MODEL-BASED DESIGN FOR FPGA DEVELOPMENT
GETC-I
CONTENT

• Why?
• Benefit
• Concept
• Use case
WHY?

- **Objectives**
  - A framework and supporting standard work and work instructions for Model based FW algorithm development for aerospace application.

- **KPIs**
  - Reusability for different algorithm module as library
  - Cost reduction (~30 to 40%)
  - Reduction in lead time for development (~30%)

- **Business Impact:**
  - Cost Reduction
    - Rapid initial design (rapid prototyping, continuous validation & verification across the different layer of design)
    - Easy rework and verification (verification based on model, possibility to generate test benches from Models)
  - Reduce risk of design error
  - Lower needed skill level for FPGA design of complex algorithm

- **Background**
  - In the current projects FW development is done manually without or minimum support of MBD
  - Model based development is the way forward for embedded development as traditional (manual driven) solution require extensive effort.
  - Framework will be used for electronics development for different type of controllers, which are algorithm rich in content.

- **Concept Block Diagram:**

---

© 2022 Collins Aerospace | Collins Aerospace proprietary | "This document does not contain any Export Controlled Technical Data."
METHOD DESCRIPTION

Current State

Challenges:

- Fixed point algorithm development and verification increase complexity
- Requirements to code generation add multilayer process and risk for error
- Close loop simulation and test analysis not easy in the current FPGA development setup
METHOD DESCRIPTION

Future State

Advantages:

- Reduce time and cost for development
- Early validation and verification using MIL and Co-Simulation (virtual integration)
- Recuse of same test case and plant model for testing
- Reusability and scalability
- Robustness
First step
Design by Model:
Design Model including Low Level Requirements and Architecture
Automatic HDL code generation

Second step
Co-simulation:
First step+VHDL simulation result comparison with Model

Third step
High Level Modeling & FPGA In the Loop:
Second step + Specification Model including Functional behavior, interface, performance and safety Verification up to Physical with FIL*
USE CASE

Motor Control law development:
• A multi-Domain problem
  • Embedded system: Control law, sensor processing and BIT (algorithm and signal processing)
  • Physical modeling (Electromanical and sensor modeling)
  • Power electronics modeling (motor derives)
• Flow of information and data from system spec till design
• Design and auto code generation
• Early verification and validation
• Co-Simulation
A simple PI control law with anti windup and command saturator in S domain
SYSTEM MODEL FOR CONTROL LAW
CONTROL LAW IN DIFFERENT DOMAIN
MODEL IN LOOP FOR CONTROL LAW
MODEL IN LOOP FOR CONTROL LAW

Post-Processing of results

<table>
<thead>
<tr>
<th>Step Response</th>
<th>S-domain Model</th>
<th>Floating Point Model (DSP)</th>
<th>Fixed point Model (FPGA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RiseTime</td>
<td>0.0125</td>
<td>0.0107</td>
<td>0.0121</td>
</tr>
<tr>
<td>TransientTime</td>
<td>0.0403</td>
<td>0.0374</td>
<td>0.0367</td>
</tr>
<tr>
<td>SettlingTime</td>
<td>0.0403</td>
<td>0.0374</td>
<td>0.0367</td>
</tr>
<tr>
<td>SettlingMin</td>
<td>361.0889</td>
<td>375.5305</td>
<td>366.0491</td>
</tr>
<tr>
<td>SettlingMax</td>
<td>416.3885</td>
<td>428.8628</td>
<td>414.4917</td>
</tr>
<tr>
<td>Overshoot</td>
<td>4.0971</td>
<td>7.2157</td>
<td>3.6229</td>
</tr>
<tr>
<td>Undershoot</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Peak</td>
<td>416.3885</td>
<td>428.8628</td>
<td>414.4917</td>
</tr>
<tr>
<td>PeakTime</td>
<td>0.1475</td>
<td>0.144</td>
<td>0.146</td>
</tr>
</tbody>
</table>
HDL CO-SIMULATION

HDL co-simulation provide the following advantages:

1. Design verification and validation
   • HDL code verification against its requirements (implementation correctness)
   • Validation of derived requirements such as sample time and discretization and fixed-point size

2. Virtual Integration
   • Muti-Domain simulation in one place to check the following:
     • Interface
     • Algorithm correctness
     • Dynamics of system
CO-SIMULATION FOR CONTROL LAW
(METHOD 1: AUTO GENERATED)
CO-SIMULATION FOR CONTROL LAW
CO-SIMULATION FOR CONTROL LAW
CO-SIMULATION FOR CONTROL LAW
## CO-SIMULATION FOR CONTROL LAW

### Step Response

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Fixed point Model (FPGA) Co-Simulation in QuestaSim</th>
<th>Fixed point Model (FPGA) Simulation in Mathwork</th>
</tr>
</thead>
<tbody>
<tr>
<td>RiseTime</td>
<td>0.0121</td>
<td>0.0121</td>
</tr>
<tr>
<td>TransientTime</td>
<td>0.0367</td>
<td>0.0367</td>
</tr>
<tr>
<td>SettlingTime</td>
<td>0.0367</td>
<td>0.0367</td>
</tr>
<tr>
<td>SettlingMin</td>
<td>366.0491</td>
<td>366.0491</td>
</tr>
<tr>
<td>SettlingMax</td>
<td>414.4917</td>
<td>414.4917</td>
</tr>
<tr>
<td>Overshoot</td>
<td>3.6229</td>
<td>3.6229</td>
</tr>
<tr>
<td>Undershoot</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Peak</td>
<td>414.4917</td>
<td>414.4917</td>
</tr>
<tr>
<td>PeakTime</td>
<td>0.146</td>
<td>0.146</td>
</tr>
</tbody>
</table>
DISCLAIMER

This document does not contain any technical data or Collins proprietary data. The information flow/data used(any values used/images/figures/graphs/diagrams do not contain any technical data from any past, current or ongoing Collins Project. The data/values used are generic. All images/only sample non-technical data values.
THANK YOU