FPGAs at 28nm: Meeting the Challenge of Modern Systems-on-a-Chip

Vaughn Betz
Senior Director, Software Engineering
Altera
Overview

- Process scaling & FPGAs
  - End user demand
  - Technological challenges

- FPGAs becoming SoCs
  - Stratix V: more hard IP
  - FPGA families targeted at more specific markets

- Stratix V & 28 nm
  - Challenges & features
  - Partial reconfiguration

- Designer productivity
  - Challenges
  - Possible software stack solutions
Broad End Market Demand

Communications
- Mobile internet and video driving bandwidth at 50% annualized growth rate
- Fixed footprints

Broadcast
- Proliferation of HD/1080p
- Move to digital cinema and 4k2k

Military
- Software defined radio
- More sensors, higher precision
- Advanced radar

Consumer/industrial
- Smart cars and appliances
- Smart Grid

Need more processing in same footprint, power and cost

© 2010 Altera Corporation
ALTERA, NIOS, QUARTUS & STRATIX are Reg. U.S. Pat. & Tm. Off. and Altera marks in and outside the U.S.
Driving Factors—Mobility and Video

Mobile Bandwidth

<table>
<thead>
<tr>
<th>Mobile Bandwidth</th>
<th>Minimum Bandwidth</th>
<th>Maximum Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>1G 1983</td>
<td>1</td>
<td>1 Gbps</td>
</tr>
<tr>
<td>2G 1991</td>
<td>10</td>
<td>10 Gbps</td>
</tr>
<tr>
<td>3G 2001</td>
<td>100</td>
<td>100 Gbps</td>
</tr>
<tr>
<td>4G 2009</td>
<td>1,000,000</td>
<td>10 Tbps</td>
</tr>
<tr>
<td>5G ~2017</td>
<td>10,000,000</td>
<td>100 Tbps</td>
</tr>
</tbody>
</table>

(LTE)

Video Bandwidth

<table>
<thead>
<tr>
<th>Video Bandwidth</th>
<th>Streaming Bandwidth (Mbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SD</td>
<td>480p</td>
</tr>
<tr>
<td>720p</td>
<td>1080i</td>
</tr>
<tr>
<td>1080p</td>
<td>4K2K</td>
</tr>
<tr>
<td>4K2K</td>
<td></td>
</tr>
</tbody>
</table>
Evolution of Video-Conferencing

**Today**

- High end in 2000: 384 kbps
- Cisco telepresence: 15 Mbps

**Tomorrow**
Communication Processing Needs

- More bandwidth: CAGR of 25% to 131% / year [By domain, Cisco]
- More data through fixed channel → more processing per symbol
- Security and quality of service needs → deep packet inspection
Moore’s Law: On-Chip Bandwidth

- Datapath width * datapath speed
- 40% / year increase in transistor density
- 20% / year transistor speed until ~90 nm
  - Total ~60% gain / year

- 40 nm and beyond:
  - Little intrinsic transistor speed gain once power controlled
  - ~40% gain / year from pure scaling
  - Need to innovate to keep up with demand
Increasing I/O Bandwidth

26% increase / year per lane
Modest growth in # lanes / chip
Scaling Economics

- TSMC Fab 15: $9B
  - 40 & 28 nm
- 90's fab cost → fabless industry

- Chip cost @ 28 nm ~$100M
- Need big market → go programmable
- “Chipless” industry emerging

© 2010 Altera Corporation
ALTERA, NIOS, QUARTUS & STRATIX are Reg. U.S. Pat. & Tm. Off. and Altera marks in and outside the U.S.
FPGAs Becoming SoCs

Example: Stratix V
## From Glue Logic to SoC

<table>
<thead>
<tr>
<th>Year</th>
<th>Generation</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>1992</td>
<td>Flex 8k</td>
<td>LUTs, FFs, Basic I/Os</td>
</tr>
<tr>
<td>1995</td>
<td>Flex 10k</td>
<td>Block RAM</td>
</tr>
<tr>
<td>1999</td>
<td>APEX 20K</td>
<td>PLLs, Complex I/Os, Hard Processor</td>
</tr>
<tr>
<td>2002</td>
<td>Stratix</td>
<td>DSP Blocks</td>
</tr>
<tr>
<td>2003</td>
<td>Stratix GX</td>
<td>Serial Transceivers</td>
</tr>
<tr>
<td>2008</td>
<td>Stratix IV GX</td>
<td>Hard PCIe Gen1/2</td>
</tr>
<tr>
<td>2010</td>
<td>Stratix V GX</td>
<td>Hard PCIe Gen1/2/3, Hard 40G / 100G Ethernet</td>
</tr>
</tbody>
</table>
Hard Block Evaluation

Develop Parameterized Soft IP

Create Configurable Hard IP

Specific IP in soft fabric

Area, Power, Speed

Estimate Usage & Dev. Cost

Include routing ports!

Net Win?

Hard PCIe?
Stratix V Transceivers

FPGA Fabric

- Fractional PLLs (fPLL)
- Embedded HardCopy Block

Power Down

Hard PCS

LC Transmit PLLs

Clock networks

Transceiver PMA

Power Down

Transceiver PMA

Transceiver PMA

Transceiver PMA

Transceiver PMA

Transceiver PMA

Transceiver PMA

Transceiver PMA

Transceiver PMA

Transceiver PMA

Transceiver PMA

Transceiver PMA

© 2010 Altera Corporation

ALTERA, NIOS, QUARTUS & STRATIX are Reg. U.S. Pat. & Tm. Off. and Altera marks in and outside the U.S.
Embedded HardCopy Blocks

- Metal programmed: reduces cost of adding device variants with new hard IP
- 700K equivalent LEs
- 14M ASIC gates
- 5X area reduction vs. soft logic
- 65% reduction in operating power
- Very low leakage when unused

PCIe Gen3
40G/100G Ethernet
Other/Custom
Hard IP Example: PCIe & Interlaken

Interlaken – PCI Express Switch/Bridge

Stratix V FPGA
5SGXA7

~630K LEs

12 Ch @ 5G Interlaken
12 Ch @ 5G Interlaken

PCle Gen3 x8
PCle Gen3 x8

630K LEs + 440K LEs = 1,070K LEs

Lower power
Higher effective density
Guaranteed timing closure → ease of use

<table>
<thead>
<tr>
<th>Hard IP</th>
<th>LE Savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interlaken (24 Ch @ 5K LEs)</td>
<td>120K LEs</td>
</tr>
<tr>
<td>PCIe Gen3 x8 (2 x 160K LEs)</td>
<td>320K LEs</td>
</tr>
<tr>
<td>Total LE savings</td>
<td>440K LEs</td>
</tr>
</tbody>
</table>
Variable-Precision DSP Block

- Efficiently supports 9x9, 18x18 and 27x27 multiplies
  - 27x27 well suited to floating point
- Cascade blocks for larger multiplies
- Can store filter coefficients in register bank inside DSP
# Stratix V Maximum Capacities

<table>
<thead>
<tr>
<th>Feature</th>
<th>Stratix V</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logic Elements</td>
<td>1.1 M</td>
</tr>
<tr>
<td>RAM bits</td>
<td>52 Mb + 7.3 Mb</td>
</tr>
<tr>
<td>18x18 multipliers</td>
<td>3680</td>
</tr>
</tbody>
</table>
| High-speed serial links      | GX: 66 full-duplex @ 12.5 Gb/s  
                                 | GT: 4 @ 28 Gb/s + 32 @ 12.5 Gb/s  |
| Hard PCIe blocks             | 4                          |
| Hard 40G / 100G PCS          | Yes                        |
| Memory interfaces            | 7 x 72-bit DDR3 DIMM @ 800 MHz |
| On-chip memory bandwidth     | ~20,000 GB/s               |
| I/O Bandwidth                | ~300 GB/s                  |
| 18x18 MACs                   | 1,840 GMAC/s               |
100G Optical System (Stratix II GX)

Other Challenges & Enhancements @ 28 nm
## Controlling Power

<table>
<thead>
<tr>
<th>Stratix V FPGA Power Reduction</th>
<th>Lower Static Power</th>
<th>Lower Dynamic Power</th>
</tr>
</thead>
<tbody>
<tr>
<td>(New techniques highlighted in yellow)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>28-nm process (high-k, more strain, small C)</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Programmable Power Technology</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Lower core voltage (0.85 V)</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Extensive hardening of IP, Embedded HardCopy Blocks</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Hard power-down of more functional blocks</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>More granular clock gating</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Selective use of high-speed transistors</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Dynamic on-chip termination</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Quartus II software PowerPlay power optimization</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
Fabric Performance

- Low operating voltage key to reasonable power
  - But costs speed
  - Logic still speeding up, routing more challenging
    → Optimize process for FPGA circuitry (e.g. pass gates)
    → Trend to bigger blocks / more hard IP

- Wire resistance rapidly increasing
  → Co-optimize metal stack & FPGA routing architecture
  - Greater mix of wire types and metal layers (H3, H6, H20, V4, V12)

- Delay to cross chip not scaling
  - Above ~300 MHz, designers pipelining interconnect
Fabric: More Registers

- Double the logic registers (4 per ALM)
- Faster registers
- Aids deep pipelining & interconnect pipelining

Memory mode: 5 registers
- Re-uses 4 ALM registers
- Adds extra register for write address
- Easier timing
Metastability Robustness
Metastability Robustness

- Loop gain at Vdd/2 dropping $\rightarrow \tau_{met}$ increasing
- Solution: register design (e.g. use lower Vt)
- Solution: CAD system analyzes & optimizes
  - 20,000 to 200,000 increase in MTBF
Pass Transistors

- Most area-efficient routing mux
- But Vdd – Vt dropping

- Bias Temperature Instability (BTI) makes worse
  - Increase / hysteresis in Vt due to Vgs state over time
  - All circuits affected, but pass transistors more sensitive to Vt shift

- Careful process and circuit design needed

- Future scaling:
  - Full CMOS?
  - Opening for a new programmable switch?
Soft Errors

- **Block RAM**: new M20K block has hard ECC
  - MLAB: can implement ECC in soft logic
- **Configuration RAM**: background ECC
  - But could take up to 33 ms to detect
- **Config. RAM circuit design to minimize SEU**

**Trends with SRAM scaling:**
- Smaller target $\rightarrow$ lower FIT rate / Mb (constant per die)
- Less charge $\rightarrow$ higher FIT for alpha, stable for neutron
- Will this stabilize at an acceptable rate?
- Known techniques to greatly reduce (at area cost) $\rightarrow$ does not threaten scaling
Stratix V Partial Reconfiguration

- Very flexible HW
- Reconfigure individual LABs, block RAMs, **routing muxes**, …
- Without disrupting operation elsewhere

<table>
<thead>
<tr>
<th>CRAM address space</th>
<th>Frame 1</th>
<th>Frame 2</th>
<th>…</th>
<th>Frame m</th>
<th>Frame m+1</th>
<th>Frame m+2</th>
<th>…</th>
<th>Frame n</th>
<th>Frame n+1</th>
<th>Frame n+2</th>
<th>…</th>
<th>Last Frame</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit i</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit i+1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Last Bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Non-PR Region**

**PR Region**
Partial Reconfiguration (PR) Overview

- Software flow is key
  - Build on existing incremental design & floorplanning tools
  - Enter *design intent*, automate low-level details
  - Simulation flow for operation, *including reconfiguration*

- Partial reconfiguration can be controlled by soft logic, or an external device
  - Load partial programming files while device operating

- Target: *multi-modal* applications
Example System: 10*10Gbps → OTN4 Muxponder

Client Side

10Gbs → 10GbE

10Gbs → OTN2

10Gbs → OTN2

Line Side

Channel 1

Channel 2

Channel 10

MUXPonder

OTN4

100Gps
Design Entry & Simulation

- One set of HDL
- Tools to simulate during reconfig

```verilog
module reconfig_channel (clk, in, out);
    input clk, in;
    output [7:0] out;

    parameter VER = 2; // 1 to select 10GbE, 2 to select OTN2

    generate
        case (VER)
            1: gige m_gige (.clk(clk), .in(in), .out(out));
            2: otn2 m_otn2 (.clk(clk), .in(in), .out(out));
            default: gige m_gige(.clk(clk), .in(in), .out(out));
        endcase
    endgenerate

endmodule
```
Incremental Design Flow Background

- Specify *partitions* in your design hierarchy
- Can independently recompile any partition
  - CAD optimizations across partitions prevented
  - Can preserve synthesis, placement and routing of unchanged partitions

![Diagram showing the incremental design flow with partitions: Top, Channel 1, Channel 2, MUXponder, OTN4.](image-url)
Partial Reconfig Instances

Top

Partial Reconfig Partition 2
C1, OTN2
C1, 10GbE
Partial Reconfig Partition 2
C2, OTN2
C2, 10GbE

Static partition

MUXponder
OTN4
Partial Reconfiguration: Floorplanning

- Define partial reconfiguration regions
  - Non-rectangular OK
  - Any number OK

- Works in conjunction with transceiver dynamic reconfiguration for dynamic protocol support
  - “Double-buffered” partial reconfig
Physical: I/Os

- PR region I/Os must stay in same spot
  - So rest of design can communicate with any instance

- Same wire?
  - FPGAs not designed to route to/from specific wires

- Solution: **automatically** insert “wire LUT”
  - **Automatically** lock down in same spot for all instances
Physical: Route-Throughs

- Can partially reconfigure individual routing muxes
- Enables routing through partial reconfig regions
  - Simplifies / removes many floorplanning restrictions
  - Quartus II records routing reserved for top-level use
  - Prevents PR instances from using it
Extending & Improving the Software Stack
Design Flow Challenges

- **HDL: low-level parallel programming language**
  - RTL ~300 kLOCs, behavioural ~40 kLOCs [NEC, ASPDAC04]

- **Timing closure**
  - Fabric speed flattening, but processing needs growing
  - Datapaths widening, device sizes growing exponentially
  - 4x28 Gbps → 336 bit datapath @ 333 MHz → need good P & R
  - Need more latency? → may cause major HDL changes

- **Compile, test, debug cycle** slower than SW
  - And tools to observe HW state less mature
  - Any timing closure issues exacerbate

- **Firmware development needs working HW**
The Competition: Many Core

Tilera TILE64
Competition: ASSP w/HW Accelerators

Ex. Cavium – Octeon CN68XX

85 application accelerators

65 nm in Q4 2010

2 process generations behind FPGAs
“Bespoke” ASSPs in FPGAs

- Connect IP with SoPC Builder
  - Integrates system & builds software headers

- Next generation: general Network-on-a-Chip
  - Topology, latency: selectable
  - Scalable enough to form heart-of-the-system
High-Level Synthesis

- Good results in some problem domains (e.g. DSP kernels)
- Often difficult to scale to large programs
- Debugging and timing closure difficult
  - Unclear how the code relates to the synthesized solution
  - How to change the ‘C’ code to make hardware run faster?
  - Few tools to drive profiling data back to the high-level code
  - Few tools to debug HW in a software-centric environment
OpenCL: *Explicitly Parallel C*

- The OpenCL programming model allows us to:
  - **Define Kernels**
    - Data-parallel computational units → can hardware accelerate
    - Including communication mechanism to kernels
  - **Describe parallelism within & between kernels**
  - **Manage Entire Systems**
    - Framework for mix of HW-accelerated and software tasks

- Still C
- Multi-target
OpenCL Structure

__kernel void sum { ... }
__kernel void transpose {...}
float cross_product { ... }

__kernel void sum
(__global const float *a,
__global const float *b,
__global float *answer)
{
    int xid = get_global_id(0);
    answer[xid] = a[xid] + b[xid];
}

Program: kernels and functions
Task-level parallelism, overall framework

Kernels: data-level parallelism
Suitable for HW or parallel SW implementation
Specify memory hierarchy

© 2010 Altera Corporation
ALTERA, NIOS, QUARTUS & STRATIX are Reg. U.S. Pat. & Tm. Off. and Altera marks in and outside the U.S.
The Past (1984): Editing Switches
The Present: HDL Design Flow

Timing & Other Constraints

Synthesis

Placement and Routing

Timing and Power Analyzer

Timing, Power and Area Optimized Design

Verilog, VHDL
The Future?

Extract Communication

SoPC Builder

Communication Fabric

Fast debug

OpenCL

Kernel Compilers

Control SW

HW kernels or SW kernels

RTL becomes assembly language

© 2010 Altera Corporation

ALTERA, NIOS, QUARTUS & STRATIX are Reg. U.S. Pat. & Tm. Off. and Altera marks in and outside the U.S.
Summary
Summary

- Huge demand for more processing
  - Possibly outstripping Moore’s law & off-chip bandwidth

- FPGAs becoming SoCs
  - More heterogeneous/hard function units
  - FPGAs specializing to markets

- 28 nm & Stratix V
  - -30 to -50% power, 1.5x I/O bandwidth, 1.5x – 2x more processing
  - Partial reconfiguration

- FPGA robustness with scaling
  - Innovation overcoming issues → scaling continues

- Tool innovation needed
  - Higher-level, fast debug cycles, push-button timing closure