ARM Fast Models

Overview

ARM® Fast Models offer a trusted portfolio of programmer’s view models of the latest ARM IP. The portfolio is fast and accurate with integrated building blocks for virtual platforms, allowing software development to start months before hardware prototypes become available.

Key Features

- Functionally accurate ARM Instruction Set Models, validated against ARM processor designs
- Models advanced ARM technologies such as caches, MMU, LPAE, virtualization, TrustZone®, and VFP
- High-performance programmer’s view models for software development and hardware/software co-verification
- Boots any OS such as Linux, Android™, Windows Embedded CE, and Symbian in seconds
- Suitable for application, firmware, and early driver development
- SystemC TLM 2.0 Export capability of ARM processor-based subsystems
- Integration with EDA solutions from Cadence, Carbon, Mentor Graphics, and Synopsys.

Fast Model Portfolio

The Fast Model Portfolio consists of high performance models for ARM CPUs and ARM CoreLink™ peripherals. It also contains visualization components and the generic parts such as timers and UARTs needed to build Virtual Platforms.

Processor models are developed as part of ARM’s design methodology, guaranteeing early availability and functional fidelity. The portfolio is regularly updated with models of the latest ARM processors aligned with the IP delivery. The portfolio includes:

- Models for ARM Cortex™-A series, Cortex-R series, Cortex-M series, ARM11™ and ARM9™ processors
- Models for ARM CoreLink IP and peripherals to create complete Virtual Platforms
- Ready to run Real-Time System Models (RTSM) of ARM development boards including Versatile™ Express
- Architecture Envelope Model (AEM) for ARMv7-A and ARMv8.

Platform Creation

To complement the models portfolio, the Fast Models provide the tools and interfaces required to create Virtual Platforms. This includes the System Canvas platform creation environment, and the System Generator simulator build system.

System Canvas

System Canvas shortens the design time for creating virtual platforms by providing a powerful and intuitive graphical user interface for entering, editing, and managing platform designs. It also acts as a unified cockpit for various tasks throughout the design, ranging from creating the design in the LISA+ text editor, to the deployment of the final virtual platform or subsystem.

Due to its intuitive block-diagram editing, no programming experience is required to design, configure and build virtual platforms based on the available models from the Fast Model Portfolio.
System Canvas includes:

- Intuitive block diagram editor for platform creation
- LISA+ text editor with syntax highlighting
- C/C++ libraries can be included by writing a LISA+ integration wrapper
- Host level debugging tools are used to debug the Virtual Platform for functional correctness.

System Generator
Platforms created in System Canvas are rendered into simulation models by System Generator. LISA+ components and system descriptions are translated into C++ source code, which is compiled into simulators with standard host software (GCC, Microsoft Visual Studio). System Generator supports the generation of three types of targets:
- A self-contained virtual platform as a standalone executable
- A self-contained virtual platform as a dynamic library
- An extensible sub-system as a SystemC compliant module with TLM-2.0 compliant transaction interfaces.

System Generator embeds the necessary interfaces to enable connection of standard debuggers from ARM and 3rd parties, along with interfaces to enable trace of the models.

Model Export
ARM collaborates with EDA partners in validating interoperability and integration of exported virtual subsystems into their SystemC based design and simulation tools. Integrated solutions are supported by Cadence, Carbon, Mentor Graphics, and Synopsys.

Debug and Trace
Virtual Platforms created in Fast Models embed debug and trace interfaces for use with ARM Development Studio 5 (DS-5™) and a range of 3rd party and proprietary debug solutions. Debug is provided by Cycle Accurate Debug Interface (CADI) and allows run control and inspection capabilities. For trace requirements the Model Trace Interface (MTI) provides an extensible framework for capturing information from the virtual platform. Fast Models includes a model debugger to enable out-of-the-box debug capabilities and an example program trace plug-in.

The Model Debugger allows inspection of any component of a virtual platform, automatically retargeting to the features and capabilities of the target.

Model Debugger is targeted at C/C++ application development on models and includes:
- Standard high level source debugging features like step, step in, step out, step over, and instruction breakpoints
- Disassembly, register, and memory views
- Stack, watch window, and expression evaluation
- Supports multi core debugging.

ARM Processors and System IP

ARM CPU models
- ARM926E-J-S™
- ARM1136J(F)-S™
- ARM1176JZ(F)-S™
- Cortex-M3
- Cortex-M4
- Cortex-R4
- Cortex-R5
- Cortex-R7
- Cortex-A5
- Cortex-A7
- Cortex-A8
- Cortex-A9
- Cortex-A15
- CCI-400
- GIC-400
- DMA-330

Platform Models
- Standard peripherals
- Memories and controllers
- Complex peripherals

Supported platforms:
- RedHat, Linux 4 and 5, 32-bit and 64-bit
- Windows XP 32-bit, and Windows 7, 32-bit and 64-bit

Example projects to build virtual platforms based on Versatile Express for single and multi-core processor configuration, and big.LITTLE processing.

www.arm.com/fastmodels

All brand names or product names are the property of their respective holders. Neither the whole nor any part of the information contained in, or the product described in, this document may be adapted or reproduced in any material form except with the prior written permission of the copyright holder. The product described in this document is subject to continuous developments and improvements. All particulars of the product and its use contained in this document are given in good faith. All warranties implied or expressed, including but not limited to implied warranties of satisfactory quality or fitness for purpose are excluded. This document is intended only to provide information to the reader about the product. To the extent permitted by local laws ARM shall not be liable for any loss or damage arising from the use of any information in this document or any error or omission in such information. Copyright © 2012 ARM Ltd.