d[12]	PIN_B6
d[13]	PIN_E1
d[14]	PIN_D5
d[15]	PIN_F7



Assignment (Time) Groups

- Assign names to user-defined groups of nodes
- Allows single assignment to constrain entire group




AE Dynamic Checking

- Validity of constraint checked during entry
- Color-coded to display status
 - Grey – disabled
 - Black – applied
 - Yellow – assignment warning
 - Dark red – incomplete
 - Bright red – error/illegal value
 - Green – enter new assignment

	From	To	Assignment Name	Value	Enabled
1		yn_out	Location	IOBANK_2	Yes
2		yvalid	Location	PIN_75	Yes
3		d	Location	IOBANK_1	No
4		clk	Clock Settings	clk	Yes
5		unknown_clock	Clock Settings	clk2	Yes
6		clkx2	Clock Settings	clk2	Yes
7	clk	clkx2	Multicycle	2	Yes
8		d	DQS Frequency	1MHz	Yes
9		yvalid		Minimum Current	Yes
10		d	I/O Standard	LVC MOS	Yes
11		yn_out	I/O Standard	LVC MOS	Yes
12		yvalid	I/O Standard	LVC MOS	Yes
13	<<new>>	<<new>>	<<new>>		

Optimization Technique

- Selects synthesis optimization goal
 - **Speed**
 - **Balanced** (default)
 - **Area**
- Applies only to hierarchical entities
 - Locate (cross-probe) from Project Navigator
 - Enables Assignment Editor Node Filter for selected entity
 - Drag and drop into Assignment Editor
- Effects synthesis & logic mapping
- Only applies to Quartus II integrated synthesis

	From	To	Assignment Name	Value	Enabled	
1		 acc:b2v_inst3	Optimization Technique -- Stratix II	Speed	Yes	
2	<<new>>	<<new>>	<<new>>			

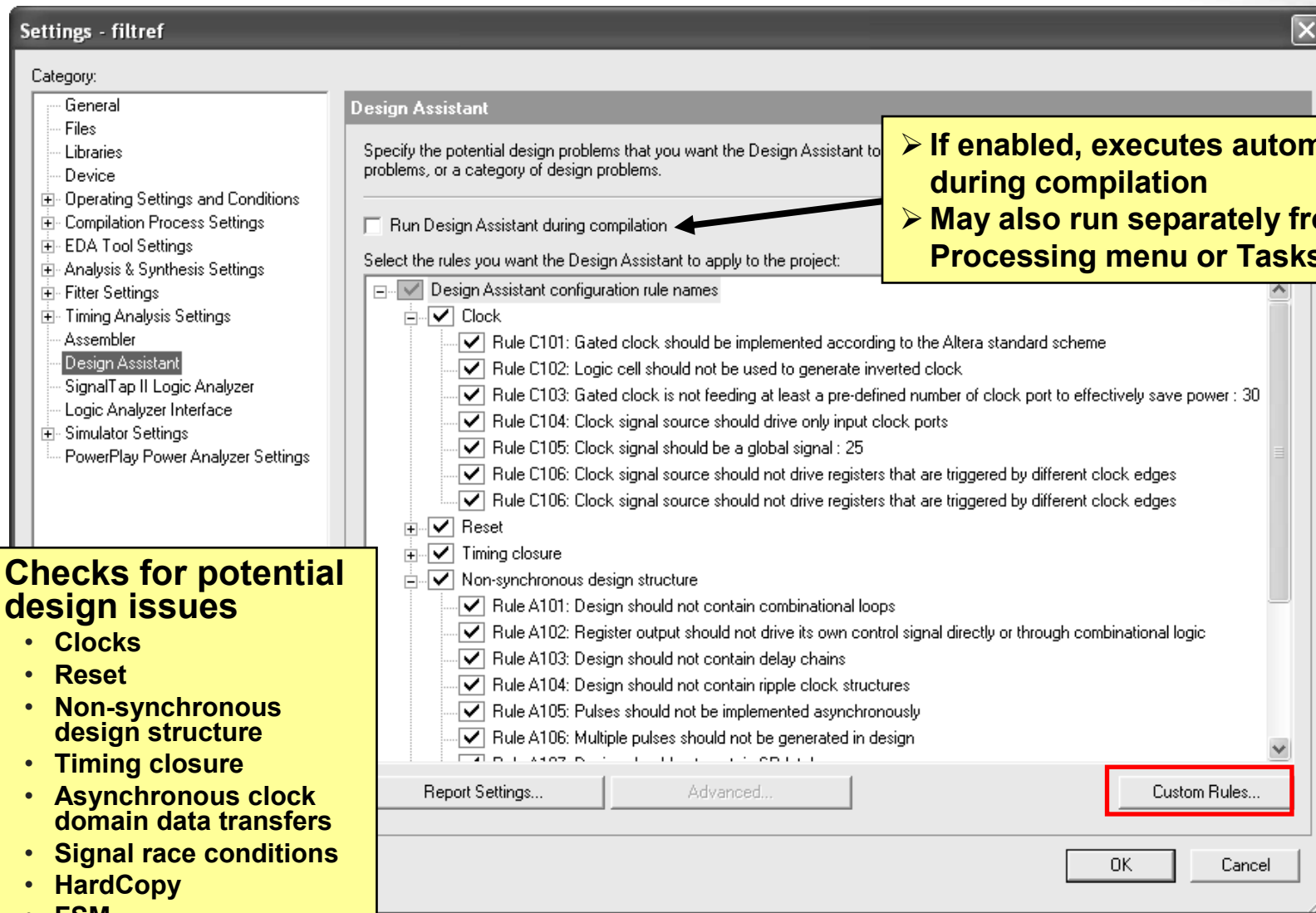
```
Tcl: set_instance_assignment -name STRATIXII_OPTIMIZATION_TECHNIQUE SPEED -to <node name>
```



Updating QSF File

- QSF not updated automatically when constraint entered or Assignment Editor saved
- QSF updated only when
 - Project is saved (**File** menu)
 - Beginning of compilation
- Change behavior to updating assignments immediately (**Tools** menu ⇒ **Options** ⇒ **General** ⇒ **Processing**)
 - May impact software performance due to file accesses

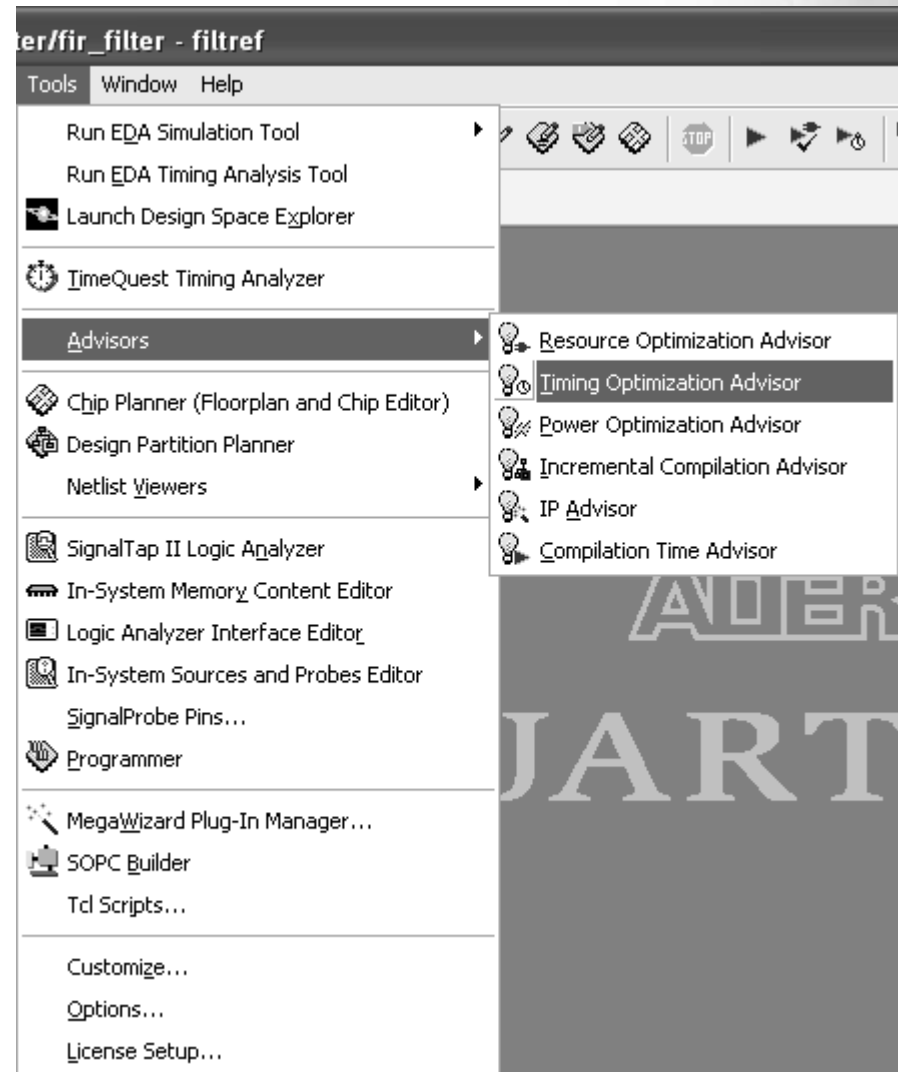
Design Assistant



- **Checks for potential design issues**
 - **Clocks**
 - **Reset**
 - **Non-synchronous design structure**
 - **Timing closure**
 - **Asynchronous clock domain data transfers**
 - **Signal race conditions**
 - **HardCopy**
 - **FSM**

Optimization Advisors

- Provide design-specific recommendations (feedback) on optimizing designs
- Access through **Tools** menu or Tasks window
- Six types
 - Resource usage optimization
 - Timing (performance) optimization
 - Power optimization
 - Incremental compilation suggestions
 - Implementing IP (DDR3 & PCIE)
 - Compilation time reduction



Example Optimization Advisor

Problem area identified

Three stages of recommendations in order of complexity

Green checkmark indicates settings already in use

Description of recommended settings

Links to adjust settings

Quartus II - D:/altera/71/qdesigns/fir_filter/fir_filter - filtref - [Timing Optimization Advisor]

File Edit Tools Window

Timing Summary

- How to use the Timing Optimization A
- General Recommendations
- Maximum Frequency (fmax)
 - Stage 1
 - Specify Fitter effort
 - Optimize for speed
 - Optimize specific clock domain
 - Turn off restructure multiplex
 - Use netlist optimizations
 - Turn on Auto Global Clock
 - Use global clocks
 - Stage 2
 - Turn on Ignore LCELL Buffers
 - Use physical synthesis optimiz
 - Run the Design Space Explor
 - Stage 3
 - Change state machine encodi
 - Flatten the hierarchy
 - Perform final placement optim
 - Set high synthesis effort
 - Duplicate logic for fan-out cor
 - Use LogicLock region assignm
 - Use location assignments & b.
 - Set maximum router effort
 - Avoid unrelated logic register
 - Optimize source code
- I/O Timing (tsu, tco, tpd)
- Hold Time & Minimum Delay Timing

Specify Fitter effort

Recommendation	Use Auto Fit or Standard Fit to meet timing requirements.
Description	You can adjust how much effort the fitter uses to get a fit on your design. Standard fit will try to maximize the design's timing performance. Auto Fit will try to meet a design's timing requirements without spending extra effort to exceed those requirements. The fast fit feature reduces the Fitter effort so that compilation time is shorter, but timing performance may be reduced.
Summary	Recommended changes have unknown effect on logic element usage, compilation time, and maximum frequency (fmax) for the design.
Action	Set the Fitter Effort to Auto Fit if it is currently set to Fast Standard Fit if it is currently set to Auto Fit in the Fitter Settings dialog box (Assignments menu). No action is needed for this recommendation. The recommendation is already made. Current Global Settings: Fitter Effort = STANDARD FIT (Recommended: STANDARD FIT or Auto Fit) Open Settings dialog box - Fitter Settings page





Quartus II Software Design Series: Foundation

I/O Planning



I/O Planning Need

- I/O standards increasing in complexity
- FPGA/CPLD I/O structure increasing in complexity
 - Results in increased pin placement guidelines
- PCB development performed simultaneously with FPGA design
 - Sometimes before!
- Pin assignments need to be verified earlier in design cycle
- Designers need easy way to transfer pin assignments into board tools

Pin Planner Window

Top View - Wire Bond
Cyclone II - EP2C5F256C6

Package View (Top or Bottom)

Groups list

Node Name	Direction	Location	I/O Bank	Vref Group	I/O Standard	Reserved
clk1	Input				3.3-V LVTTTL (default)	Reserved
dataa[7]	Input				3.3-V LVTTTL (default)	
dataa[6]	Input				3.3-V LVTTTL (default)	
dataa[5]	Input				3.3-V LVTTTL (default)	
dataa[4]	Input				3.3-V LVTTTL (default)	
dataa[3]	Input				3.3-V LVTTTL (default)	
dataa[2]	Input				3.3-V LVTTTL (default)	
dataa[1]	Input				3.3-V LVTTTL (default)	

All Pins list



Assigning Pins Using Pin Planner

Drag & drop single pin; tooltips provide pin information

Named:	Node Name	Direction	Location
1	clk	Input	
2	clkx2	Input	
3	d[7]	Input	
4	d[6]	Input	
5	d[5]	Input	
6	d[4]	Input	
7	d[3]	Input	
8	d[2]	Input	

Drag & drop multiple highlighted pins or buses

Choose one-by-one or pin alignment direction (Pin Planner toolbar or Edit menu)

Named:	Node Name	Direction	Location
1	clk	Input	
2	clkx2	Input	
3	d[7]	Input	
4	d[6]	Input	
5	d[5]	Input	
6	d[4]	Input	
7	d[3]	Input	
8	d[2]	Input	

Assigning Pins Using Pin Planner (2)

Double-click pin or I/O bank to open Properties dialog box

Drag & drop to I/O bank, VREF block, or device edge

Filter nodes displayed

Pin Properties

Pin number: PIN_B16
 Node name: d[4]
 I/O standard: 3.3-V LVTTTL (default)
 Reserved:
 Properties:

Name	Value
I/O Bank	3
General Function	Column I/O
Special Function	DQSn11T
Pad ID	345
VREF Pad ID	N/A

Pin List Table:

Node Name	Direction	Location	I/O Bank
d[7]	Input	PIN_B19	3
d[6]	Input	PIN_B18	3
d[5]	Input	PIN_B17	3
d[4]	Input	PIN_B16	3
d[3]	Input	IOBANK_3	3
d[2]	Input	IOBANK_3	3
d[1]	Input	IOBANK_3	3
d[0]	Input	IOBANK_3	3

Filter Menu:

- Pins: all
- Pins: assigned
- Pins: unassigned
- Pins: input
- Pins: output
- Pins: bidirectional
- Pins: all
- customize>>

Reserving I/O Pins

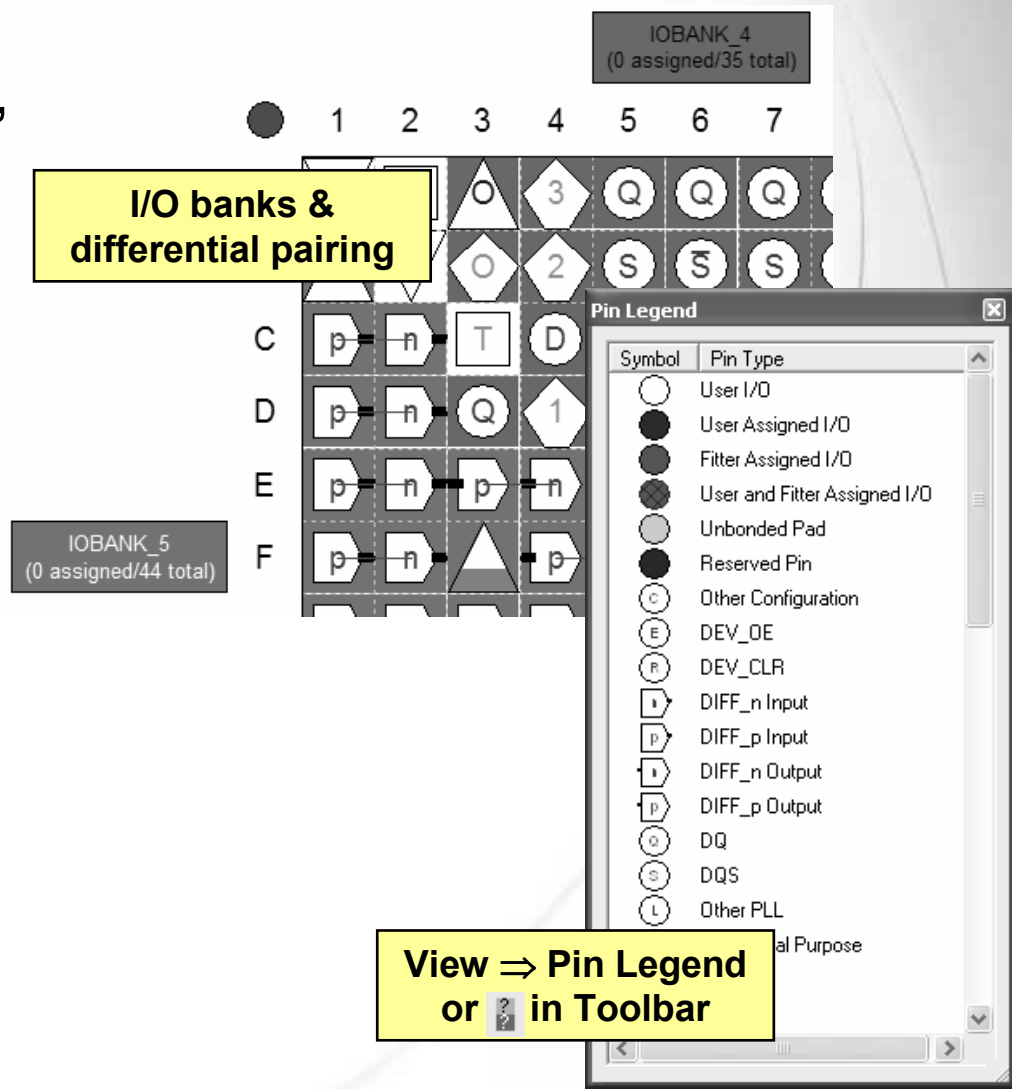
- Type reserved I/O name directly into Pins List & select reserve configuration

Node Name	Direction	Location	I/O Bank	Vref Group	I/O Standard	Reserved
yn_out[4]	Output				3.3-V LVTTTL (default)	
yn_out[3]	Output				3.3-V LVTTTL (default)	
yn_out[2]	Output				3.3-V LVTTTL (default)	
yn_out[1]	Output				3.3-V LVTTTL (default)	
yn_out[0]	Output				3.3-V LVTTTL (default)	
yvalid	Output				3.3-V LVTTTL (default)	
my_reserved_pin	Input				3.3-V LVTTTL (default)	
<<new node>>						

- Or right-click on pin in Package View and choose **Reserve** ⇒ **As...**
 - Pin name set to *user_reserve_<pin_number>*
 - Reserved pins colored blue in Package View
- Set initial state of other unused pins in Device settings in Settings dialog box (**Device & Pin Options** button)

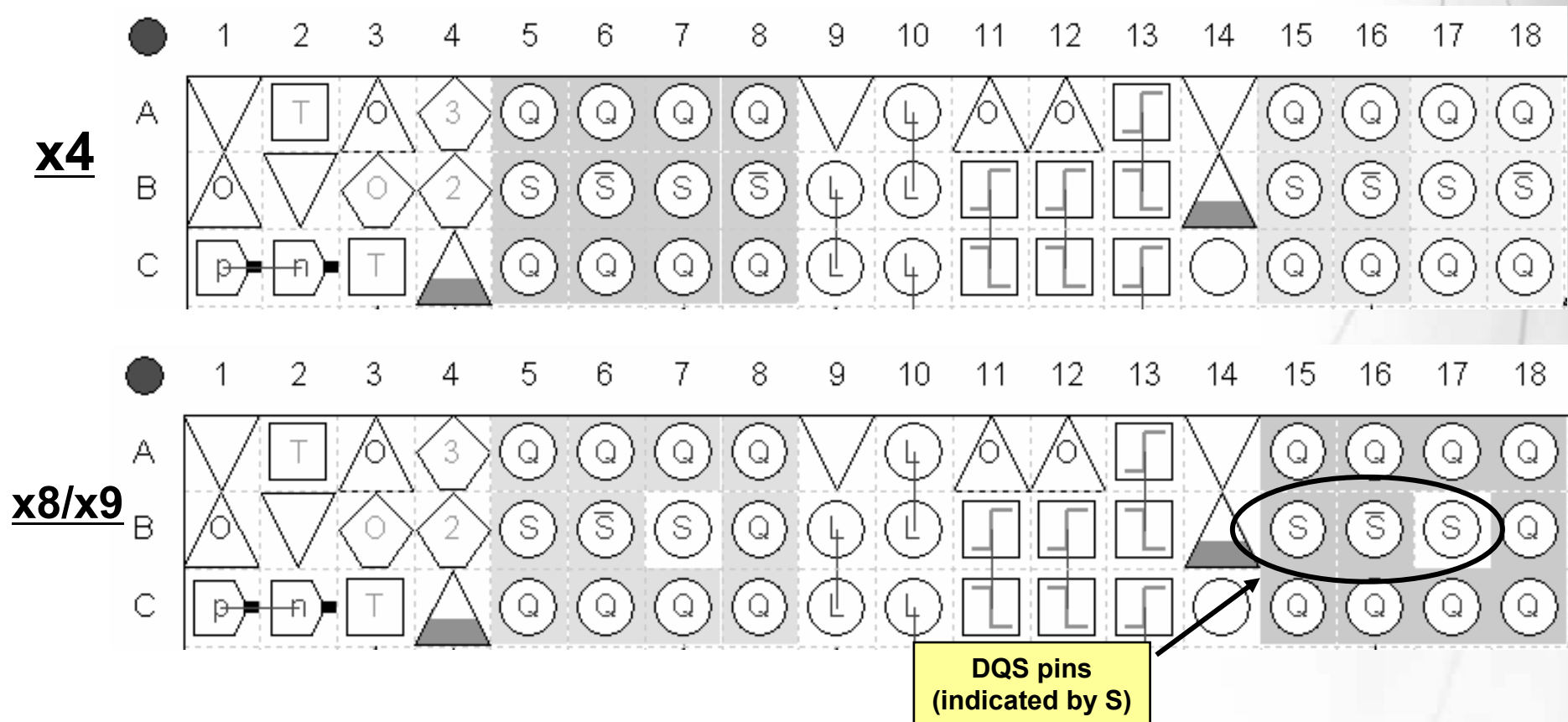
Other Pin Planner Features

- Displays (**View** ⇒ **Show**, Toolbar buttons, or right-click in Package View)
 - Device edges
 - I/O banks
 - VREF groups
 - Differential pin pairing
 - DQ/DQS pins (next slide)
- Easy-to-read pin legend



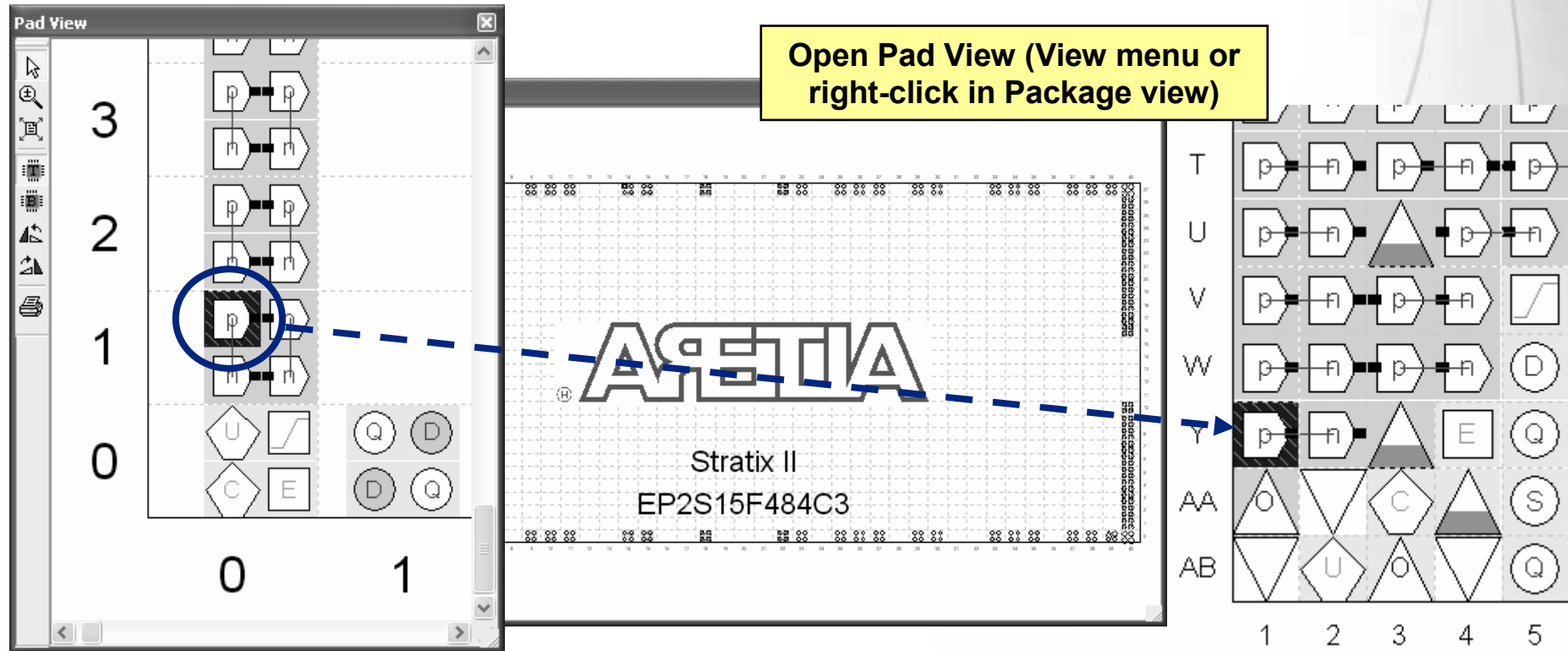
Show DQ/DQS Pins

- Show color-coded DQ/DQS sets in x4, x8/x9, x16/x18, or x32/x36 modes in the Package View for DDR interfaces



Pad View

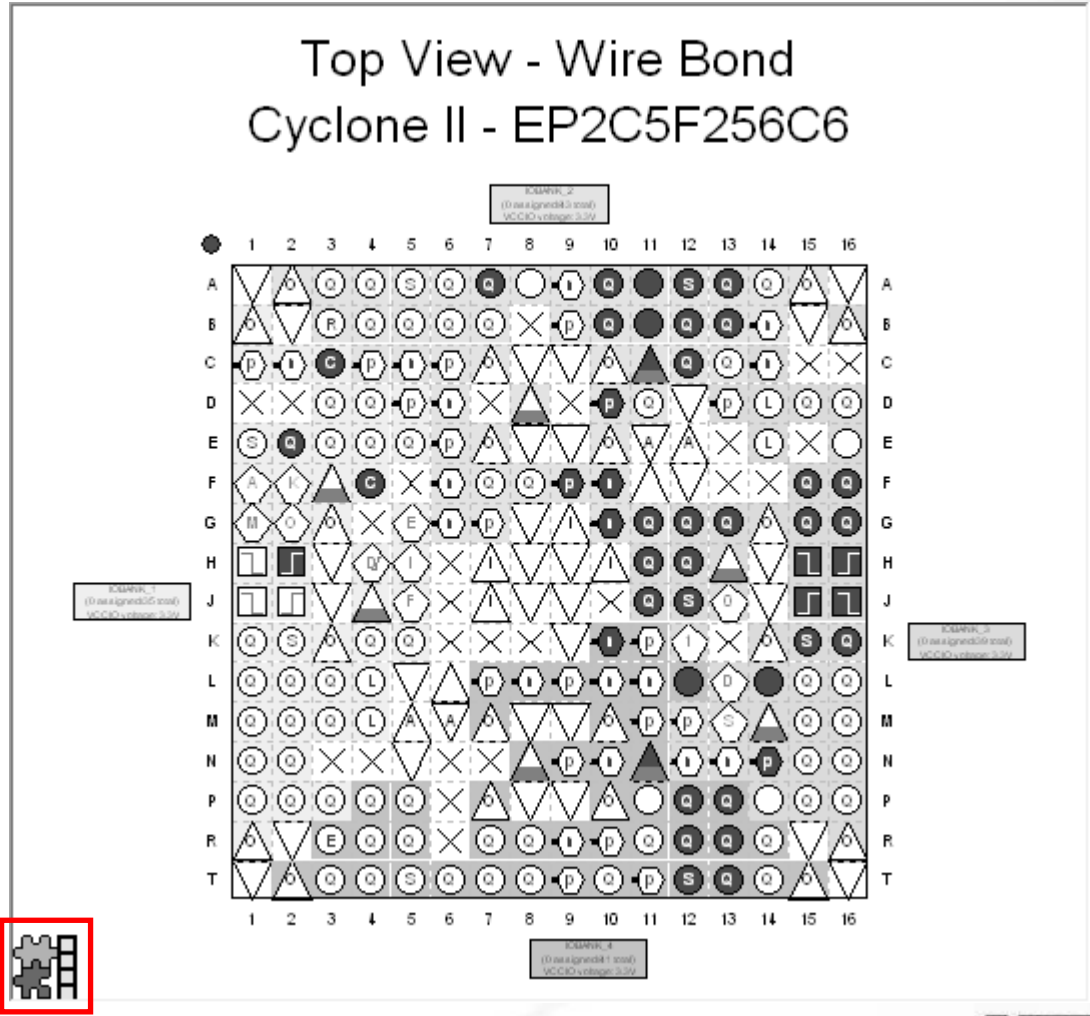
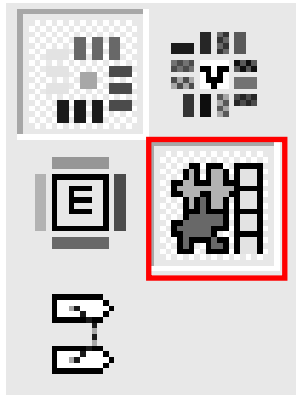
- Cross-reference package pin location to silicon pad location
 - Assign pins in Pad View based on pad location
- Reversed “Altera” indicates flip-chip die



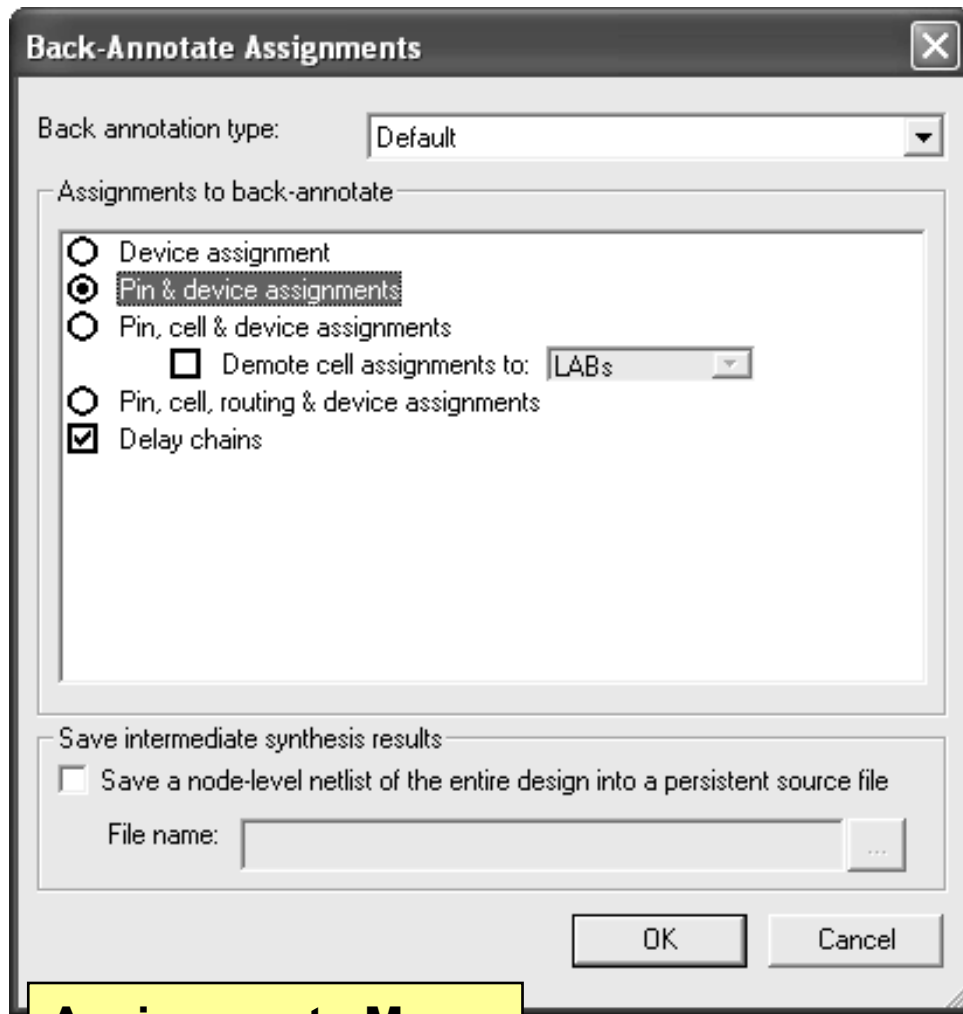
Show Fitter Placements

- View I/O locations automatically selected by Fitter

View ⇒ Show or in Toolbar

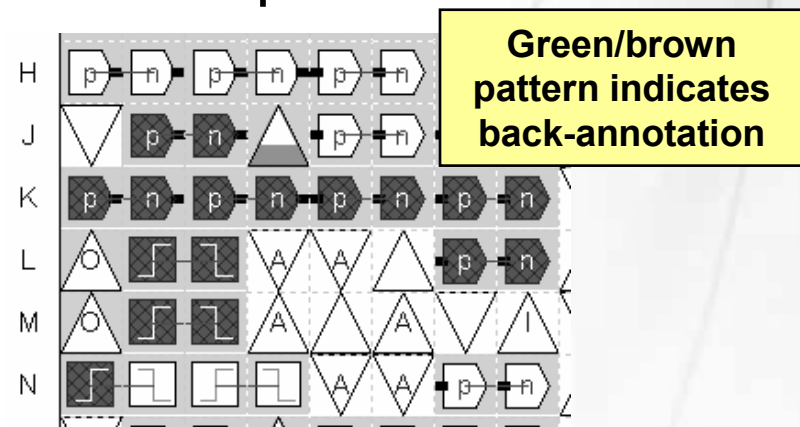


Back-Annotation



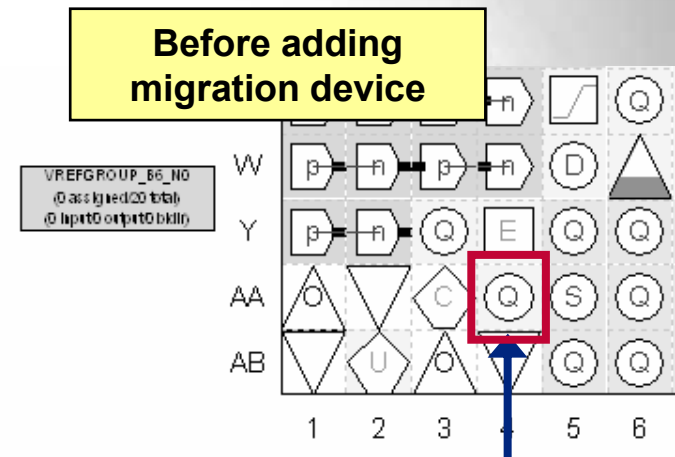
Assignments Menu

- Use to lock fitter-chosen (green) pin assignments for future compilations
 - Copies device & resource locations chosen by fitter into QSF file
 - Pins
 - Logic
 - Routing
- “Locks down” locations in floorplan



Pin Migration View

- Select migration devices in Device Settings
- View & compare pin function differences between migration devices
- Package View adjusts to prevent non-migratable assignments



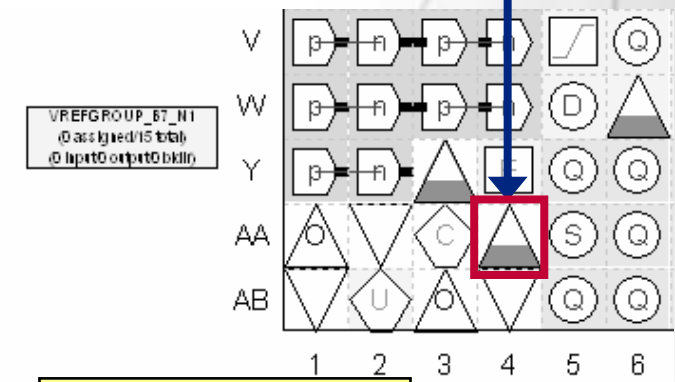
View ⇒ Pin Migration View or right-click in Package View

Pin Migration View

Current Device: EP2515F484C3

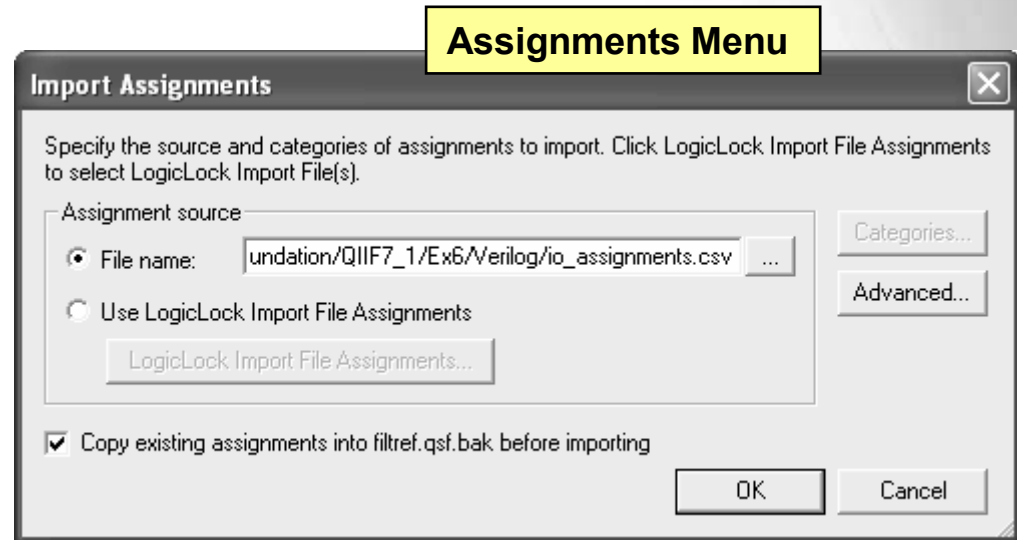
Pin Number	Pin Function	Migration Result			Migration Devices					
		Pin Function	I/O Bank	VREF Group	EP2515F484C3			EP2560F484I4		
1	PIN_A6	Column I/O	4	B4_N1	Column I/O	4	B4_N1	Column I/O	4	B4_N2
2	PIN_A7	Column I/O	4	B4_N1	Column I/O	4	B4_N1	Column I/O	4	B4_N2
3	PIN_A8	Column I/O	4	B4_N1	Column I/O	4	B4_N1	Column I/O	4	B4_N2
4	PIN_A10	Column I/O	9	B4_N1	Column I/O	9	B4_N1	Column I/O	9	B4_N2
5	PIN_A16	Column I/O	3	B3_N0	Column I/O	3	B3_N0	Column I/O	3	B3_N1
6	PIN_A17	Column I/O	3	B3_N0	Column I/O	3	B3_N0	Column I/O	3	B3_N1
7	PIN_A18	Column I/O	3	B3_N0	Column I/O	3	B3_N0	Column I/O	3	B3_N1
8	PIN_A19	Column I/O	3	B3_N1	Column I/O	3	B3_N1	Column I/O	3	B3_N2
9	PIN_A21	Dedicated...	2	B2_N1	Dedicated...	2	B2_N1	Dedicated...	2	B2_N2
10	PIN_AA4	VREFB7N2	7		Column I/O	7	B7_N1	VREFB7N2	7	B7_N2
11	PIN_AA3	Column I/O	7	B7_N0	Column I/O	7	B7_N0	Column I/O	7	B7_N1
12	PIN_AA12	Column I/O	8	B8_N1	Column I/O	8	B8_N1	Column I/O	8	B8_N2
13	PIN_AA13	Column I/O	8	B8_N1	Column I/O	8	B8_N1	Column I/O	8	B8_N2

Device... Pin Finder... Show only highlighted pins Show migration differences



Import/Export via CSV

- Use spreadsheet Comma Separated Value (.CSV) file to enter or edit I/O locations
- Convenient for transferring assignments between project revisions
- CSV column names must match Pin Planner column headings
 - To
 - Pin name
 - Assignment Name
 - Location
 - Value
 - PIN_<pin_number>
 - I/O standard



	A	B	C
1	To	Assignment Name	Value
2	d[7]	Location	PIN_J4
3	d[6]	Location	PIN_H4

Type I/O Assignments & Scripting

- Type pin-related assignments directly into QSF
- Type pin-related assignments into separate Tcl
 - Source Tcl file in project QSF
 - Execute Tcl file to write assignments into QSF

```
Quartus II - D:/altera/71/qdesigns/fir_filter/fir_filter - filtref - [io_assignments.tcl]
File Edit View Project Processing Tools Window
25
1
2 set_location_assignment IOBANK_4 -to reset
3 set_location_assignment IOBANK_4 -to yn_out
4 set_location_assignment -name RESERVE_PIN "AS INPUT TRI-STATED" -to me
5 set_location_assignment PIN_E14 -to yvalid
6 set_location_assignment PIN_C13 -to clkx2
7 set_location_assignment PIN_C16 -to newt
8

Quartus II - D:/altera/71/qdesigns/fir_filter/fir_filter - filtref - [filtref.qsf]
File Edit View Project Processing Tools Window
25
40 set_global_assignment -name VECTOR_WAVEFORM_FILE fir.vwf
41
42 # Pin & Location Assignments
43 # =====
44 set_location_assignment PIN_N20 -to clk
45 source io_assignments.tcl
46
```


I/O Assignment Analysis Command



Run from Pin Planner toolbar

- Use to check legality of all I/O assignments without full compilation
- Requirements
 - I/O declaration
 - HDL port declaration
 - Reserved pin
 - Pin-related assignments
 - I/O standard
 - Current strength
 - Pin location (pin, bank, edge)
 - PCI clamping diode
 - Toggle rate

Processing menu ⇒
Start ⇒ Start I/O
Assignment Analysis
or Tasks window



I/O Assignment Analysis Output

Compilation Report (Fitter section)

- Pin-out file
- I/O pin tables
- Output pin loading
- I/O rules checking*

Partial placement results also shown in floorplan

Detailed messages on I/O assignment issues

- Compiler assumptions
- Device & pin migration issues
- I/O bank voltages & standards

Flow Summary

Flow Status	Successful - Tue May 08 16:00:56 2007
Quartus II Version	7.1 Build 156 04/30/2007 SJ Full Version
Revision Name	filtrf
Top-level Entity Name	filtrf
Family	Stratix II
Device	EP2S15F484C3
Timing Models	Final
Met timing requirements	N/A
Logic utilization	N/A
Combinational ALUTs	58
Dedicated logic registers	58
Total registers	58
Total pins	22 / 335 (7 %)
Total virtual pins	0
Total block memory bits	0 / 419,328 (0 %)
DSP block 9-bit elements	0 / 96 (0 %)
Total PLLs	0 / 6 (0 %)
Total DLLs	0 / 2 (0 %)

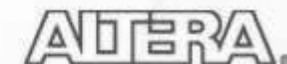
Messages

Type	Flag	Message
Info		Info: Selected device migration path implemented 8 pin(s) as GND
Warning		Warning: Devices selected for migration have different speed grades
Info		Info: Selected device migration path cannot use 8 pins as regular I/Os
Info		Info: Selected device migration path cannot use 8 pins as DQS I/Os
Info		Info: Selected device migration path cannot use 5 pins as nDQS I/Os
Info		Info: Selected device migration path cannot use 2 pins as DQ I/Os
Info		Info: Fitter converted 1 user pins into dedicated programming pins

*Note: See Appendix for special reports and information generated only for Arria GX, Stratix II, II GX, and HardCopy II devices

© 2008 Altera Corporation—Confidential

Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation



Live I/O Checking

- Perform limited I/O checks live as assignments made
- Status window alerts to failing assignments
 - Errors detailed in Messages window and Package view tooltips
- Full I/O Assignment Analysis still required



In Pin Planner, View menu ⇒ Live I/O Check Status Window

Turn on or off in Pin Planner toolbar

Live I/O Check Status

Performing live I/O check provides preliminary I/O pin verification. Run the Filter for complete I/O pin verification.

Detailed message information appears in the Messages window.

✘ 0 error messages
 ⚠ 3 warning messages

Live I/O check passed

Turn Off Live I/O Check

Live I/O Check Status

Performing live I/O check provides preliminary I/O pin verification. Run the Filter for complete I/O pin verification.

Detailed message information appears in the Messages window.

✘ 8 error messages
 ⚠ 3 warning messages

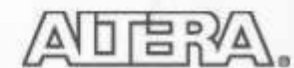
Live I/O check failed

Turn Off Live I/O Check

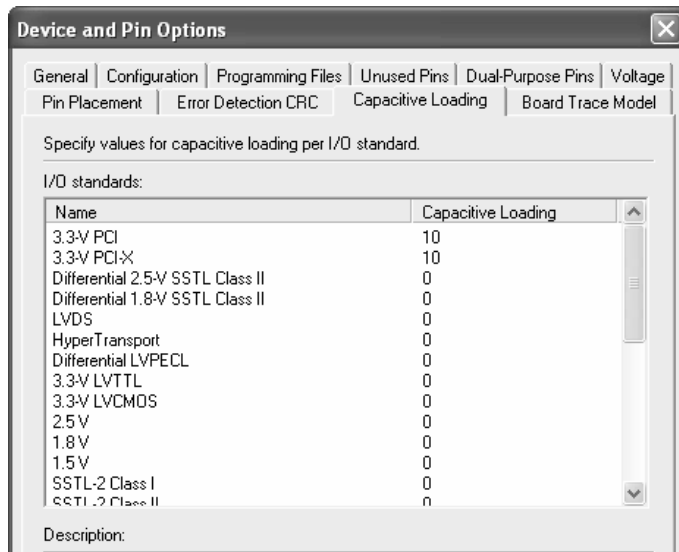
Type	Flag	Message
✘	+	Error: I/O bank 2 contains input or bidirectional pins with I/O standards that make it impossible to choose a le
✘	+	Error: I/O bank 2 contains input or bidirectional pins with I/O standards that make it impossible to choose a le
✘	-	Error: I/O bank 2 contains input or bidirectional pins with I/O standards that make it impossible to choose a le
	i	Info: Can't select VCCIO 2.5V for I/O bank due to 1 input or bidirectional pins
	i	Info: Can't select VCCIO 3.3V for I/O bank due to 1 input or bidirectional pins
	⚠	Warning: The Reserve All Unused Pins setting has not been specified, and will default to 'As output driving grou
	✘	Error: Live I/O check failed

I/O BANK_2
(16 assigned/43 total)
(11 input/5 output/0 bi-dir)

7 8 9 10



“Board-Aware” Settings: Output Pin Load

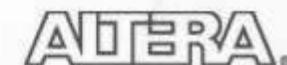


Capacitive Loading tab of Device and Pin Options button in Device Settings

- Specifies output pin loading in picofarads (pf)
 - Changes default loading value of I/O standard
 - Changes t_{co} of output pins
- Allows designer to accurately model board conditions
- Specify for entire I/O standard in Device Settings
- Apply to individual output or bidirectional pins in Assignment Editor or Pin Planner All Pins list

Node Name	Direction	Location	I/O Standard	Output Pin Load	I/O Bank
13	reset	PIN_N3	3.3-V LVTTTL (default)		B
14	yn_out[7]	PIN_J6	3.3-V LVTTTL (default)	20	B
15	yn_out[6]	PIN_L8	3.3-V LVTTTL (default)	20	B
16	yn_out[5]	PIN_H1	3.3-V LVTTTL (default)	20	B
17	yn_out[4]	PIN_K2	3.3-V LVTTTL (default)	20	B
18	yn_out[3]	PIN_H2	3.3-V LVTTTL (default)	20	B
19	yn_out[2]	PIN_J5	3.3-V LVTTTL (default)	20	B
20	yn_out[1]	PIN_L2	3.3-V LVTTTL (default)	20	B
21	yn_out[0]	PIN_K5	3.3-V LVTTTL (default)	20	B
22	yvalid	PIN_L7	3.3-V LVTTTL (default)		B
23	~DATA0~	PIN_E13	3.3-V LVTTTL (default)		B

```
Tcl: set_instance_assignment -name OUTPUT_PIN_LOAD <value> -to <pin name>
```



Advanced I/O Timing

- Enhances analysis (over capacitive loading) by allowing user to enter board-level parameters (Cyclone III, Stratix II, & Stratix III devices only)
 - Use in lieu of or in addition to HSPICE & IBIS modeling
- View signal integrity metrics in Compilation Report (TimeQuest folder)

Enable in TQ settings, then Device Settings ⇒ Device & Pin Options

Specify values for Board Trace Model per I/O standard.

I/O standard: 3.3-V LVTTTL

Board trace model:

Name	Value
Near pull-up resistance	open
Near pull-down resistance	open
Near capacitance	open
Near series resistance	short
Transmission line distributed inductance	0
Transmission line distributed capacitance	0
Transmission line length	0
Far pull-up resistance	open
Far pull-down resistance	open
Far capacitance	open
Far series resistance	short
Termination voltage	0

Description:
Specifies board trace, termination, and capacitive load parameters for each I/O standard. Note: These settings affect Advanced I/O Timing only and are used instead of Capacitive Loading to determine I/O timing and power. If Advanced I/O

Set for all pins using I/O standard

Set parameters for specific I/O pin(s)

Stratix II
EP2S15F484C3
pin(s): yn_out[7]; yn_out[6]; yn_out[5]; yn_out[4]; yn_out[3]; yn_out[2]; yn_out[1]; yn_out[0]
I/O standard for selected pin(s): 3.3-V LVTTTL

on chip/off chip

Vtt: 0 V

Rnh: open Ohm

Rns: 22 Ohm

Rnl: open Ohm

Rfl: open Ohm

Rfs: short Ohm

Rfh: open Ohm

Cn: open F

Cf: open F

C_per_length: 0 F/inch

L_per_length: 0 H/inch

T_length: 0 Inch(es)

C_f parameter equivalent to output pin load

Right-click on output pin(s) in Pin Planner ⇒ Board Trace Model



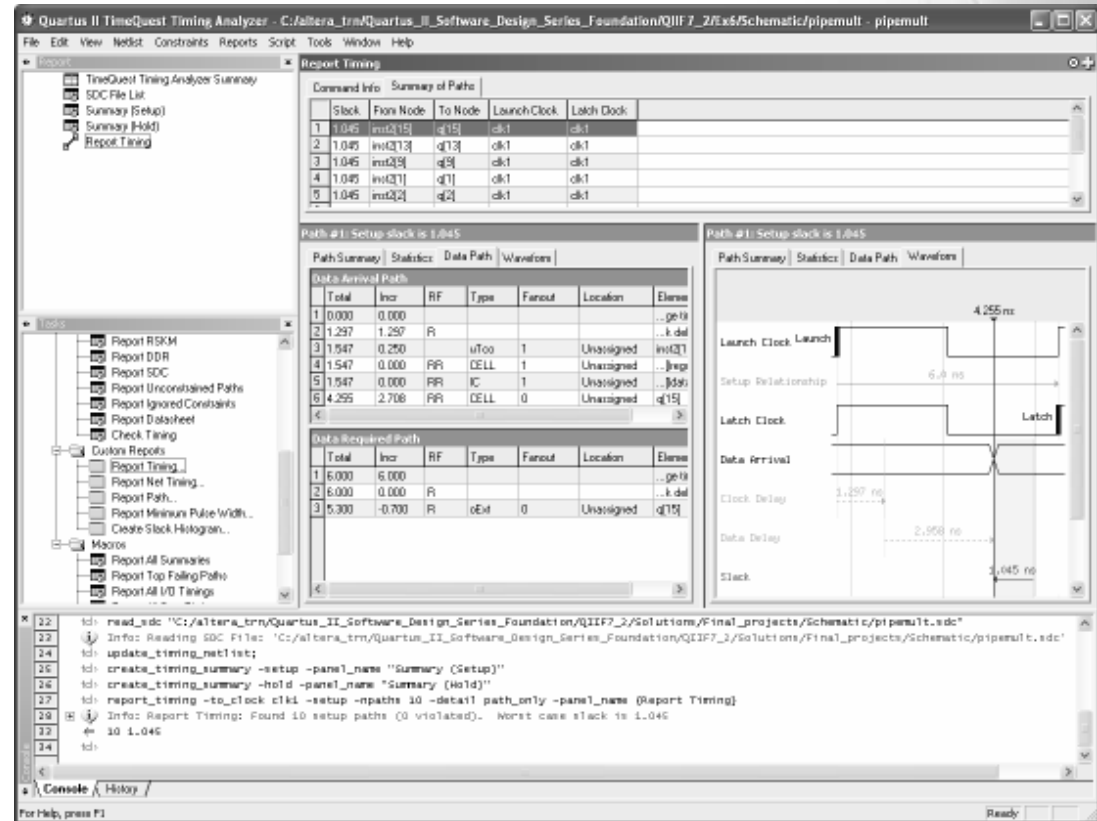
Quartus II Software Design Series: Foundation

Timing Analysis

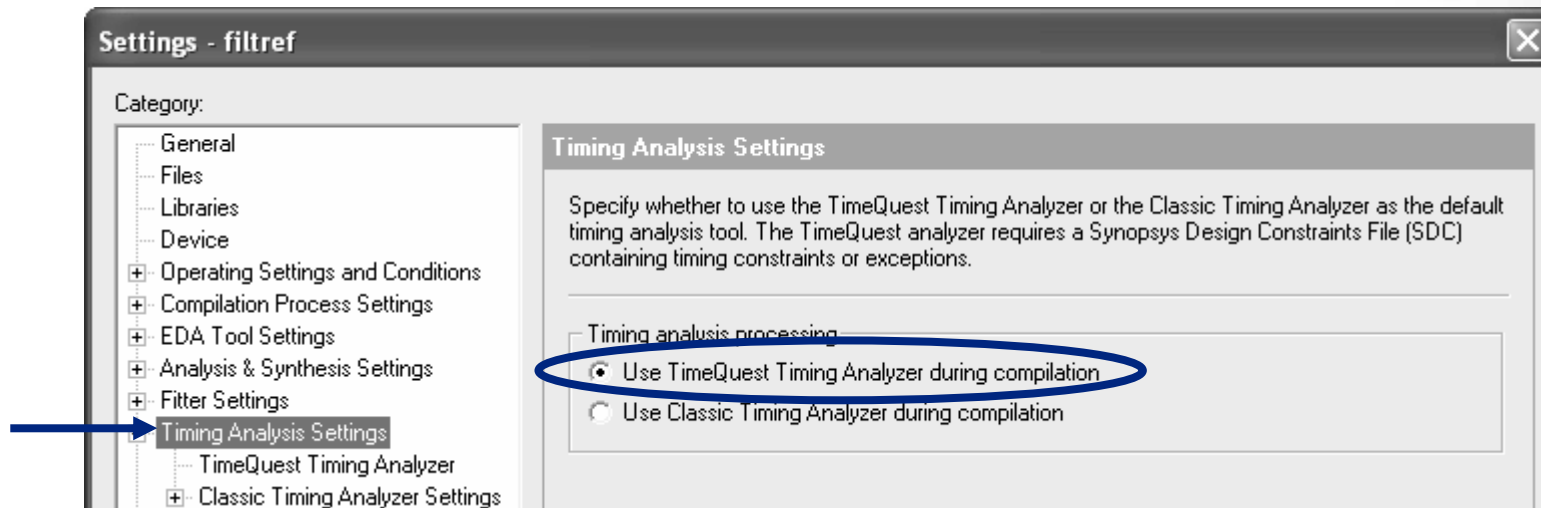


TimeQuest Timing Analyzer (TA)

- Timing engine in Quartus II software
- Provides timing analysis solution for all levels of experience
- Features
 - Synopsys Design Constraints (SDC) support
 - Standardized constraint methodology
 - Easy-to-use interface
 - Constraint entry
 - Standard reporting
 - Scripting emphasis
 - Presentation focuses on using GUI



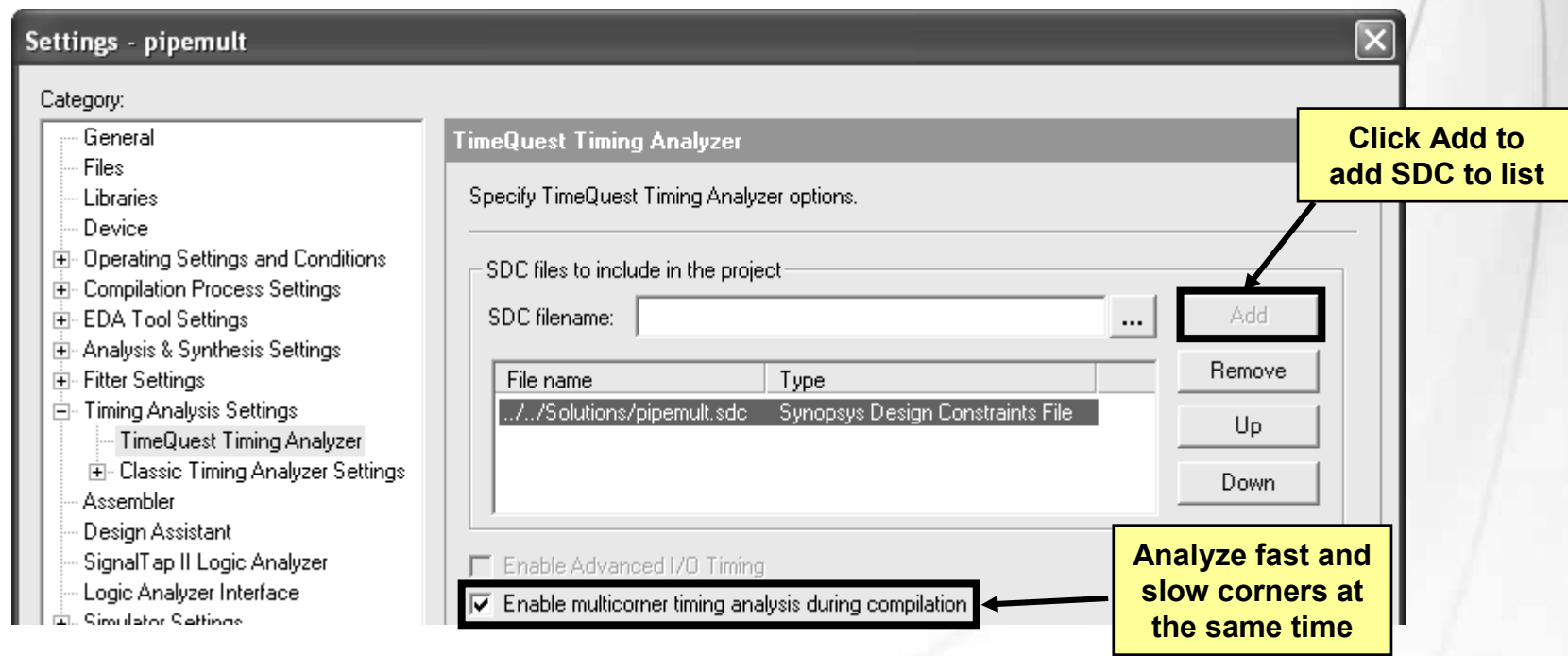
Enabling in the Quartus II Software




- Notes:**
- Arria GX device *only* supports Timequest TA.
 - TimeQuest TA is enabled by default for new Stratix III and Cyclone III designs.

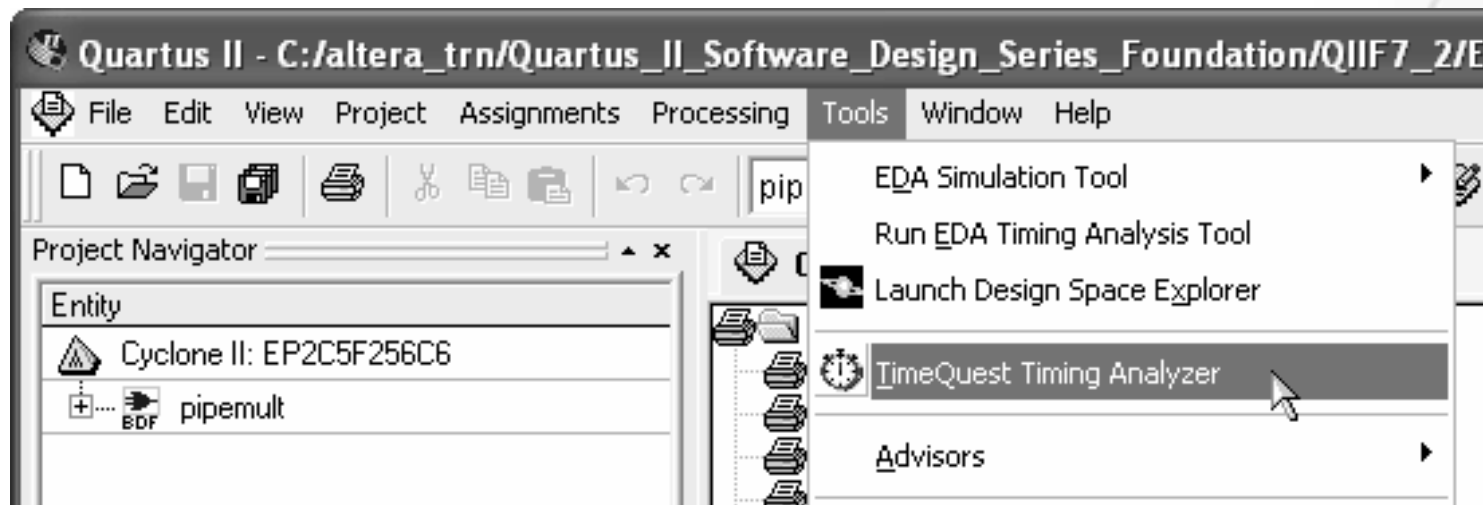
Adding SDC File to Quartus II Project

- Add SDC files to TimeQuest Timing Analyzer page of Settings dialog box
- Multicorner timing analysis checks all process corners in one analysis
 - On by default for Cyclone II & III, Stratix II & III devices



Opening the TimeQuest Interface

- Toolbar button 
- **Tools** menu
- Tasks window
- Stand-alone mode
 - `quartus_staw`
- Command line



TimeQuest GUI

Menu access all TimeQuest features

Report Pane

View Pane

Tasks Pane

Console Pane

The screenshot shows the TimeQuest GUI with several panes highlighted by yellow boxes:

- Menu:** File, Edit, View, Netlist, Constraints, Reports, Script, Tools, Window, Help
- Report Pane:** Contains a tree view with items like "TimeQuest Timing Analyzer Summary", "SDC File List", "Summary (Setup)", "Summary (Hold)", and "Report Timing".
- Tasks Pane:** Contains a list of tasks such as "Report RSKM", "Report DDR", "Report SDC", "Report Unconstrained Paths", "Report Ignored Constraints", "Report Datasheet", "Check Timing", "Custom Reports", "Report Timing...", "Report Net Timing...", "Report Path...", "Report Minimum Pulse Width...", "Create Slack Histogram...", "Macros", "Report All Summaries", "Report Top Failing Paths", and "Report All I/O Timings".
- View Pane:** Displays timing information for "Path #1: Setup slack is 1.045". It includes a "Data Arrival Path" table and a "Data Required Path" table. The "Data Arrival Path" table is as follows:

	Total	Incr	RF	Type	Fanout	Location	Element
1	0.000	0.000					...ge tir
2	1.297	1.297	R				...k del
3	1.547	0.250		uTco	1	Unassigned	inst2[1
4	1.547	0.000	RR	CELL	1	Unassigned	...]regc
5	1.547	0.000	RR	IC	1	Unassigned	...]data
6	4.255	2.708	RR	CELL	0	Unassigned	q[15]

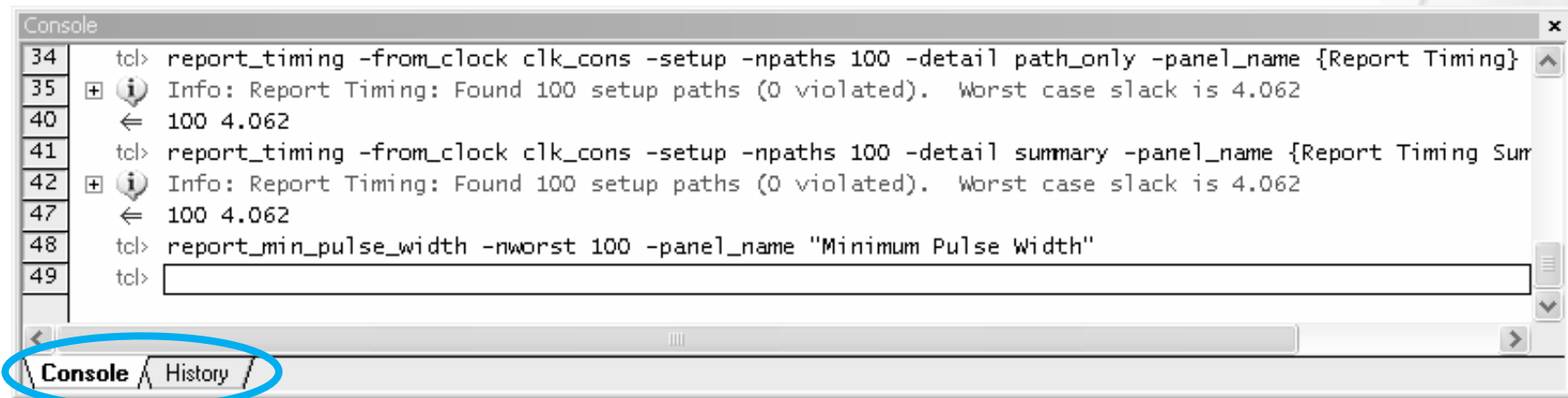
 The "Data Required Path" table is as follows:

	Total	Incr	RF	Type	Fanout	Location	Element
1	6.000	6.000					...ge tir
2	6.000	0.000	R				...k del
3	5.300	-0.700	R	oExt	0	Unassigned	q[15]
- Console Pane:** Shows a list of commands and their outputs, including "read_sdc", "update_timing_netlist", "create_timing_summary", and "report_timing".



Console pane

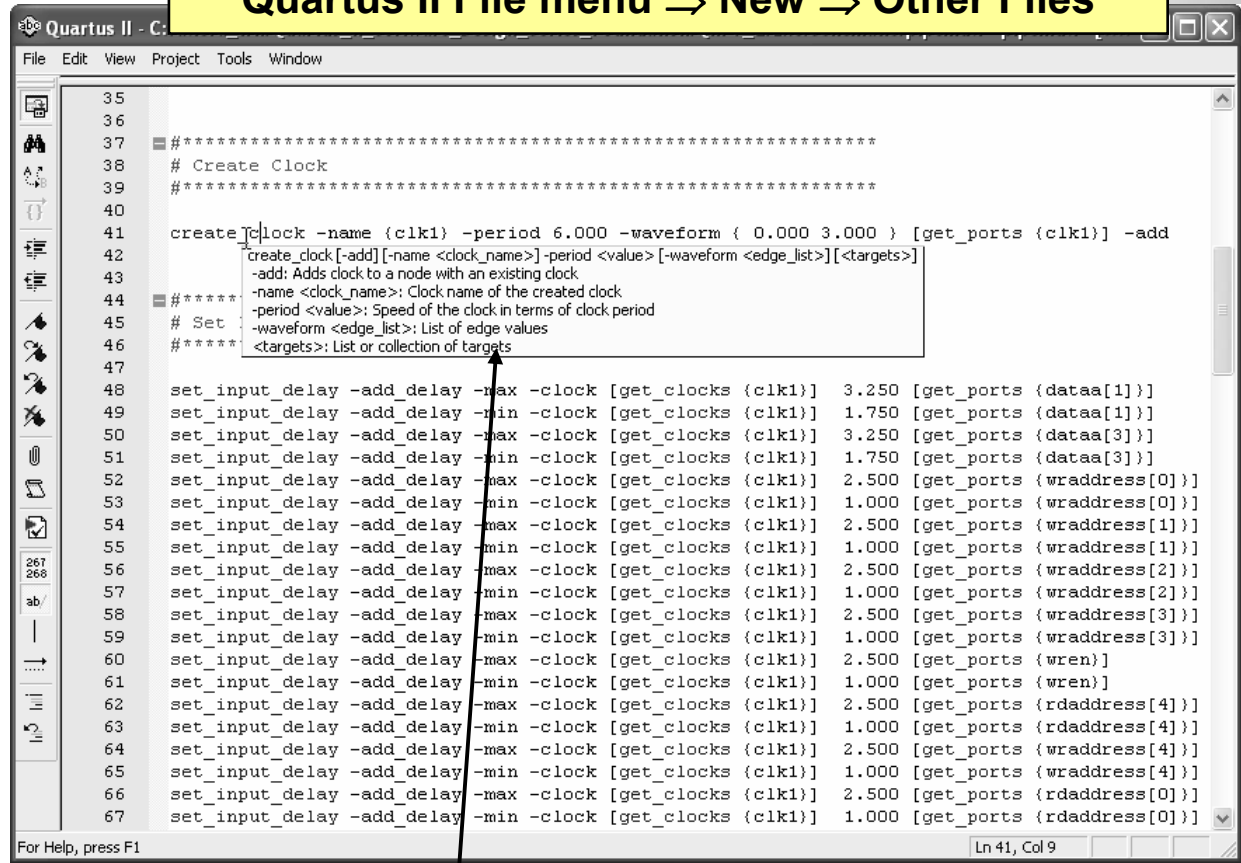
- Allows direct entry and execution of SDC & Tcl commands
 - Displays equivalent of command executed by GUI
- Displays TimeQuest output messages
- History tab records all executed SDC & Tcl commands
 - Copy & paste to create scripts or SDC files



SDC File Editor = Quartus II Text Editor

- Use Quartus II editor to create and/or edit SDC
- SDC editing unique features (for .sdc files)
 - Access to GUI dialog boxes for constraint entry (**Edit ⇒ Insert Constraint**)
 - Syntax coloring
 - Tooltip syntax help

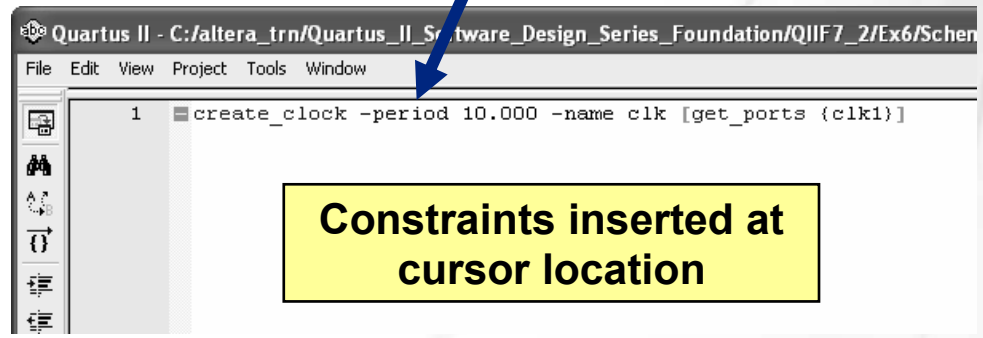
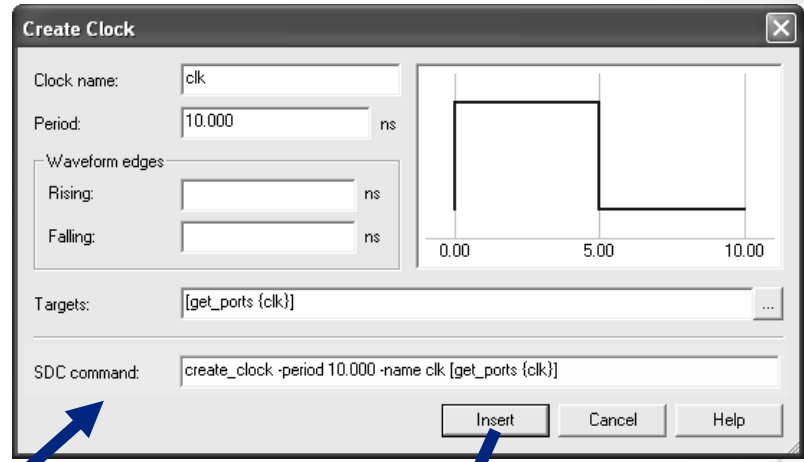
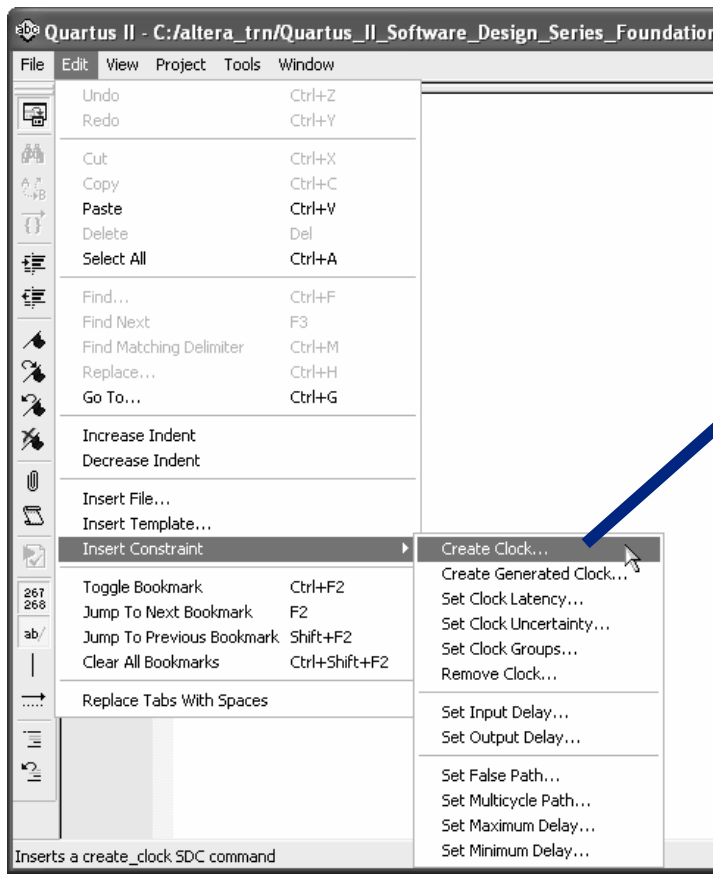
TimeQuest File menu ⇒ New/Open SDC File
Quartus II File menu ⇒ New ⇒ Other Files



Place cursor over command to see tooltip

SDC File Editor (cont.)

Construct an SDC file using the TimeQuest graphical constraint creation tools

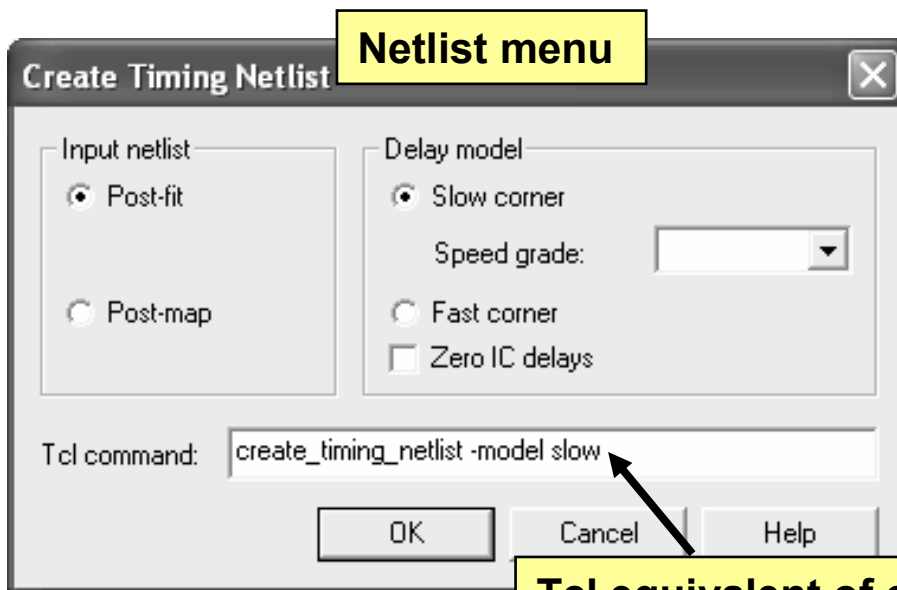


Steps to Using TimeQuest Tool

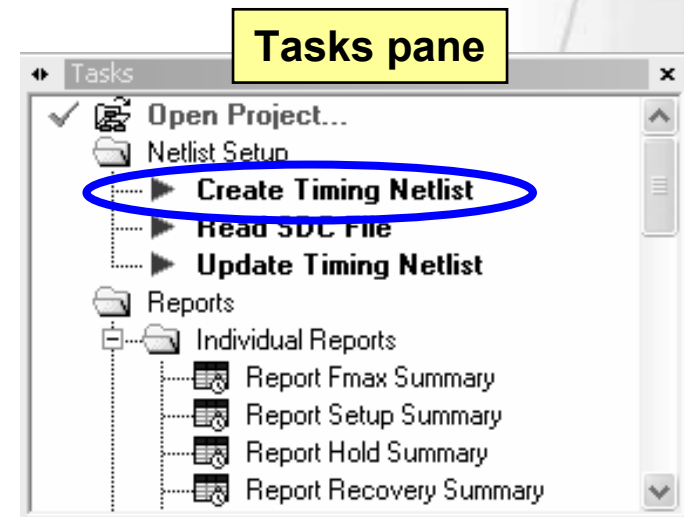
1. Generate timing netlist
2. Enter SDC constraints by creating or reading in an SDC file
3. Update timing netlist
4. Generate timing reports

1) Generate Timing Netlist

- Create a timing netlist (i.e. database) based on compilation results
 - Post-synthesis (mapping) or post-fit (if design already fully compiled)
 - Worst-case (slow; max. temp.), best-case (fast; min. temp.) timing models
 - Set custom operating conditions
- To execute:



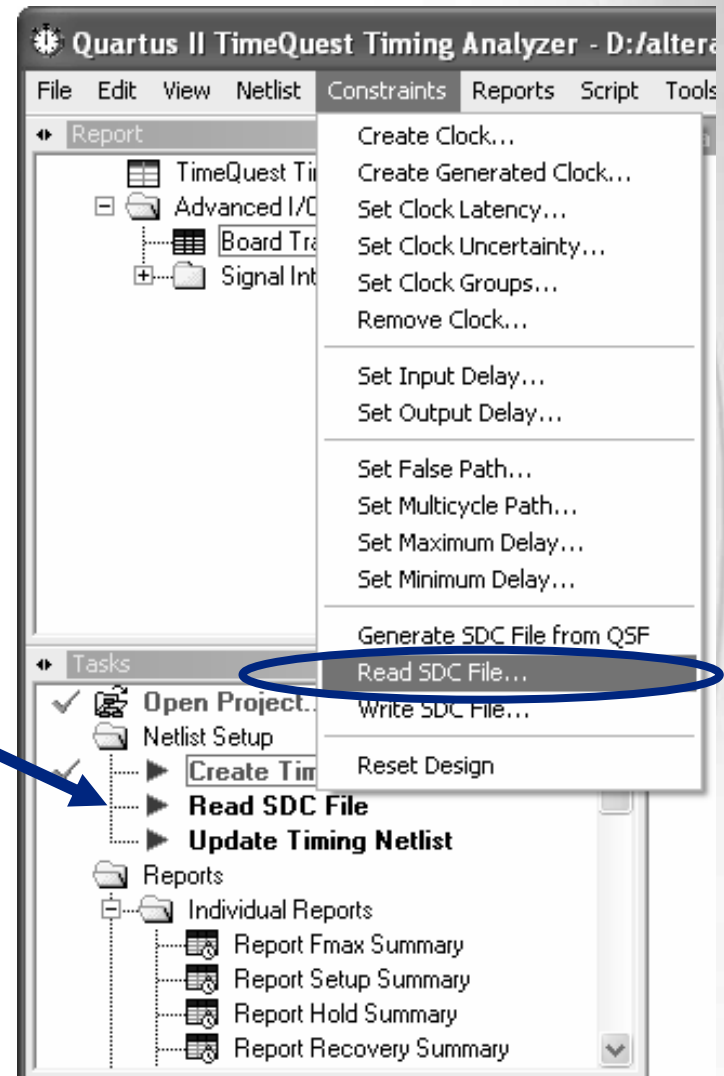
Tcl equivalent of command



Tcl: `create_timing_netlist`

2) Create or Read in SDC File

- Create SDC file using SDC file editor
 - Don't enter constraints using **Constraints** menu
- Read in constraints & exceptions from existing SDC file
 - Skip if no SDC file
- Execution
 - **Read SDC File** (Tasks pane or **Constraints** menu)
- File precedence (if no filename specified)
 - Files specifically added to Quartus II project
 - <current_revision>.sdc (if it exists in project directory)

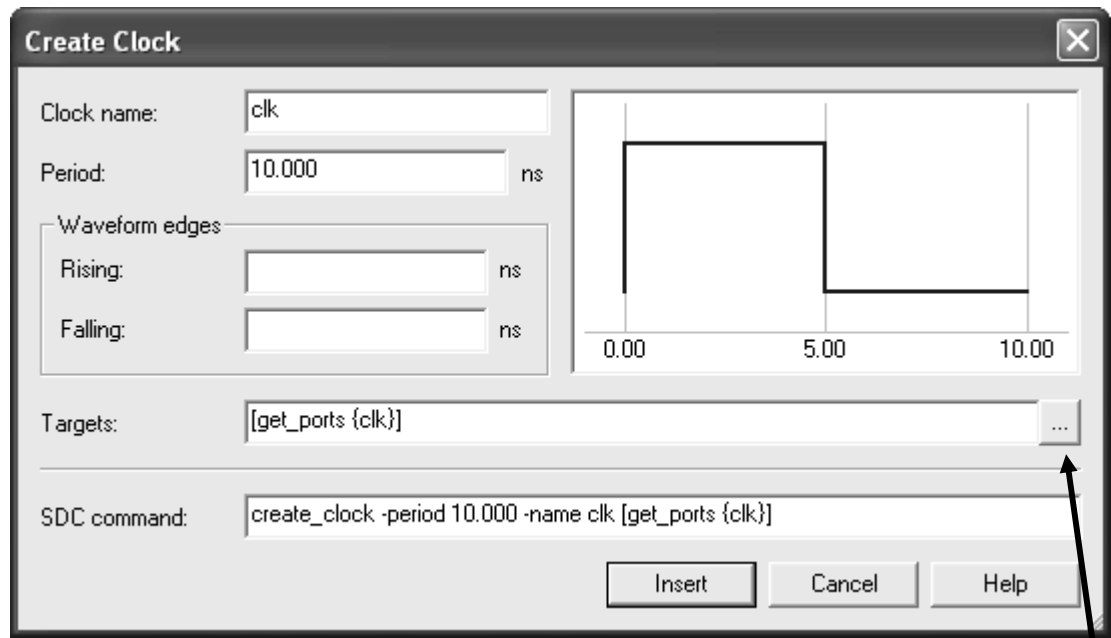


Tcl: read_sdc [<filename>]

Create Clock

In SDC File Editor,
Edit menu ⇒ Insert Constraint

- Create Clock fields:
- Clock Name – Assign name to clock setting; defaults to target node name
 - Period – Clock period in nanoseconds
 - Waveform edges – Use for non-50% duty cycle clocks
 - Targets – Port or pin to which clock setting is being applied



Name Finder (next slide)

Important Note: All design clocks are related by default. This means the timing analyzer will analyze paths between clock domains whether you have specifically related them or not.

SDC: create_clock

Name Finder

- Search the SDC netlist for node names
 - Similar to the Quartus II Node Finder

Select collection to search

Edit command here or final command to use wildcards

Name Finder

Collection: **get_ports** Filter: *

Options

- Case-insensitive
- Hierarchical
- Compatibility mode
- No duplicates

Matches

List

67 matches found

- clk_in_100mhz
- clkout
- din_a[0]
- din_a[1]
- din_a[2]
- din_a[3]
- din_a[4]
- din_a[5]
- din_a[6]
- din_a[7]
- din_b[0]
- din_b[1]
- din_b[2]
- din_b[3]

9 selected names

- clk_in_100mhz
- din_a[0]
- din_a[1]
- din_a[2]
- din_a[3]
- din_a[4]
- din_a[5]
- din_a[6]
- din_a[7]

SDC command: [get_ports {clk_in_100mhz din_a[0] din_a[1] din_a[2] din_a[3] din_a[4] din_a[5] din_a[6] din_a[7]}]

OK Cancel Help

Generated Clocks

- Clock signals derived from a previously created clock
 - E.g. clock dividers, ripple clocks, PLLs
 - Must be defined by a constraint

Create Generated Clock fields:

- Clock Name – Assign name to clock setting
- Relationship to source – Specify how generated clock is related to base clock. The Based on waveform section allows for more complexity in the relationship to the base clock (not discussed)
- Targets – Port or pin to which clock setting is being applied

SDC: `create_generated_clock`

In SDC File Editor,
Edit menu ⇒ Insert Constraint

Create Generated Clock

Clock name:

Source:

Relationship to source

Based on frequency

Divide by:

Multiply by:

Duty cycle:

Phase:

Offset:

Based on waveform

Edge list:

Edge shift list: ns ns ns

Invert waveform

Targets:

SDC command:

Insert Cancel Help

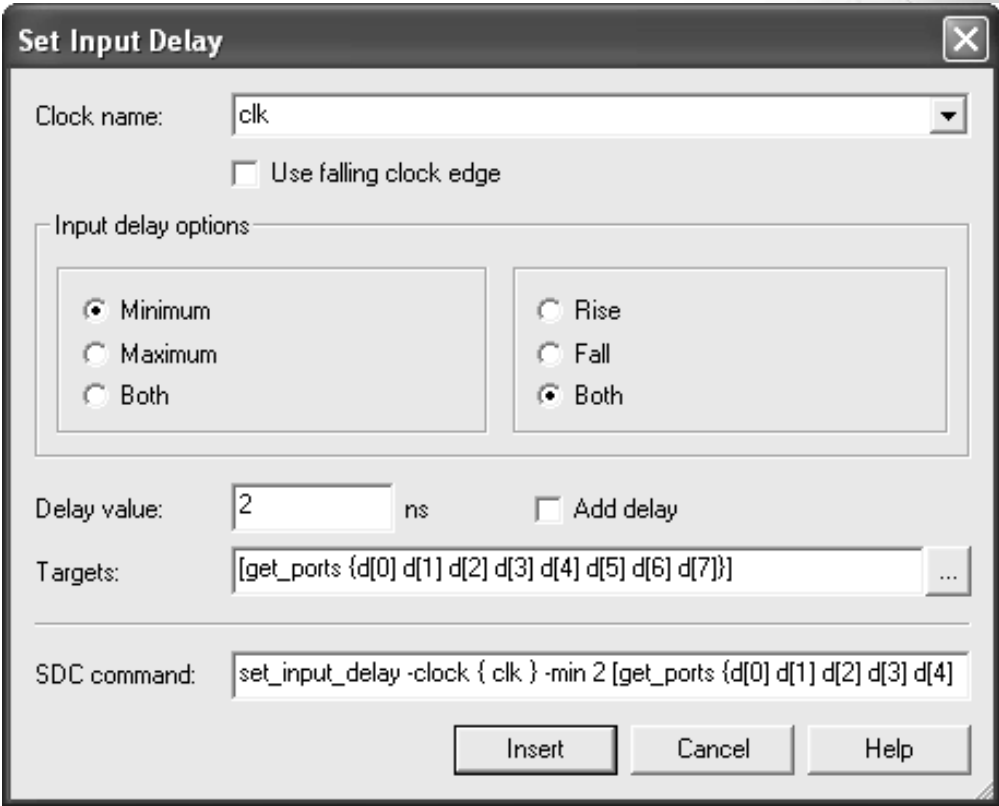
Synchronous I/O Constraining

- Specify system-level timing constraints
- Settings
 - Input/output maximum delay
 - **Maximum** amount of time a signal can take to arrive and still meet **setup** timing
 - Input/output minimum delay
 - **Minimum** amount of time a signal can remain active and still meet **hold** timing
- Pairs of max/min I/O delay constraints specify range of valid delay values for meeting FPGA and third-party device timing

Set Input/Output Delay

In SDC File Editor,
Edit menu ⇒ Insert Constraint

- Set Input/Output Delay fields:**
- Clock Name – Specify source clock
 - Input/Output delay options – Choose max or min constraint. Rise/Fall indicates if the constraint applies particularly to a rising or falling edge transition (advanced).
 - Delay value – Total off chip delay
 - Add delay – Must use if applying multiple sets of input/output delays to the same port (e.g. input ports feeding multiple internal registers)
 - Targets – Port to which setting is being applied



SDC: set_input_delay
SDC: set_output_delay



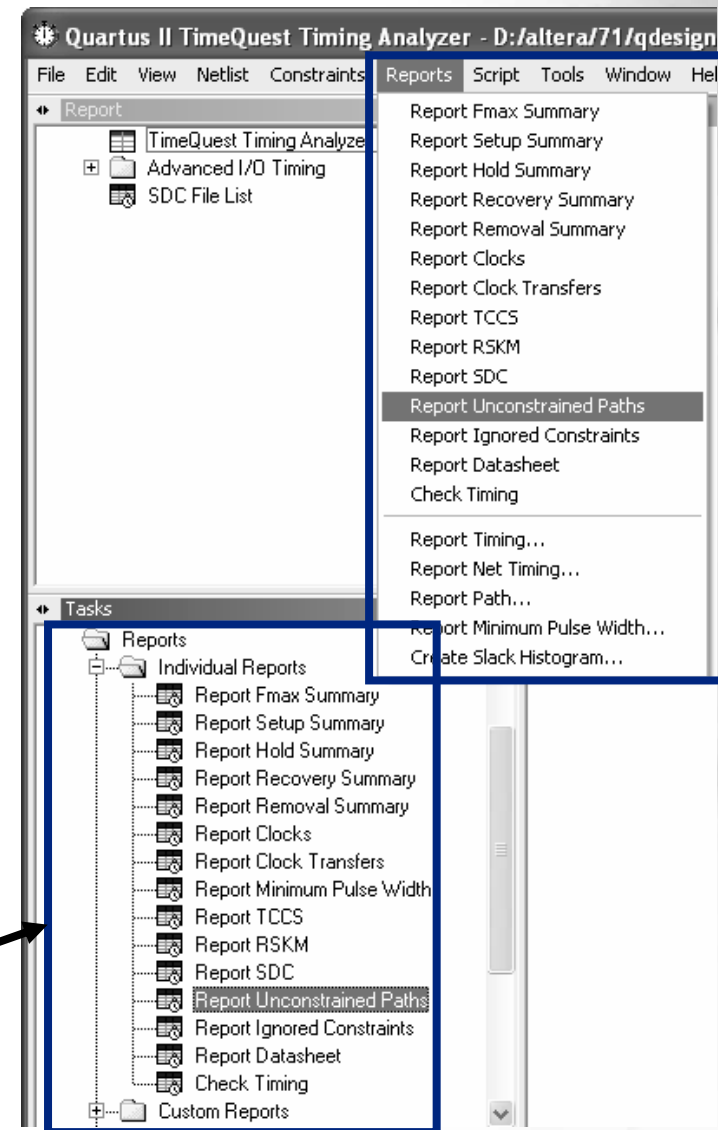
3) Update Timing Netlist

- Apply SDC constraints/exceptions to current timing netlist
- Generates warnings
 - Undefined clocks
 - Partially defined I/O delays
 - Combinatorial loops
- Update timing netlist after adding any new constraint
- Execution
 - **Update Timing Netlist** (Tasks pane or **Netlist** menu)

Tcl: update_timing_netlist

4) Generate Timing Reports

- Verify timing requirements and locate violations
- Check for fully constrained design or ignored timing constraints
- Two Methods
 - Tasks pane
 - Automatically creates/updates netlist & reads default SDC file if needed
 - **Reports** menu
 - Must have valid netlist to access



Double-click on individual report

Useful Reports for Design Constraining

- **Report Clocks**
 - Use to ensure all clocks have been defined correctly
- **Report Unconstrained Paths**
 - Use to determine if any constraints are missing
- **Report SDC**
 - Use to review what constraints have currently been applied to the netlist
- **Report Ignored Constraints**
 - Use to determine if any constraints being ignored due to possible typos or other errors in constraints

Generating Detailed Reports

Choose Report Timing (Reports menu) or double-click on Report Timing (Tasks pane)

Select level of detail

Select where to send output report

Tcl: report_timing

Report Timing

Clocks
 From clock: clk_cons
 To clock: clkx2_cons

Targets
 From: *
 Through:
 To: *mult*

Analysis type
 Setup Recovery
 Hold Removal

Paths
 Report number of paths: 32
 Maximum slack limit: ns

Output
 Detail level: Path Only
 Report panel name: Summary
 File name: Full Path

File options
 Overwrite Append

Tcl command: report_timing -from_clock clk_cons -to_clock clkx2_cons -from * -to

