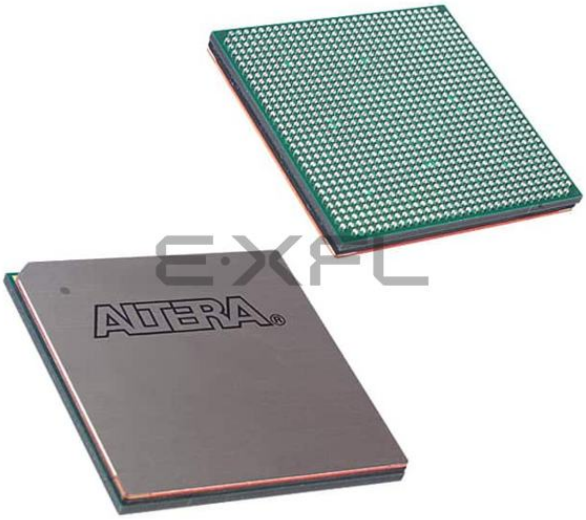


Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

Embedded - FPGAs, or Field Programmable Gate Arrays, are advanced integrated circuits that offer unparalleled flexibility and performance for digital systems. Unlike traditional fixed-function logic devices, FPGAs can be programmed and reprogrammed to execute a wide array of logical operations, enabling customized functionality tailored to specific applications. This reprogrammability allows developers to iterate designs quickly and implement complex functions without the need for custom hardware.

Applications of Embedded - FPGAs

The versatility of Embedded - FPGAs makes them indispensable in numerous fields. In telecommunications.

Details	
Product Status	Obsolete
Number of LABs/CLBs	7904
Number of Logic Elements/Cells	79040
Total RAM Bits	5658048
Number of I/O	782
Number of Gates	-
Voltage - Supply	1.425V ~ 1.575V
Mounting Type	Surface Mount
Operating Temperature	-
Package / Case	1020-BBGA
Supplier Device Package	1020-FBGA (33x33)
Purchase URL	https://www.e-xfl.com/product-detail/intel/hc1s80f1020ay

Table 1–1. HardCopy Stratix Devices and Features

Device	LEs (1)	M512 Blocks	M4K Blocks	M-RAM Blocks	DSP Blocks (2)	PLLs (3)
HC1S25	25,660	224	138	2	10	6
HC1S30	32,470	295	171	2 (4)	12	6
HC1S40	41,250	384	183	2 (4)	14	6
HC1S60	57,120	574	292	6	18	12
HC1S80	79,040	767	364	6 (4)	22	12

Notes to Table 1–1:
(1) LE: logic elements.
(2) DSP: digital signal processing.
(3) PLLs: phase-locked loops.
(4) In HC1S30, HC1S40, and HC1S80 devices, there are fewer M-RAM blocks than in the equivalent Stratix FPGA. All other resources are identical to the Stratix counterpart.

Features

HardCopy Stratix devices are manufactured on the same 1.5-V, 0.13 µm all-layer-copper metal fabrication process (up to eight layers of metal) as the Stratix FPGAs.

- Preserves the functionality of a configured Stratix device
- Pin-compatible with the Stratix counterparts
- On average, 50% faster than their Stratix equivalents
- On average, 40% less power consumption than their Stratix equivalents
- 25,660 to 79,040 LEs
- Up to 5,658,408 RAM bits available
- TriMatrix memory architecture consisting of three RAM block sizes to implement true dual-port memory and first-in-first-out (FIFO) buffers
- Embedded high-speed DSP blocks provide dedicated implementation of multipliers, multiply-accumulate functions, and finite impulse response (FIR) filters
- Up to 12 PLLs (four enhanced PLLs and eight fast PLLs) per device which provide identical features as the FPGA counterparts, including spread spectrum, programmable bandwidth, clock switchover, real-time PLL reconfiguration, advanced multiplication, and phase shifting
- Supports numerous single-ended and differential I/O standards
- Supports high-speed networking and communications bus standards including RapidIO™, UTOPIA IV, CSIX, HyperTransport technology, 10G Ethernet XSBI, SPI-4 Phase 2 (POS-PHY Level 4), and SFI-4
- Differential on-chip termination support for LVDS

The HardCopy Stratix family consists of base arrays that are common to all designs for a particular device density. Design-specific customization is done within the top two metal layers. The base arrays use an area-efficient sea-of-logic-elements (SOLE) core and extend the flexibility of high-density Stratix FPGAs to a cost-effective, high-volume production solution. With a seamless migration process employed in numerous successful designs, functionality-verified Stratix FPGA designs can be migrated to fixed-function HardCopy Stratix devices with minimal risk and guaranteed first-time success.

The SRAM configuration cells of the original Stratix devices are replaced in HardCopy Stratix devices by metal connects, which define the function of each logic element (LE), digital signal processing (DSP) block, phase-locked loop (PLL), embedded memory, and I/O cell in the device. These resources are interconnected using metallization layers. Once a HardCopy Stratix device has been manufactured, the functionality of the device is fixed and no re-programming is possible. However, as is the case with Stratix FPGAs, the PLLs can be dynamically configured in HardCopy Stratix devices.

HardCopy Stratix and Stratix FPGA Differences

To ensure HardCopy Stratix device functionality and performance, designers should thoroughly test the original Stratix FPGA-based design for satisfactory results before committing the design for migration to a HardCopy Stratix device. Unlike Stratix FPGAs, HardCopy Stratix devices are customized at the time of manufacturing and therefore do not have programmability support.

Since HardCopy Stratix devices are customized within the top two metal layers, no configuration circuitry is required. Refer to "Power-Up Modes in HardCopy Stratix Devices" on page 2–7 for more information.

Depending on the design, HardCopy Stratix devices can provide, on average, a 50% performance improvement over equivalent Stratix FPGAs. The performance improvement is achieved by die size reduction, metal interconnect optimization, and customized signal buffering. HardCopy Stratix devices consume, on average, 40% less power than their equivalent Stratix FPGAs.

☞ Designers can use the Quartus II software to design HardCopy Stratix devices, estimate performance and power consumption, and maximize system throughput.

Table 2–1. HardCopy Stratix and Stratix Device Comparison (Part 2 of 2)	
HardCopy Stratix	**Stratix**
The I/O drive strength for single-ended I/O pins are slightly different and is modeled in the HardCopy Stratix IBIS models.	The I/O drive strength for single-ended I/O pins are found in Stratix IBIS models.
In the HC1S40 780-pin FineLine BGA® device, the I/O pins U12 and U18 must be connected to ground.	In the HC1S40 780-pin FineLine BGA device, the I/O pins U12 and U18 are available as general-purpose I/O pins.
The BSDL file describes re-ordered Joint Test Action Group (JTAG) boundary-scan chains.	The JTAG boundary-scan chain is defined in the BSDL file.

Note to Table 2–1:

(1) Performance and power consumption are design dependant.

Logic Elements

Logic is implemented in HardCopy Stratix devices using the same architectural units as the Stratix device family. The basic unit is the logic element (LE) with logic array blocks (LAB) consisting of 10 LEs. The implementation of LEs and LABs is identical to the Stratix device family.

In the HardCopy Stratix device family, all extraneous routing resources not essential to the specific design are removed for performance and die size efficiency. Therefore, the MultiTrack interconnect for routing implementation between LABs and other device resources in the Stratix device family is no longer necessary in the HardCopy Stratix device family.

Table 2–2 illustrates the differences between HardCopy Stratix and Stratix logic.

Table 2–2. HardCopy Stratix and Stratix Logic Comparison	
HardCopy Stratix	**Stratix**
All routing connections are direct and all unused routing is removed.	MultiTrack routing stitches routing resources together to provide a path.

Embedded Memory

TriMatrix™ memory blocks from Stratix devices, including M512, M4K, and M-RAM memory blocks, are available in HardCopy Stratix devices. Embedded memory is seamlessly implemented in the equivalent resource.

When designing with very tight timing constraints (for example, DDR or quad data rate [QDR]), or if using the programmable drive strength option, Altera recommends verifying final drive strength using updated IBIS models located on the Altera website at **www.altera.com**. Differential I/O standards are unaffected.

I/O pin placement and `VREF` pin placement rules are identical between HardCopy Stratix and Stratix devices. Unused pin settings will carry over from Stratix device settings and are implemented as tri-stated outputs driving ground or outputs driving V_{CC}.

In Stratix EP1S40 780-pin FineLine BGA FPGAs, the I/O pins `U12` and `U18` are available as general-purpose I/O pins. In the FPGA prototype, EP1S40F780_HARDCOPY_FPGA_PROTOTYPE, and in the Hardcopy Stratix HC1S40 780-pin FineLine BGA device, the I/O pins `U12` and `U18` must be connected to ground. HC1S40 780-pin FineLine BGA and EP1S40F780_HARDCOPY_FPGA_PROTOTYPE pin-outs are identical.

Table 2–7 illustrates the differences between HardCopy Stratix and Stratix I/O pins.

Table 2–7. HardCopy Stratix and Stratix I/O Pin Comparison

HardCopy Stratix	Stratix
The IOEs are equivalent, but not identical to, the FPGA IOEs due to slight design optimizations for HardCopy devices.	IOEs are optimized for the FPGA architecture.
The I/O drive strength for single-ended I/O pins are slightly different and are found in the HardCopy Stratix IBIS models.	The I/O drive strength for single-ended I/O pins are found in Stratix IBIS models.
In the HC1S40 780-pin FineLine BGA device, the I/O pins `U12` and `U18` must be connected to ground.	In the EP1S40 780-pin FineLine BGA device, the I/O pins `U12` and `U18` are available as general-purpose I/O pins.

Power-Up Modes in HardCopy Stratix Devices

Designers do not need to configure HardCopy Stratix devices, unlike their FPGA counterparts. However, to facilitate seamless migration, configuration can be emulated in HardCopy Stratix devices.

The modes in which a HardCopy Stratix device can be made ready for operation after power-up are: instant on, instant on after 50 ms, and configuration emulation. These modes are briefly described below.

Table 3–3. 32-Bit HardCopy Stratix Device IDCODE

Device	IDCODE (32 Bits) *(1)*			
	Version (4 Bits)	Part Number (16 Bits)	Manufacturer Identity (11 Bits)	LSB (1 Bit) *(2)*
HC1S25	0000	0010 0000 0000 0011	000 0110 1110	1
HC1S30	0000	0010 0000 0000 0100	000 0110 1110	1
HC1S40	0000	0010 0000 0000 0101	000 0110 1110	1
HC1S60	0000	0010 0000 0000 0110	000 0110 1110	1
HC1S80	0000	0010 0000 0000 0111	000 0110 1110	1

Notes to Table 3–3:

(1) The most significant bit (MSB) is on the left.

(2) The `IDCODE`'s least significant bit (LSB) is always 1.

Figure 3–1 shows the timing requirements for the JTAG signals.

Figure 3–1. HardCopy Stratix JTAG Waveforms

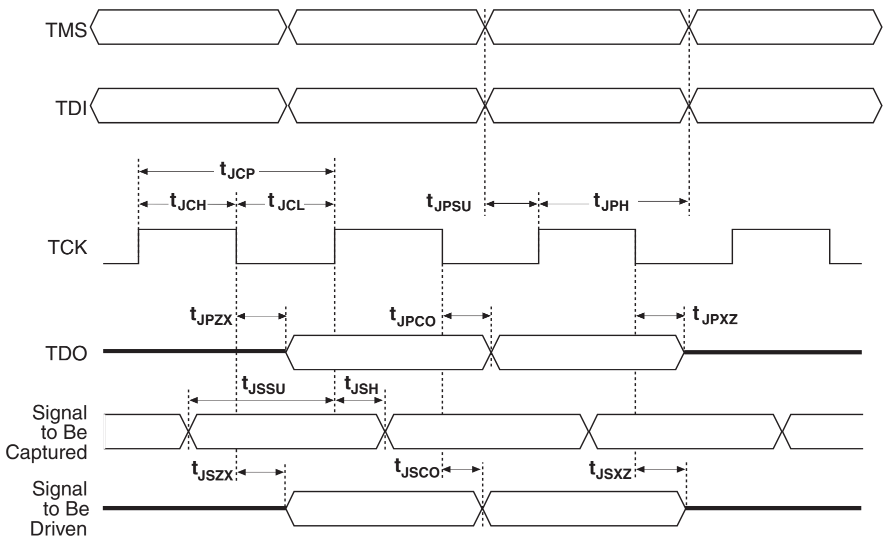


Table 4–10. 3.3-V PCML Specifications

Symbol	Parameter	Conditions	Minimum	Typical	Maximum	Unit
V_{CCIO}	I/O supply voltage		3.135	3.3	3.465	V
V_{ID}	Input differential voltage swing		300		600	mV
V_{ICM}	Input common mode voltage		1.5		3.465	V
V_{OD}	Output differential voltage		300	370	500	mV
ΔV_{OD}	Change in V_{OD} between high and low				50	mV
V_{OCM}	Output common mode voltage		2.5	2.85	3.3	V
ΔV_{OCM}	Change in V_{OCM} between high and low				50	mV
V_T	Output termination voltage			V_{CCIO}		V
R_1	Output external pull-up resistors		45	50	55	Ω
R_2	Output external pull-up resistors		45	50	55	Ω

Table 4–11. LVPECL Specifications

Symbol	Parameter	Conditions	Minimum	Typical	Maximum	Unit
V_{CCIO}	I/O supply voltage		3.135	3.3	3.465	V
V_{ID}	Input differential voltage swing		300		1,000	mV
V_{ICM}	Input common mode voltage		1		2	V
V_{OD}	Output differential voltage	$R_L = 100\ \Omega$	525	700	970	mV
V_{OCM}	Output common mode voltage	$R_L = 100\ \Omega$	1.5	1.7	1.9	V
R_L	Receiver differential input resistor		90	100	110	Ω

Table 4–29. 1.5-V Differential HSTL Specifications						
Symbol	**Parameter**	**Conditions**	**Minimum**	**Typical**	**Maximum**	**Unit**
V_{CCIO}	I/O supply voltage		1.4	1.5	1.6	V
V_{DIF} (DC)	DC input differential voltage		0.2			V
V_{CM} (DC)	DC common mode input voltage		0.68		0.9	V
V_{DIF} (AC)	AC differential input voltage		0.4			V

Table 4–30. CTT I/O Specifications						
Symbol	**Parameter**	**Conditions**	**Minimum**	**Typical**	**Maximum**	**Unit**
V_{CCIO}	Output supply voltage		2.05	3.3	3.6	V
V_{TT}/V_{REF}	Termination and input reference voltage		1.35	1.5	1.65	V
V_{IH}	High-level input voltage		$V_{REF} + 0.2$			V
V_{IL}	Low-level input voltage				$V_{REF} - 0.2$	V
V_{OH}	High-level output voltage	$I_{OH} = -8$ mA	$V_{REF} + 0.4$			V
V_{OL}	Low-level output voltage	$I_{OL} = 8$ mA			$V_{REF} - 0.4$	V
I_O	Output leakage current (when output is high *Z*)	$GND \leq V_{OUT} \leq V_{CCIO}$	–10		10	µA

Table 4–31. Bus Hold Parameters										
Parameter	**Conditions**	**V_{CCIO} Level**								**Unit**
		1.5 V		**1.8 V**		**2.5 V**		**3.3 V**		
		Min	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	
Low sustaining current	$V_{IN} > V_{IL}$ (maximum)	25		30		50		70		µA
High sustaining current	$V_{IN} < V_{IH}$ (minimum)	–25		–30		–50		–70		µA
Low overdrive current	$0 V < V_{IN} < V_{CCIO}$		160		200		300		500	µA
High overdrive current	$0 V < V_{IN} < V_{CCIO}$		–160		–200		–300		–500	µA
Bus hold trip point		0.5	1.0	0.68	1.07	0.7	1.7	0.8	2.0	V

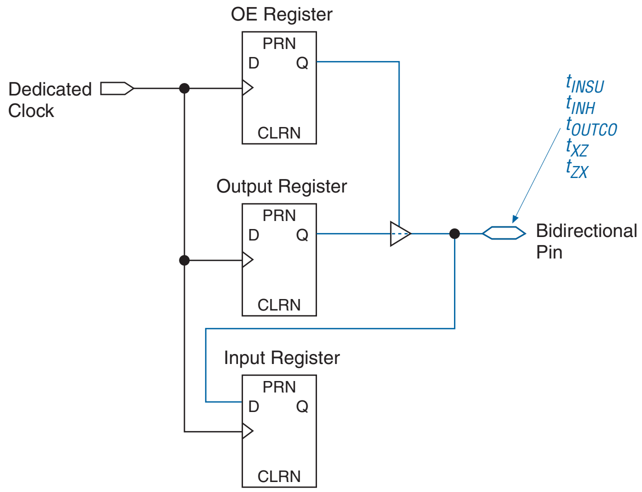
The final timing numbers and actual performance for each HardCopy Stratix design is available when the design migration is complete and are subject to verification and approval by Altera and the designer during the HardCopy Design review process.

For more information, refer to the *HardCopy Series Back-End Timing Closure* chapter in the *HardCopy Series Handbook*.

External Timing Parameters

External timing parameters are specified by device density and speed grade. Figure 4–1 shows the pin-to-pin timing model for bidirectional IOE pin timing. All registers are within the IOE.

Figure 4–1. External Timing in HardCopy Stratix Devices



All external timing parameters reported in this section are defined with respect to the dedicated clock pin as the starting point. All external I/O timing parameters shown are for 3.3-V LVTTL I/O standard with the 4-mA current strength and fast slew rate. For external I/O timing using standards other than LVTTL or for different current strengths, use the I/O standard input and output delay adders in the *Stratix Device Handbook*.

Table 4–50 shows the high-speed I/O timing for HardCopy Stratix devices.

Table 4–50. High-Speed I/O Specifications (Part 1 of 2) Notes (1), (2)

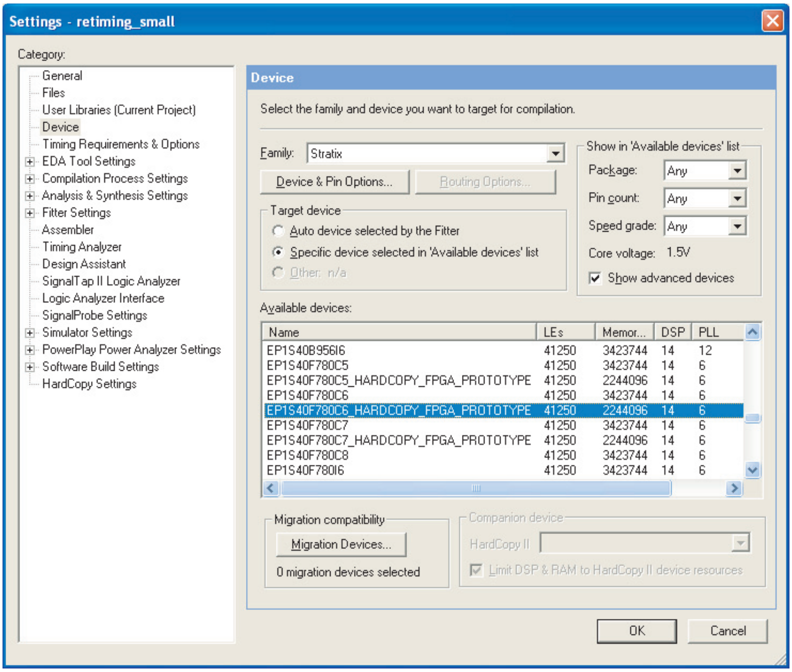
Symbol	Conditions	Performance			Unit
		Min	Typ	Max	
f_{HSCLK} (Clock frequency) (LVDS, LVPECL, HyperTransport technology) $f_{HSCLK} = f_{HSDR} / W$	*W* = 4 to 30 (Serdes used)	10		210	MHz
	W = 2 (Serdes bypass)	50		231	MHz
	W = 2 (Serdes used)	150		420	MHz
	W = 1 (Serdes bypass)	100		462	MHz
	W = 1 (Serdes used)	300		717	MHz
f_{HSDR} Device operation (LVDS, LVPECL, HyperTransport technology)	*J* = 10	300		840	Mbps
	J = 8	300		840	Mbps
	J = 7	300		840	Mbps
	J = 4	300		840	Mbps
	J = 2	100		462	Mbps
	J = 1 (LVDS and LVPECL only)	100		462	Mbps
f_{HSCLK} (Clock frequency) (PCML) $f_{HSCLK} = f_{HSDR} / W$	*W* = 4 to 30 (Serdes used)	10		100	MHz
	W = 2 (Serdes bypass)	50		200	MHz
	W = 2 (Serdes used)	150		200	MHz
	W = 1 (Serdes bypass)	100		250	MHz
	W = 1 (Serdes used)	300		400	MHz
f_{HSDR} Device operation (PCML)	*J* = 10	300		400	Mbps
	J = 8	300		400	Mbps
	J = 7	300		400	Mbps
	J = 4	300		400	Mbps
	J = 2	100		400	Mbps
	J = 1	100		250	Mbps
TCCS	All			200	ps
SW	PCML (*J* = 4, 7, 8, 10)	750			ps
	PCML (*J* = 2)	900			ps
	PCML (*J* = 1)	1,500			ps
	LVDS and LVPECL (*J* = 1)	500			ps
	LVDS, LVPECL, HyperTransport technology (*J* = 2 through 10)	440			ps

Table 4–51. Enhanced PLL Specifications (Part 3 of 3)					
Symbol	**Parameter**	**Min**	**Typ**	**Max**	**Unit**
t_{ARESET}	Minimum pulse width on ARESET signal	10 *(11)*			ns
		500 *(12)*			ns

Notes to Table 4–51:

(1) The minimum input clock frequency to the PFD (f_{IN}/N) must be at least 3 MHz for HardCopy Stratix device enhanced PLLs.
(2) Refer to "Maximum Input and Output Clock Rates".
(3) t_{FCOMP} can also equal 50% of the input clock period multiplied by the pre-scale divider *n* (whichever is less).
(4) This parameter is timing analyzed by the Quartus II software because the `scanclk` and `scandata` ports can be driven by the logic array.
(5) Actual jitter performance may vary based on the system configuration.
(6) Total required time to reconfigure and lock is equal to $t_{DLOCK} + t_{CONFIG}$. If only post-scale counters and delays are changed, then t_{DLOCK} is equal to 0.
(7) The VCO range is limited to 500 to 800 MHz when the spread spectrum feature is selected.
(8) Lock time is a function of PLL configuration and may be significantly faster depending on bandwidth settings or feedback counter change increment.
(9) Exact, user-controllable value depends on the PLL settings.
(10) The LOCK circuit on HardCopy Stratix PLLs does not work for industrial devices below –20°C unless the PFD frequency > 200 MHz. Refer to the *Stratix FPGA Errata Sheet* for more information on the PLL.
(11) Applicable when the PLL input clock has been running continuously for at least 10 µs.
(12) Applicable when the PLL input clock has stopped toggling or has been running continuously for less than 10 µs.

Figure 5–2. Selecting a HARDCOPY_FPGA_PROTOTYPE Device



By choosing the HARDCOPY_FPGA_PROTOTYPE device, all the design information, available resources, package option, and pin assignments are constrained to guarantee a seamless migration of your project to the HardCopy Stratix device. The netlist resulting from the HARDCOPY_FPGA_PROTOTYPE device compilation contains information about the electrical connectivity, resources used, I/O placements, and the unused resources in the FPGA device.

4. On the Assignments menu, click **Settings**. In the **Category** list, select **HardCopy Settings** and specify the input transition timing to be modeled for both clock and data input pins. These transition times are used in static timing analysis during back-end timing closure of the HardCopy device.

5. Add constraints to your HARDCOPY_FPGA_PROTOTYPE device, and on the Processing menu, click **Start Compilation** to compile the design.

☞ Performance estimation is not supported for HardCopy APEX devices in the Quartus II software. Your design can be optimized by modifying the RTL code or the FPGA design and the constraints. You should contact Altera to discuss any desired performance improvements with HardCopy APEX devices.

Buffer Insertion

Beginning with version 4.2, the Quartus II software provides improved HardCopy Stratix device timing closure and estimation, to more accurately reflect the results expected after back-end migration. The Quartus II software performs the necessary buffer insertion in your HardCopy Stratix device during the Fitter process, and stores the location of these buffers and necessary routing information in the Quartus II Archive File. This buffer insertion improves the estimation of the Quartus II Timing Analyzer for the HardCopy Stratix device.

Placement Constraints

Beginning with version 4.2, the Quartus II software supports placement constraints and LogicLock regions for HardCopy Stratix devices. Figure 5–6 shows an iterative process to modify the placement constraints until the best placement for the HardCopy Stratix device is achieved.

Altera recommends physical synthesis optimizations for the HARDCOPY_FPGA_PROTOTYPE. The work done in the prototype enhances performance in the HardCopy Stratix device after migration. Duplicating combinational logic and registers can increase area utilization, which limits placement flexibility when designs exceed 95% logic element (LE) utilization. However, duplicating combinational logic and registers can help with performance by allowing critical paths to be duplicated when their endpoints must reach different areas of the device floorplan.

For more information on netlist and design optimization, refer to *Area Optimization and Timing Closure* in volume 2 of the *Quartus II Development Software Handbook*.

Using LogicLock Regions in HardCopy Stratix Designs

Create LogicLock regions in the HARDCOPY_FPGA_PROTOTYPE project and migrate the regions into the HardCopy Stratix optimization project using the Quartus II software. LogicLock regions can provide significant benefits in design performance by carefully isolating critical blocks of logic, including:

- MegaCore® IP functions
- I/O interfaces
- Reset or other critical logic feeding global clock lines
- Partitioned function blocks

You must compile your design initially without LogicLock regions present and review the timing analysis reports to determine if additional constraints or LogicLock regions are necessary. This process allows you to determine which function blocks or data paths require LogicLock regions.

Create LogicLock regions in the HARDCOPY_FPGA_PROTOTYPE design project in the Quartus II software. This transfers the LogicLock regions to the HardCopy design project after the HardCopy Timing Optimization Wizard is run. Although the Quartus II software transfers the contents of the LogicLock region, the area, location, and soft boundary settings revert to their default settings in the HardCopy project immediately after the HardCopy Timing Optimization Wizard is run.

If you are using LogicLock regions, Altera recommends you use the **Migration Only** setting in the HardCopy Timing Optimization Wizard to create the HardCopy design project. You should not compile your design automatically using the **Full Compilation** or **Migrate and Compile** options in the wizard. Open the HardCopy design project and verify that the LogicLock region properties meet your desired settings before compiling the HardCopy optimization project. LogicLock soft regions are turned on by default in the HardCopy Stratix design. While this does allow the Fitter to place all logic in your design with fewer restrictions, it is not optimal for performance improvement in the HardCopy Stratix design.

Recommended LogicLock Settings for HardCopy Stratix Designs

Altera recommends the following LogicLock region settings for the HARDCOPY_FPGA_PROTOTYPE:

- Turn on **Reserve Unused Logic**
- Turn off **Soft Region**
- Select either **Auto** or **Fixed** as the **Size** (design-dependent)
- Select either **Floating** or **Locked** as the **Location** (design-dependent)

When using the **Reserve Unused Logic** setting in a design with high resource utilization (> 95% LE utilization), and a large number of LogicLock regions, the design may not fit in the device. Turning off **Reserve Unused Logic** in less critical LogicLock regions can help Fitter placement. The LEs allowed to float in placement and be packed into unused LEs of LogicLock regions may not be placed optimally after migration to the HardCopy Stratix device since they are merged with other LogicLock regions.

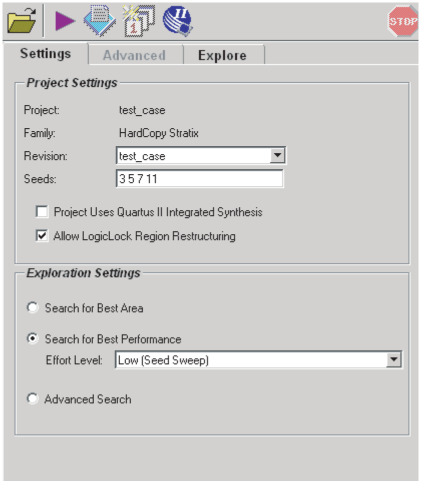
After running the HardCopy Timing Optimization Wizard, the LogicLock region properties are reset to their default conditions. This allows a successful and immediate placement of your design in the Quartus II software. You can further refine the LogicLock region properties for additional benefits.

Altera recommends using the following properties for LogicLock regions in the HardCopy design project:

- Turn off **Soft Region**
- Select either **Auto** or **Fixed** as the **Size** after you are satisfied with the placement and timing result of a LogicLock region in a successful HardCopy Stratix compilation
- Select either **Floating** or **Locked** as the **Location** after you are satisfied with the placement and timing results
- **Reserve Unused Logic** is not applicable in the HardCopy Stratix device placement because logic array block (LAB) contents can not be changed after the HardCopy Timing Optimization Wizard is run

An example of a well partitioned design using LogicLock regions effectively for some portions of the design is shown in Figure 6–1. Only the most critical logic functions required are placed in LogicLock regions in order to achieve the desired performance in the HardCopy Stratix in your open project in the Quartus II software, select **Launch Design Space Explorer** (Tools menu). An example of the DSE GUI and DSE Settings window for the HardCopy Stratix device is shown in Figure 6–2.

Figure 6–2. DSE Settings Window in the DSE GUI



Recommended DSE Settings for HardCopy Stratix Designs

The HardCopy Stratix design does not require all advanced settings or effort-level settings in DSE. Altera recommends using the following settings in DSE for HardCopy Stratix designs:

- In the **Settings** tab (Figure 6–2), make the following selections:
  - Under **Project Settings**, enter several seed numbers in the **Seeds** box. Each seed number requires one full compile of the HardCopy Stratix project.
  - Under **Project Settings**, select **Allow LogicLock Region Restructuring.**
  - Under **Exploration Settings**, select **Search for Best Performance,** and select **Low (Seed Sweep)** from the **Effort Level** menu.
- Turn on **Archive all Compilations** (Options menu).

After running DSE with the seed sweep setting, view the results and identify which seed settings produced the best compilation results. Use the archive of the identified seed, or merge the compilation settings and seed number from the DSE archived project into your primary HardCopy Stratix project.

Performance Improvement Example

With the design used for the performance improvement example in this section, the designer was seeking performance improvement on an HC1S30F780 design for an intellectual property (IP) core consisting of approximately 5200 LEs, 75,000 bits of memory, and two digital signal processing (DSP) multiplier accumulators (MACs). The final application needed to fit in a reserved portion of the HC1S30 device floorplan, so the entire block of IP was initially bounded in a single LogicLock region. The IP block was evaluated as a stand-alone block.

Initial Design Example Settings

The default settings in the Quartus II software version 4.2 were used, with the following initial constraints added:

- The device was set to the target Stratix FPGA device which is the prototype for the HC1S30F780 device:
  ```
  set_global_assignment -name DEVICE
  EP1S30F780C6_HARDCOPY_FPGA_PROTOTYPE
  ```

- A LogicLock region was created for the block to bound it in the reserved region.

- The LogicLock region properties were set to **Auto Size** and **Floating Location**, and **Reserve Unused Logic** was turned on:
  ```
  set_global_assignment -name LL_STATE FLOATING
  set_global_assignment -name LL_AUTO_SIZE ON
  set_global_assignment -name LL_RESERVED OFF
  set_global_assignment -name LL_SOFT OFF
  ```

- Virtual I/O pins were used for the ports of the core since this core does not interface to pins in the parent design, and the I/O pins were placed outside the LogicLock region and are represented as registers in LEs.

The initial compilation results yielded 65.30-MHz f_{MAX} in the FPGA. The block was constrained through virtual I/O pins and a LogicLock region to keep the logic from spreading throughout the floorplan.

for **Speed**. The **Fitter effort** is set to **Standard Fit (highest effort)**. The next features enabled are the **Physical Synthesis Optimizations** as seen in the Tcl assignments below and in Figure 6–5:

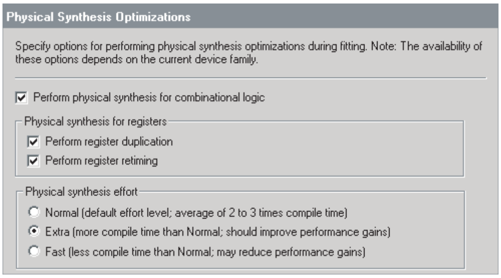
```
set_global_assignment -name
PHYSICAL_SYNTHESIS_COMBO_LOGIC ON

set_global_assignment -name
PHYSICAL_SYNTHESIS_REGISTER_DUPLICATION ON

set_global_assignment -name
PHYSICAL_SYNTHESIS_REGISTER_RETIMING ON

set_global_assignment -name PHYSICAL_SYNTHESIS_EFFORT
EXTRA
```

Figure 6–5. Physical Synthesis Optimization Settings



The compiled design shows a performance increase in the FPGA, running at an f_{MAX} of 74.34 MHz, requiring additional LE resources as a result of the physical synthesis and logic duplication. In this example, you can see how performance can be increased in the Stratix FPGA device at the expense of additional LE resources, as this design's LE resources grew almost 12% over the beginning compilation. The compiled FPGA design's statistics are provided in Table 6–4.

Table 6–4. Compiled FPGA Design Statistics

Result Type	Results
f_{MAX}	74.34 MHz
Total logic elements	5,781/32,470 (17%)
Total LABs	610/3,247 (18%)

Running the HardCopy Timing Optimization wizard on this design and compiling the HardCopy Stratix project yields an f_{MAX} of 92.01 MHz, a 24% improvement over the FPGA timing.

Design Space Explorer

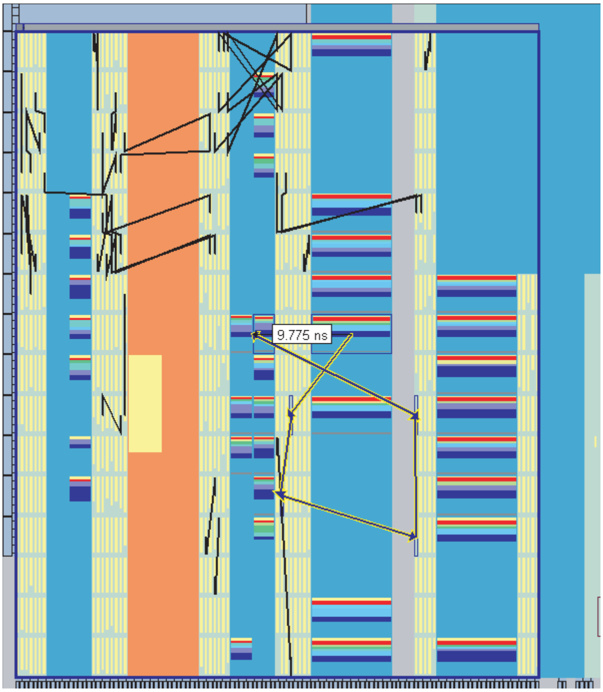
The available Fitter Settings produce an additional performance improvement. The DSE feature is used on the Stratix FPGA device to run through the various seeds in the design and select the best seed point to use for future compiles. This can often yield additional performance benefits as the Quartus II software further refines placement of the LEs and performs clustering of associated logic together.

For this design example, DSE was run with high effort (physical synthesis) and multiple placement seeds. Table 6–5 shows the DSE results. The base compile matches the fifth compile in the DSE variations, showing that the work already done on the design before DSE was optimal. The FPGA project was optimized before running DSE.

Table 6–5. DSE Results

Compile Point	Clock Period: CLK	Logic Cells
Base (Best)	13.451 ns (74.34 MHz)	5,781
1	13.954 ns	5,703
2	13.712 ns	6,447
3	14.615 ns	5,777
4	13.911 ns	5,742
5	13.451 ns	5,781
6	14.838 ns	5,407
7	14.177 ns	5,751
8	14.479 ns	5,827
9	14.863 ns	5,596
10	14.662 ns	5,605
11	14.250 ns	5,710
12	14.016 ns	5,708
13	13.840 ns	5,802
14	13.681 ns	5,788
15	14.829 ns	5,644

Figure 6–8. New Critical Path



Examining this new critical path placement, you can see that there is room for further performance improvement through additional location assignments. The current slowest path is 9.775 ns of delay. Manually moving the LABs in this critical path and placing them between the M4K and M512 endpoints, and subsequently recompiling, shows improved results not only for this path, but for several other paths, as this path contained a major timing bottleneck. The critical path between this start and endpoint was reduced to 8.797 ns (Figure 6–9). However, the entire design only improved to 100.30 MHz because other paths are now the slowest paths in the design. This illustrates that fixing one major bottleneck path can raise the entire design performance since one high fanout node can affect multiple timing paths, as was the case in this example.