Model-Based Design for Medical Applications using HDL Coder

Rob Reilink, M.Sc Ph.D
DEMCON Profile

- Established in 1993 (25 yr)
- ~ 500 employees
- ~ €50M turnover
- >10 year experience with Model-Based Design
Model-based design @ DEMCON

- Link between model and requirements
- Execution specs
- Test & verification
- Design with simulation
- Automatic code generation
- Code generation
- Continuous integration
- Unit testing
- Processor in the loop
- Hardware in the loop
- Simulation of concepts
- Detailed simulation of components
- Rapid prototyping
- What-if studies

2007
PC
xPC / Simulink Realtime

- More compact

2011
DSP / microcontroller
Embedded Coder

- Higher speed

2017
FPGA
HDL Coder

More compact
Higher speed
Use case: precision cut surgical instrument

- Piezo actuator driven by adjustable sine wave
- Class-D power amplifier for energy efficiency
- Business case: more compact, more energy-efficient, more flexible
Piezo actuator needs to be driven at its resonance frequency (~40kHz)

Adjust frequency to achieve 0° phase difference between voltage and current
Frequency domain behavior of a piezo actuator

- Phase-locked loop is used to control piezo frequency
Challenges

▪ Uncertainty in piezo actuator behavior
  ▪ Product variations
  ▪ Interaction with tissue
  ▪ Desired behavior for optimal cutting

▪ Short development time

▪ Reliable PLL stability / locking
▪ More complicated control & signal processing
▪ High loop frequency

Need for testing using actual actuator on tissue!
Use case: precision cut surgical instrument high-level model

- **Sine generation**
- **Measurement of phase of current**
- **Measurement of phase of voltage**
- **PI-controller**
- **Measured piezo voltage and current**

To PWM generator
From reference implementation to FPGA: fixed point

- High-level (golden reference) model designed by Mechatronic System Engineer
- Fixed-point conversion
- ‘Sine Wave Function’ blocks replaced by ‘Sine and Cosine HDL Optimized’ blocks
- Trigonometric block <atan2> replaced by CORDIC-based four quadrant inverse tangent Matlab function
- Target low-cost Xilinx Artix-7 FPGA (no SoC required)
Model-based design verification: FPGA model vs reference
HDL Coder Workflow

- fixed-point conversion
- floating-point support
- discrete-time
- HDL supported blocks
- oversampling factor
- workflow advisors
HDL Coder timing analysis, critical path

- Very useful feature to find computational bottle-necks
- In our case: sine & cosine computation
HDL Coder Pipelining

Lesson learned: automatic delay balancing is great for complex signal path applications.
Lesson learned: Automatic resource sharing is a very powerful and flexible feature.
From reference implementation to FPGA: floating point

- For the final implementation: use of floating point
- Model synthesizable within few days
- Only minor adaptions required: single precision datatypes and non-HDL blocks replaced
- IEEE (native) floating point support for all trigonometric & math blocks (sin, cos, sincos, atan, atan2)

<table>
<thead>
<tr>
<th></th>
<th>Fixed point</th>
<th>Floating point</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUTs</td>
<td>10k</td>
<td>25k</td>
</tr>
<tr>
<td>DSP slices</td>
<td>50</td>
<td>100</td>
</tr>
<tr>
<td>Development time</td>
<td>~1 week</td>
<td>~1 day</td>
</tr>
</tbody>
</table>

~2x more resources
~5x less development effort
Achievements

▪ Early prototype with limited development effort
▪ Energy-efficient piezo actuator
▪ Cost-efficient by incorporating controller in the existing FPGA
▪ Reliable PLL operation
▪ Fast iterations using HDL coder
Conclusions

- Less chances of coding errors due to high-level implementation
- Improves collaboration between FPGA engineers and other disciplines (system engineers)
- Resource sharing & pipelining optimizations are much easier as compared to bare VHDL coding
  - Only setting appropriate numbers / check boxes instead of re-implementing
- Native floating point support speeds up transition from high-level model to implementation
  - No / less need to worry about data types
  - Good support of a.o. trigonometric functions
  - Same model for ‘high level’ simulations and for FPGA code generation

→ Current project status: alpha-phase hardware validation