

Embedded Award 2017

Firma: MathWorks

Produkt: HDL Coder

Produktbeschreibung HDL Coder:

<p>HDL Coder™ generates portable, synthesizable Verilog® and VHDL® code from MATLAB® functions, Simulink® models, and Stateflow® charts. HDL Coder thereby enables FPGA programming and ASIC synthesis of algorithms, e.g. for digital signal processing and communications, for digital controllers, and for finite state machines without the need for developers to be HDL experts.</p>	<p>HDL Coder™ generiert plattformunabhängigen, synthetisierbaren Verilog® - und VHDL® - Programmcode aus MATLAB® -Funktionen, Simulink® -Modellen und Stateflow® - Diagrammen. Damit ermöglicht HDL Coder die FPGA-Programmierung und ASIC-Synthese von Algorithmen z.B. für die digitale Signalverarbeitung und Kommunikationstechnik, für digitale Steuerungen und finite Zustandsautomaten, ohne dass Entwickler HDL-Experten sein müssen.</p>
<p>For the implementation of the generated HDL code on selected FPGA development platforms, HDL Coder has a guided workflow and integrates the tools of the FPGA manufacturers under a uniform interface. The generated HDL code can be optimized to meet various objectives such as area, speed and power, with early resource and critical paths estimation at the RTL level.</p>	<p>Für die Implementierung des generierten HDL Codes auf ausgewählten FPGA-Entwicklungsplattformen verfügt HDL Coder über einen geführten Workflow und integriert die Tools der FPGA-Hersteller unter einer einheitlichen Oberfläche. Der generierte HDL Code lässt sich hinsichtlich verschiedener Kriterien, wie Fläche, Geschwindigkeit und Verlustleistung optimieren, mit frühzeitigen Abschätzungen der Ressourcen und kritischen Pfade auf RTL-Ebene.</p>
<p>HDL Coder accelerates the development process and enables developers to explore different hardware architectures, thus finding the best solution. Additionally, HDL Coder enables early verification and prototyping in typical test environments used by algorithm designers and supports certification processes for applications with functional safety requirements, e.g. through traceability of generated HDL code. HDL Coder generates VHDL and Verilog test benches for rapid verification of generated HDL code.</p>	<p>HDL Coder beschleunigt den Entwicklungsprozess und ermöglicht es dem Entwickler, verschiedene Hardware-Architekturen zu untersuchen und die optimale Lösung zu finden. Darüber hinaus ermöglicht HDL Coder eine frühzeitige Verifikation und Prototyping mit den Test-Umgebungen der Algorithmen-Entwickler und unterstützt den Zertifizierungsprozess für Anwendungen mit funktionalen Sicherheitsanforderungen, u.a. durch die Nachverfolgbarkeit des generierten HDL-Codes. HDL Coder generiert zudem spezifisch anpassbare VHDL- und Verilog-Testbenches für die schnelle Verifizierung des generierten HDL-Codes.</p>
<p>On the one hand, the HDL Workflow Advisor in HDL Coder automatically converts MATLAB code from floating-point to fixed-point. This capability lets you model your algorithm at a high level using abstract MATLAB constructs</p>	<p>Der enthaltene HDL Workflow Advisor konvertiert einerseits MATLAB-Code automatisch von Gleitkomma- zu Festkomma-Code. Dadurch kann ein Algorithmus auf einer hohen Ebene mithilfe abstrakter MATLAB-</p>

<p>and System objects while providing options for generating HDL code that is optimized for hardware implementation.</p> <p>On the other hand, HDL Coder Native Floating Point lets you generate synthesizable VHDL or Verilog directly from single-precision Simulink models, eliminating months of tedious fixed-point conversion. Hardware design has traditionally required data types to be converted to fixed point in order to realize mathematical operations on FPGA/ASIC. But many applications contain calculations with high dynamic range, which requires very large fixed-point word sizes to accommodate the magnitude and precision. Native Floating Point does not require floating-point processing units or hard floating-point DSP blocks on the target ASIC or FPGA and can even be extended to embedded processors and programmable logic controllers that do not have built-in floating-point units, allowing users to model high dynamic range applications with pure integer hardware.</p>	<p>Konstrukte und System Objects modelliert werden, während gleichzeitig Optionen für die Generierung von HDL-Code bereitgestellt werden, die für die Hardware-Implementierung optimiert sind.</p> <p>Durch die native Gleitkomma-Unterstützung kann andererseits zielunabhängiger, synthetisierbarer VHDL- oder Verilog-Code aus Gleitkommamodellen mit einfacher Genauigkeit generiert werden, ohne dass eine aufwändige Festkomma-Konvertierung erforderlich wäre. Hardwareentwicklung erforderte traditionell diese Konvertierung, um mathematische Operationen auf FPGA/ASIC zu realisieren. Manche Anwendungen basieren jedoch auf Berechnungen mit hohem Dynamikumfang, wodurch sehr große Festkomma-Wortlängen erforderlich wären, um sowohl Magnitude als auch Genauigkeit Rechnung zu tragen. Native Gleitkomma-Unterstützung benötigt keine Gleitkomma-Prozessoren oder harte Gleitkomma-DSP-Blöcke auf FPGA/ASIC und kann sogar auf eingebetteten Prozessoren und PLCs genutzt werden, die keine Gleitkomma-Einheiten aufweisen, wodurch Nutzer Anwendungen mit hohem Dynamikumfang auf reiner Integer-Hardware modellieren können.</p>
<p>HDL Coder provides a workflow advisor that automates the programming of Xilinx® and Intel® FPGAs. You can control HDL architecture and implementation, highlight critical paths, and generate hardware resource utilization estimates. HDL Coder provides traceability between your Simulink model and the generated Verilog and VHDL code, enabling code verification for high-integrity applications adhering to DO-254 and other standards.</p>	<p>Der HDL Workflow Advisor, der die Programmierung von Xilinx® - und Intel® -FPGAs automatisiert, bietet Kontrolle über die HDL-Architektur und Implementierung, hebt kritische Pfade hervor und erstellt Hardwareressourcennutzungs-Schätzungen. HDL Coder bietet Rückverfolgbarkeit zwischen dem Simulink-Modell und dem generierten Verilog- und VHDL-Code und ermöglicht die Code-Verifizierung für Anwendungen hoher Integrität, die DO-254 und anderen Standards entsprechen müssen.</p>