Stratix HardCopy
Design Flow
With Quartus-II Ver. 3.0
Agenda

- HardCopy Stratix Overview
- Quartus II 3.0 Features For HardCopy
  - Compilation And Supported Devices
  - Design Assistant
  - HardCopy Optimization Wizard
  - HardCopy Timing Constraints
  - HardCopy Files Wizard
  - HardCopy Power Estimation
- Designing For HardCopy
  - Clock
  - Reset
  - Timing Closure
  - Non-synchronous Design Structure
HardCopy Stratix Overview
Stratix HardCopy Value Proposition

- Industry’s Only Complete Solution from Prototype to Production
  - Benefits of Designing with FPGAs
  - Seamless Migration from Proven FPGA Design to Custom Design
  - Unified & Complete Design Methodology with Single Design Tool

- Single Source for Devices, Tools & Intellectual Property (IP)

Seamless Migration
Simplified Technology
~70% Die Size Reduction
~1/5th ASIC Development Time
FPGA to HardCopy Device

- Remove Configuration Circuitry
- Remove Programmable Routing
- Remove Programmability for Logic & Memory
- Add Embedded Testability
- Customize with Two Metal Layers

FPGA Architecture with ASIC Routing

Stratix™ EP1S25

HardCopy HC1S25
HardCopy Silicon Technology

- Same Process Technology as FPGA
- Common Base Die
- Eight Metal Layers in HardCopy Stratix Devices
  - Two for Customer Design
Verification

- Timing Verification
  - Industry-Standard Tools
    - Synopsys PrimeTime Tool

- Structural Verification
  - Boundary Scan, BIST
  - Automatic Test Pattern Generation (ATPG)
    - No Need for Functional Vectors from Customer
    - Ensures High-Fault Coverage, ~99%

- Altera Delivers Tested Devices
Implementation Timeline

- Migration & Verification: ~2 Weeks
- Custom Masks, Prototype Fabrication, Assembly & Test: ~5-7 Weeks
- Prototype Approval: ~3 Weeks
- Production: ~8 Weeks

Highly Automated Migration Process Ensures Fast Turnaround & Minimal Customer Involvement

~18-20 Weeks to Volume Production
Migration Flow

**Customer**

- Provide SRAM Object File (*.sof)
- Provide Constraint Files (1)
- Review Results & Sign-Off

**Altera**

- Generate Netlist
- Check for Test & Fix
- Generate ATPG Vectors
- Place & Route (2)
- Verify Timing
- Fabricate Prototypes
- Assembly & Test
- Send Tested Prototypes to Customer

- Begin Volume Production

(1) *.sof, *.rpt, .pin, .esf, .psf, .vo, .sdo
  - *.apc – Placement Constraint Files Optional
(2) Altera Will Use *.apc if Provided & Only Route Design

2 Weeks

5-7 Weeks

Production Devices in 8 Weeks after Prototype Approval
HardCopy Device Performance

- Smaller Die
- Routing Benefits
- Reduced Internal Delays
- I/O Speed Unchanged
- $f_{\text{MAX}}$ Change Is Design-Dependent
- Performance Estimation through Quartus® II Software

On the Average, 50% Performance-Improvement over FPGA
Power Benefits

- ~40% Lower Power Consumption than FPGA
  - I/O Power Remains Unchanged
  - Power Estimation Tool Supported by Quartus II Design Software
    - Also Available on Altera Web Site
# HardCopy Vs. ASIC Devices

<table>
<thead>
<tr>
<th>Category</th>
<th>HardCopy</th>
<th>ASIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design Time</td>
<td>2-3 Weeks</td>
<td>Months</td>
</tr>
<tr>
<td>Design Effort</td>
<td>Minimal</td>
<td>Significant</td>
</tr>
<tr>
<td>Investment in Tools</td>
<td>None</td>
<td>Significant</td>
</tr>
<tr>
<td>Staff Needed</td>
<td>None</td>
<td>Significant</td>
</tr>
<tr>
<td>Package Design Effort</td>
<td>None</td>
<td>Yes</td>
</tr>
<tr>
<td>Board Re-Design</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Fabrication &amp; Assembly Cycle Time</td>
<td>7 Weeks</td>
<td>12 Weeks</td>
</tr>
<tr>
<td>Non-Recurring Engineering (NRE) Cost</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>Price per Part</td>
<td>Low</td>
<td>Lowest</td>
</tr>
<tr>
<td>Time to Volume</td>
<td>Weeks</td>
<td>Months</td>
</tr>
</tbody>
</table>

**HardCopy Process Leverages the Benefits of FPGA Design & Engineering**
HardCopy Vs. ASIC Conversion Flow

ASIC Conversion Generates New Netlist
Design Needs Validation & Adjustment

ASIC Conversion Flow

FPGA

Verilog / VHDL ➔ Synthesis ➔ ASIC

New Netlist

HardCopy Flow

Map Same Netlist

HardCopy
# HardCopy Migration Is Not ASIC Conversion

<table>
<thead>
<tr>
<th></th>
<th>Altera FPGA</th>
<th>HardCopy</th>
<th>ASIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logic Elements</td>
<td>Same as FPGA</td>
<td>Same as FPGA</td>
<td>Re-Synthesis to Gates</td>
</tr>
<tr>
<td>Memory Blocks</td>
<td>Same as FPGA</td>
<td>Same as FPGA</td>
<td>Compiled/ Cell Based</td>
</tr>
<tr>
<td>I/O Pins</td>
<td>Same as FPGA</td>
<td>Same as FPGA</td>
<td>Different I/O Library</td>
</tr>
<tr>
<td>Phase-Locked Loops (PLLs)</td>
<td>Same as FPGA</td>
<td>Same as FPGA</td>
<td>Different Design</td>
</tr>
<tr>
<td>Intellectual Property</td>
<td>Same as FPGA</td>
<td>Same as FPGA</td>
<td>Re-Qualification &amp; License</td>
</tr>
<tr>
<td>Packaging</td>
<td>Same as FPGA</td>
<td>Same as FPGA</td>
<td>Custom</td>
</tr>
<tr>
<td>Process Geometry</td>
<td>Same as FPGA</td>
<td>Same as FPGA</td>
<td>Same/Different/Hybrid</td>
</tr>
<tr>
<td>Foundry</td>
<td>Same as FPGA</td>
<td>Same as FPGA</td>
<td>Same/Different</td>
</tr>
<tr>
<td>Interconnect Routing</td>
<td>Similar SOG* of Standard Cell ASIC Routing</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* SOG – Sea of Gates

**ASIC Conversion Requires Starting Over**
Quartus II 3.0 Features
For HardCopy
HardCopy Process Flow

Start Quartus HardCopy flow

APEX

FPGA Family?

Stratix

Select Stratix hardcopy_fpga_prototype device

Compile (to the Stratix hardcopy_fpga_prototype device)

Run design assistant

Need to improve performance?

no

yes

pass?

Timing settings for external clock gitter

Run HardCopy Optimization Wizard

Close Quartus FPGA project

Open Quartus HardCopy optimization project

Compile (to a HardCopy Stratix device)

Placement info for actual HardCopy

Estimated Fmax of HardCopy Stratix device

Run HardCopy files wizard

.qar file for delivery to Altera

no

yes

20 YEARS of ALTEA INNOVATION
HARDCOPY_FPGA_PROTOTYPE Devices

- Part Of The Stratix Family
- Is A **Virtual** FPGA Device To Guide Quartus II
  - Has Reduced Feature Set From Its Equivalent FPGA
    - Less MRAMs As An Example
  - Has An Identical Timing Model As The FPGA
  - Has The Identical Floorplan Of The FPGA
- Is Not The Actual Hardcopy Device
  - Does Not Match The Floorplan Of The Hardcopy Device
  - Matches The Pin-out Of The Equivalent Hardcopy Device
- Always Has Equivalent Real FPGA
  - This Is NOT A New Hardware Die
HARDCOPY_FPGA_PROTOTYPE vs. FPGA vs. HardCopy devices

EP1S30F780C5

Blanked out MRAM blocks

EP1S30F780C5_HARDCOPY_FPGA_PROTOTYPE

HC1S30F780

20 YEARS of ALTERA INNOVATION
<table>
<thead>
<tr>
<th>Physical FPGA</th>
<th>FPGA Prototype Device for HardCopy</th>
<th>HardCopy Device</th>
</tr>
</thead>
<tbody>
<tr>
<td>EP1S25F672C6</td>
<td>EP1S25F672C6_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S25F672</td>
</tr>
<tr>
<td>EP1S25F672C7</td>
<td>EP1S25F672C7_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S25F672</td>
</tr>
<tr>
<td>EP1S30F780C5</td>
<td>EP1S30F780C5_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S30F780</td>
</tr>
<tr>
<td>EP1S30F780C6</td>
<td>EP1S30F780C6_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S30F780</td>
</tr>
<tr>
<td>EP1S30F780C7</td>
<td>EP1S30F780C7_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S30F780</td>
</tr>
<tr>
<td>EP1S40F780C5</td>
<td>EP1S40F780C5_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S40F780</td>
</tr>
<tr>
<td>EP1S40F780C6</td>
<td>EP1S40F780C6_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S40F780</td>
</tr>
<tr>
<td>EP1S40F780C7</td>
<td>EP1S40F780C7_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S40F780</td>
</tr>
<tr>
<td>EP1S60F1020C6</td>
<td>EP1S60F1020C6_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S60F1020</td>
</tr>
<tr>
<td>EP1S60F1020C7</td>
<td>EP1S60F1020C7_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S60F1020</td>
</tr>
<tr>
<td>EP1S80F1020C6</td>
<td>EP1S80F1020C6_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S80F1020</td>
</tr>
<tr>
<td>EP1S80F1020C7</td>
<td>EP1S80F1020C7_HARDCOPY_FPGA_PROTOTYPE</td>
<td>HC1S80F1020</td>
</tr>
</tbody>
</table>
## Stratix Supported Devices

### Table 1: Hardcopy Stratix Devices Vs. Equivalent Stratix Devices

<table>
<thead>
<tr>
<th>Device</th>
<th>LEs</th>
<th>M512 Blocks</th>
<th>M4K Blocks</th>
<th>M-RAM Blocks</th>
<th>DSP Blocks</th>
<th>PLLS</th>
<th>Max. User I/Os</th>
</tr>
</thead>
<tbody>
<tr>
<td>HC1S25F672</td>
<td>25,660</td>
<td>224</td>
<td>138</td>
<td>2</td>
<td>10</td>
<td>6</td>
<td>473</td>
</tr>
<tr>
<td>EP1S25F672</td>
<td>25,660</td>
<td>224</td>
<td>138</td>
<td>2</td>
<td>10</td>
<td>6</td>
<td>473</td>
</tr>
<tr>
<td>HC1S30F780</td>
<td>32,470</td>
<td>295</td>
<td>171</td>
<td>2</td>
<td>12</td>
<td>10</td>
<td>597</td>
</tr>
<tr>
<td>EP1S30F780</td>
<td>32,470</td>
<td>295</td>
<td>171</td>
<td>2</td>
<td>12</td>
<td>10</td>
<td>597</td>
</tr>
<tr>
<td>HC1S40F780</td>
<td>41,250</td>
<td>384</td>
<td>183</td>
<td>2</td>
<td>14</td>
<td>12</td>
<td>615</td>
</tr>
<tr>
<td>EP1S40F780</td>
<td>41,250</td>
<td>384</td>
<td>183</td>
<td>2</td>
<td>14</td>
<td>12</td>
<td>615</td>
</tr>
<tr>
<td>HC1S60F1020</td>
<td>57,120</td>
<td>574</td>
<td>292</td>
<td>6</td>
<td>18</td>
<td>12</td>
<td>773</td>
</tr>
<tr>
<td>EP1S60F1020</td>
<td>57,120</td>
<td>574</td>
<td>292</td>
<td>6</td>
<td>18</td>
<td>12</td>
<td>773</td>
</tr>
<tr>
<td>HC1S80F1020</td>
<td>79,040</td>
<td>767</td>
<td>364</td>
<td>6</td>
<td>22</td>
<td>12</td>
<td>773</td>
</tr>
<tr>
<td>EP1S80F1020</td>
<td>79,040</td>
<td>767</td>
<td>364</td>
<td>6</td>
<td>22</td>
<td>12</td>
<td>773</td>
</tr>
</tbody>
</table>
Compilation

- In An Open Project, Select The Target HardCopy Device Thru The Compiler Setting Option. Make Sure To Select a HARDCOPY_FPGA_PROTOTYPE device
- Assignments -> Device Menu Option Will Bring You To The Settings Window
- Compile Your Design
Design Assistant

- Design Assistant Checks Your Design Against Design Rules That Are Selected Thru The Assignment Menu. Turn It On During Compilation.
Rules Are In Place To Guarantee Smooth Migration From FPGA To HardCopy Device
Checks The Viability Of The Design And Assesses Its Risk
Need To Make Sure All Violations Are Reviewed Informational Messages Are Acceptable
HardCopy Optimization Wizard

- HardCopy Will Always Perform As Good As FPGA. You May Not Need To Run Optimization Wizard Unless You Don’t Meet Your Performance In FPGA.

- The Optimization Wizard Is Used To Generate HardCopy Files That Will Approximate The Real Delays In The HardCopy Device.

- The optimization wizard creates a new project directory, a .VSM file, and back-annotates the pin information from the original project.
HardCopy Optimization Wizard (continue)

- All Pin Assignments Are Preserved.
- All Pin Properties Are Preserved
  - I/O Standard, Drive Strength, ...
- Global Assignments Are Maintained
- All Core Location Assignments Are Removed.
- All Logic Lock Regions Are Removed.
- All Timing Assignments Are Migrated
  - Including Multi-cycle, Point To Point Cuts, ...
HardCopy Optimization Wizard (Menu)

- Project Assignments Processing Tools Wizard
  - Add Current File to Project
  - Add/Remove Files in Project...

- Archive Project...
- Restore Archived Project...

- Generate Tcl File for Project...

- HardCopy Optimization Wizard...
  - HardCopy Files Wizard...
  - HardCopy Power Estimation

- Locate in Timing Closure Floorplan
- Locate in Last Compilation Floorplan
- Locate in Chip Editor
- Locate in Design File

- Set Compiler Focus to Current Entity Ctrl+J

- Open Current Focus Entity
- Open Selected Entity
- Open Parent Entity
- Open Top-Level File in Hierarchy

Quartus II

Directory c:\documents and settings\john\desktop\vhc training\vhc_hardcopy_optimization does not exist. Do you want to create it?

- Yes
- No
New Project

- We now have a new Quartus II Project in a new directory
  - Example: <my design>_hardcopy_optimization
  - Assigned to the equivalent Hardcopy member in the “Stratix Hardcopy” family.

- Old Quartus II project for FPGA_PROTOTYPE compile maintained

- Compiling the new project gives you the hardcopy timing estimation.

- Can change timing assignments
  - To optimize Hardcopy designs differently
  - To monitor timing differently
HardCopy Timing Constraints

- In the New HardCopy Project Directory, open the project, specify HardCopy Timing Constraint, compile the design, and look up the Timing Analyzer results for the new FMAX.
HardCopy Files Wizard (formerly passport)

- The Files Wizard Could Be Run With Or Without An Optimization Run
- Runs Design Assistant With All Rules Enabled Regardless Of What The User Turned On/Off
- The Files Wizard Goes Thru An Interactive List Of Questions To Gather Design Specifications
- The Files Wizard Generates A .Qar File That Gets Sent To Altera For Conversion
Signal Tap Is Fully Supported In HardCopy Stratix, But Not In HardCopy APEX
HardCopy Files Wizard (Continue)
Details Of The Files archived in the .qar file are Listed In Chapter 14 of the HardCopy Device Handbook
HardCopy Power Estimation

- In the HardCopy Project Directory, invoke the Power Calculator as shown thru the menu.
- Upon running the Power Estimation Command, Quartus II automatically sends information and configures the HardCopy Power Calculator on Altera’s Web site.
- Some information will have to be entered manually. Most information can be changed manually.
Designing For HardCopy (Back-Up Slides)
Designing For HardCopy

- Guidelines To Permit A Successful FPGA To HardCopy Device Conversion
- Good General Design Practice Guidelines
- Design Assistant Examines The Conformance Of Your Design Against These Design Rules
- Take Corrective Design Actions Based On Feedback From Design Assistant
- Case study
HardCopy Recommendation 1

Clocks

- All Clock Signals In A Design Should Be Global Signals. Clock Signals That Are Mapped To Regular Logic Can Affect The Performance Of The Design (i.e. Might Not Work)

Can DO

- More Difficult STA

Better To Do

- Easier STA
- Two Extra Pins
HardCopy Recommendation 1

Do

Any Time A Circuit Needs To Use Both Edges Of The Clock, The Duty Cycle Has To Be Accurately Described For Proper Static Timing Analysis

Try To Use Same Edge Clocking To Avoid Warnings By The Design Assistant

Dont

50% duty cycle clock

Duty Cycle Will Determine Success In Meeting Timing Not Frequency (Probably Not The Intention)
HardCopy Recommendation 2

Reset Trees Have Inherent Delays In Them That Might Cause A Glitch On The Output Of Registers. If This Presents A Problem In Your Design, Make Reset Part Of Your Input Logic.

Glitch Free Reset

Always @ (Posedge Clk) Begin
  If (!Rst)
    Q <= 1'b0;
  Else
    Q <= D;
End
HardCopy Recommendation 3  Timing Closure

- Minimize Excessive Number Of Loads On Nets. This Will Improve Your Chances Of Meeting Your Design Goals
- If The Two Registers Are Triggered By Clock Edges At The Same Time, A Hold Time Violation May Occur. This Is Only A Design Assistant Info Message
HardCopy Recommendation 4
Non-synchronous Design Structure

- A Design Should Not Contain Any Combinatorial Loops
  These Combinatorial Loops Can Cause Significant Stability And Reliability Problems In A Design

- A Design Should Not Contain Any Combinatorial Loops Where The Output Of A Register Directly Drives One Of Its Own Control Signals

Glitch here due to feedback path from QN to RN

Combinatorial feedback path