Open Virtual Platforms™
Fast Processor Models

Electronics systems are becoming more complex, and the driver of this complexity is software. Software embodies the key features of the products, enabling the differentiation required for new products to succeed. Software development is also on the critical path to an on-time, high quality, successful delivery of a new product.

New products, whether full systems or systems on chips (SoCs), integrate multiple processors and other components. These complex designs make it essential that the software development starts as early as possible in the product design cycle.

Virtual platforms, or virtual prototyping, or software simulation is one technique that is being rapidly adopted by teams developing these leading edge products. Whether for mobile devices, home entertainment products, automotive electronics, or other systems, virtual platforms provide the best solution to early development of software.

Open Virtual Platforms (OVP™) modeling and simulation technology enable the simulation of embedded systems running at 100s of million instructions per second. These simulations are instruction accurate, and can be used to model individual processors, chips, subsystems and complete systems.

Visit **www.OVPworld.org** to download models and simulator

OVP™ Components

OVP has three main components:

- the OVP APIs that enable a C model to be written
- a library of open source processor / peripheral models
- OVPsim™: a simulator that executes these models

Use OVP to create a simulation model of a platform including all components and connect it to your debugger to provide a very efficient fast embedded software development environment.

	openCores OR1K			ARM			MIPS32		
Benchmark	Simulated Instructions	Run time	Simulated MIPS	Simulated Instructions	Run time	Simulated MIPS	Simulated Instructions	Run time	Simulated MIPS
linpack	5,028,517,285	5.59s	900	1,119,997,559	1.21s	926	1,450,948,024	1.03s	1402
Dhrystone	2,118,116,353	1.86s	1139	1,082,061,368	1.42s	763	974,078,238	1.12s	873
Whetstone	10,924,250,068	9.96s	1097	1,324,605,091	1.59s	833	4,265,685,378	2.83s	1510
peakSpeed1	1,700,004,130	0.81s	2109	1,400,001,624	0.83s	1697	1,400,001,382	0.59s	2379
peakSpeed2	7,600,010,989	3.5s	2174	7,600,001,850	4.48s	1695	6,400,001,651	2.84s	2257
	Virage ARC			NEC v850			PowerPC		
Benchmark	Simulated Instructions	Run time	Simulated MIPS	Simulated Instructions	Run time	Simulated MIPS	Simulated Instructions	Run time	Simulated MIPS
linpack	4,184,162,664	5.53s	757	4,991,344,159	7.06s	707	3,163,966,113	3.95s	802
Dhrystone	1,262,082,476	1.56s	809	2,564,132,573	2.66s	965	786,068,243	1.03s	763
Whetstone	7,830,770,799	7.61s	1030	6,202,408,024	7.21s	860	3,515,867,502	3.65s	963
peakSpeed1	1,500,001,786	1.02s	1471	1,200,003,128	0.87s	1374	1,000,002,001	0.65s	1549
peakSpeed2	6,800,002,055	3.91s	1740	6,000,007,540	3.26s	1842	5,200,002,884	2.5s	2078

All measurements on 2.83GHz Intel Core2, OVPsim 20100528.0

OVP™ Performance

As seen in the table above, OVP Fast Processor Models run fast, very fast. 100s and 1,000s of millions of instructions per second. OVPsim, the execution engine for these models, uses state of the art code morphing simulation technology to get this performance. This applies not only to single core processors, but also to the leading edge multicore processors from ARM and MIPS. This industry leading performance is not just available in simple benchmarks. For example, an OVP virtual platform will boot Linux in less than 10 seconds.

SystemC Integration

Many development teams have adopted SystemC for virtual platform behavioral and peripheral components, utilizing IEEE 1666 compliant simulators. All OVP processor core models include a native SystemC TLM-2.0 interface, enabling easy use in SystemC simulation environments. In fact, OVP models have been used with the OSCI simulator, as well as with the SystemC simulators from all the major vendors.

OVP™ Fast Processor Models

The OVP Fast Processor Model library includes the following cores:

ARM
- ARM7
- ARM7TDMI
- ARM920T
- ARM922T
- ARM940T
- ARM926EJS
- ARM946ES
- ARM966ES
- ARM1020E
- ARM1026EJS
- ARM1136J-S
- ARM1156T2-S
- ARM Cortex-M3
- ARM Cortex-M4
- ARM Cortex-A5
- ARM Cortex-A8
- ARM Cortex-A9
- ARM Cortex-A9 MP

Available Q2 2012
- ARM Cortex-A15
- ARM Cortex-R4

MIPS
- MIPS M4K
- MIPS 4KEc
- MIPS 4KEm
- MIPS 4KEp
- MIPS 24Kc
- MIPS 24Kf
- MIPS 24KEc
- MIPS 24KEf
- MIPS 34Kc
- MIPS 34Kf
- MIPS 74Kc
- MIPS 74Kf
- MIPS 1004Kc
- MIPS 1004Kf
- MIPS M14Kc
- MIPS M14Kf
- MIPS 1074Kc
- MIPS 1074Kf

ARC (Synopsys)
- ARC 605
- ARC 6xx
- ARC 7xx

NEC (Renesas)
- V850 ES
- V850 E1
- M16C

Available Q2 2012
- V850 E2

OpenCores
- OR1K

Open Virtual Platforms

Visit **www.OVPworld.org** to download models and simulator

PowerPC
- E200 core (MPC55xx)

Xilinx
- MicroBlaze

Available from other developers
- **Vinchip** VinRZ5110
- **SPARC** v8

OVP™ Fast Processor Models

To get the high speed required for real applications, processor hardware is modeled only to the minimum necessary level for *correct or plausible instruction behavior* so that software cannot tell it is not running on real hardware. Processor core models are instruction accurate. There is no timing information, no cycle approximation. These models are developed to simulate as fast as possible, enabling software developers to use OVP as the early development environment.

Wherever possible, processor core model functionality is verified either by the processor vendor themselves, or using technology and test suites provided by the processor vendor. This is the case for the vast majority of processor core models available from OVP.

Debugging and 3rd Party Tool Interfaces

OVPsim has interfaces to GDB for debugging, and can also be used in an Eclipse environment. In general, any debugger that utilizes the RSP socket for connecting to the processor can be used with an OVP simulation.

OVP virtual platforms can also be used with other tools for tasks such as hardware-software co-verification. In addition to the native SystemC/TLM2.0 interfaces, OVP simulations have been integrated with various hardware design language (HDL) simulators, hardware emulators and FPGA prototype tools.

Benefits of Virtual Platforms for Embedded Software Development

Virtual platforms lower software development costs, increase quality and reduce the risks involved with the software development side of delivering advanced electronic systems. This is accomplished by enabling:

- ✓ Ability to run the real hardware executables on the virtual platform
- ✓ Early start of software development and hardware-software integration
- ✓ Accessibility of the virtual platform for the entire development team, no matter the location
- ✓ Full visibility and controllability of the simulation environment
- ✓ Repeatable, deterministic simulation makes debugging easier
- ✓ Flexibility of the virtual platform to easily accommodate changes in the specification
- ✓ Ability of the virtual platform to connect to real world resources