Is the 2nd Wave of HLS the One Industry Will Surf on?

Jason Cong
Chancellor’s Professor
UCLA Computer Science Department
cong@cs.ucla.edu

Chief Technology Advisor
AutoESL Design Technologies, Inc.
www.autoesl.com
The Demand for High-Level Synthesis is Real

- Embedded processors are in almost every SoC
  - Need SW/HW co-design and exploration
  - C/C++/SystemC is a more natural starting point
- Huge silicon capacity requires high-level of abstraction
  - 700,000 lines of RTL for a 10M gate design is too much!
- Verification drives the acceptance of SystemC
  - Need executable model to verify against RTL design
  - More and more SystemC models are available
- Need and opportunity for aggressive power optimization
  - Simultaneous functional, structural, and temporal optimization for power.
- Accelerated computing or reconfigurable computing also need C/C++ based compilation/synthesis to FPGAs
Opportunity for High-Level Synthesis

- Life of an RTL designer is getting more and more miserable
  - Complexity (80+M gates)
  - Correctness - First-time working silicon ($2M mask cost)
  - Performance (interconnects dominate)
  - Routability (what/how to measure at RTL level??)
  - Power (yet another dimension)
  - ...

- Real *opportunity* for automation/exploration by high-level synthesis with BETTER quality
Significant Progress on HLS

- Wide acceptance of C/C++/SystemC for design modeling and simulation
  - Pave the way for C/C++/SystemC based HLS

- Better compilation infrastructure
  - Leveraging the progress in the compiler community

- Advancements of core HLS algorithms – e.g. research from UCLA:
  - SDC-based scheduling
  - Distributed register file based architecture
  - Simultaneous computation and communication synthesis
  - Pattern-based synthesis
  - HLS for power …
A New Generation of HLS Tool – E.g. AutoESL

- Best language coverage
  - Pure ANSI C/C++ synthesis
  - SystemC/TLM synthesis

- Aggressive power optimization
  - Clock gating
  - Operation gating
  - Frequency scaling
  - Power/performance trade-off …

- Best QoR
  - Leveraging 8+ years of research from UCLA on ESL synthesis

- Ideal for reuse and arch-exploration
  - Platform-based synthesis
  - Separate source & constraint
  - Link to implementation flows

C/C++/SystemC

Compilation & Elaboration
Advanced Code Transformation

Behavior & Interface Synthesis
Performance/Power/Area Optimizations

Microarchitecture Generation

RTL HDLs
RTL SystemC

RTL Constraints (Timing/Layout)

ASIC/FPGA RTL Synthesis
Place-and-Route

User Constraints & Directives

Unique ESL synthesis technology

Simulator/Verifier
**MPEG4 4CIF by AutoPilot vs Manual Design**

- Frame rate
  - 60 fps based on estimation for 4CIF video
  - 200 fps on v2p board for 1CIF video

<table>
<thead>
<tr>
<th>Block</th>
<th>Manual Design</th>
<th>AutoPilot</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BRAM#</td>
<td>MULT#</td>
</tr>
<tr>
<td>Parser/VLD</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Copy Control</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Motion Comp</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Texture/IDCT</td>
<td>6</td>
<td>25</td>
</tr>
<tr>
<td>Texture Update</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>30</td>
</tr>
</tbody>
</table>

|                | 0.0% | -13.3% | 3.2%  |
Quickly generate multiple solutions with the same sample rate, but different area/power profiles

Manual design took 4 months while C-based synthesis using AutoPilot in two weeks

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Latency</th>
<th>Area (mm^2)</th>
<th>Clock (MHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>manual</td>
<td>96</td>
<td>1.17</td>
<td>150</td>
</tr>
<tr>
<td>config1</td>
<td>116</td>
<td>1.10</td>
<td>150</td>
</tr>
<tr>
<td>config2</td>
<td>86</td>
<td>1.12</td>
<td>150</td>
</tr>
<tr>
<td>config3</td>
<td>81</td>
<td>1.30</td>
<td>100</td>
</tr>
<tr>
<td>config4</td>
<td>64</td>
<td>1.55</td>
<td>75</td>
</tr>
</tbody>
</table>

TSMC65nmLP Library
Next Challenges

- Even better QoR, out-of-box success
  - Further algorithmic innovation for HLS
- Aggressive power optimization
- Physical synthesis above RTL
- Integrated synthesis and verification
- Synthesis support for variability and reliability