OCP-IP overview
The Open Core Protocol International Partnership (OCP-IP) is a nonprofit organization delivering the first fully supported, openly licensed core-centric protocol that comprehensively fulfills system-level integration requirements. OCP-IP was announced in December 2001 to promote and support the Open Core Protocol (OCP) as the complete socket standard that ensures rapid creation and integration of interoperable virtual components. The OCP facilitates IP core reusability, reduces design time and risk, and reduces manufacturing costs for SoC designs.

OCP allows designers to build cores independent of specific bus protocols, and of any particular design implementation. This allows easier reuse of OCP-compliant cores across multiple SoC designs. OCP eliminates the need to repeatedly modify the core itself, and preserves the verification and test benches by defining all of the core’s natural interface capabilities. They are therefore presented in an unchanging, universally understood manner.

The OCP-IP Governing Steering Committee participants are:
- Nokia
- Sonics Inc.
- STMicroelectronics
- Texas Instruments
- Toshiba Semiconductor Group

The group rapidly exceeded 100 members including IP companies, integrated device manufacturers, system companies, and design houses. OCP-IP has also initiated strategic alliances with several other industry standards organizations including:
- ECSI
- Si2
- OSCI
- VSIA

VSIA endorses the OCP socket, and OCP-IP is an Adoption Group of the VSI Alliance. The success of OCP is a result of the OCP definition of sockets.

Socket definition
For decades, Local Area Networks (LANs) grappled with the same issues that are now emerging for SoC designers. In the end, LAN designers created well-defined interfaces by defining physical connections and protocols for exchanging information over those physical connections. The appearance of these industry conventions enabled the computing industry to provide independently developed and functionally diverse plug-and-play products that commercial enterprises assembled into highly custom LAN configurations. So, the successful implementation of a widely accepted interface definition is not without precedent.

OCP uses the interface concept of SoC sockets. Ideally, a SoC socket enables core designers to concentrate on their core functionality and the associated interconnects (for example, USB, 802.11b, or SDRAM). Similarly, SoC system integrators should be able to concentrate on SoC timing, core service bandwidth, latency requirements, and final floor-plan design independent of core functionality. The socket would therefore provide the necessary physical and exchange protocol delineation necessary to achieve this well-defined layering.

Core transport independence
To achieve this, note that an ideal SoC socket must be transport implementation agnostic (in effect, not dependent upon a specific bus or other interconnect). SoC cores therefore interface to an inter-core transport mechanism via the interface, but the precise transport mechanics (such as computer-style bus, a cross bar, or a configurable on-chip network) would be unknown to the core. For example, an IP core with an OCP conformant interface is bus independent as shown in Figure 1.
This requirement is essential, or core designs would instantiate transport knowledge within their designs, thereby limiting their reuse in SoC designs that use differing transport mechanics. A transport-unaware approach ensures implementation independence, which allows a system designer to select the optimum interconnect for their system’s needs.

**Dimension independence**
Because of bandwidth requirement diversity, the ideal interface should allow designers to configure interface implementations along various dimensions. For example, these dimensions include the interface data widths required to meet bandwidth requirements, exchange handshake protocols, and exchange acknowledgments. This enables SoC designers to tailor core and SoC designs with minimized complexity and circuit areas, while supporting core and SoC requirements.

**OCP benefits**
The solution to maximizing core reuse potential requires adopting a well-conceived and specified core-centric protocol as the native core interface. By selecting an adopted industry standard, core designers not only ensure core reuse for cores developed within their own enterprise, they also enable reuse outside their enterprise under Intellectual Property (IP) licensing agreements. Finally, they also maximize their ability to license and incorporate third-party IP within their own SoC designs. In other words, they achieve SoC design agility, and the ability to rapidly implement design solutions when licensing IP.

In addition, a rigorous IP core interface specification, combined with an optimized system interconnect, allows core developers to focus on developing core functions. This eliminates the typical advance knowledge requirements regarding potential end-systems that might utilize a core, as well as the other IP cores that might be present in the application(s). Cores simply need a useful interface that decouples them from system requirements. The interface then assumes the attributes of a socket – an attachment interface that is powerful, frugal, and well understood across the industry.

By this methodology, system integrators realize the benefits of partitioning components through layered hardware – designers no longer have to contend with a myriad of diverse core protocols and inter-core delivery strategies. Using a standard IP core interface eliminates the need to adapt each core during each SoC integration, allowing system integrators the otherwise unrealized luxury of focusing on SoC design issues. And, since the cores are truly decoupled from the on-chip interconnect, hence each other, it becomes trivial to exchange one core for another to meet evolving system and market requirements.

In summary, for true core reuse, cores must remain completely untouchable as designers integrate them into any SoC. This can only occur when a change in bus width, bus frequency, or bus electrical loading does not require core modification. In other words, a complete socket insulates cores from the vagaries of, and change to, the SoC interconnect mechanism.

The existence of such a socket enables supporting tool and collateral development for protocol, checkers, models, test benches, and test generators. This allows independent core development that delivers plug-and-play modularity without core interconnect rework. Additionally, this allows core development in parallel with system design thereby saving precious design time.

**Core interface requirements**
Core interface design requirements are very diverse, and no single specific implementation can possibly address all of the requirements. Standardized core interface specification requires:

- Error handling
- Interrupts
- Test
- More than data-flow signaling
- Control and status
- Flags and software flow control
- Scalability across a family of requirements
- Capture all signaling between the core and the system
- Ability for designers to configure specific interface instantiations along a number of dimensions (such as bus width and data handshaking)

**OCP introduction**
OCP is a freely available, bus-independent protocol that supports all core-centric considerations discussed previously. Specifically, it completely captures all of an IP core’s communication requirements. As a highly configurable interface, OCP is not a one-size-fits-all protocol. Rather, it comprises a continuum of protocols that share a common definition.

OCP explicitly supports sideband signals via optional extensions to the basic OCP data set. These sideband signals include: reset, interrupt, error, and control/status information. In addition, a generic flag bus accommodates any unique core signaling needs. An optional OCP test interface extension supports scan, JTAG, and clock control. This enables core debug and manufacturing test when integrated into SoCs.

System designers can therefore tailor a specific OCP configuration to exactly match their core requirements. Through straightforward configuration procedures, OCP supports simple, low-performance cores with very simple and frugal OCP interfaces, while also supporting complex, high-performance cores with more complex interfaces.

An IP developer can therefore complete an IP core design using the OCP interface. No end-application knowledge is required beyond the OCP, allowing complete independence for members of the often global design teams. The system integrator is also free to choose the on-chip interconnect that best suits the system requirements of the application, then effectively wraps that interconnect to present OCP interfaces to the cores.

**Conclusions**
A standard socket core protocol is essential to the SoC design community. OCP is the only complete, fully supported, and proven socket. Adopting OCP avoids incompatible or proprietary solution proliferation, and expands the total available market for commercial and legacy IP cores.

The complete, fully supported core-centric OCP delivers substantial and demonstrable benefits over older style bus-centric protocols. OCP is a core-centric, openly licensed, royalty-free core interface protocol. It does not restrict or otherwise interfere with inherent core capabilities. It is scalable and configurable to match different communication requirements associated with different core and SoC designs.
Cores with OCP interfaces and OCP interconnect systems enable true modular plug-and-play integration, allowing the system integrators to optimally choose cores, and the best application interconnect system. This ensures the designer that the cores and the system can work in parallel, and therefore shorten design times. In addition, not having system logic in the cores allows the cores to be immediately reused with no additional time for core modification and reverification.

Finally, verification and test suites, when written to OCP specifications, are completely portable across multiple designs, rarely requiring even minor adjustments for a particular interface bridge.

**OCP-IP membership benefits**

Website benefits:
- Access to the Members Only website
- Access to Members Discussion Forum
- Free vendor listing
- Free IP and EDA product and design service listing
- E-mail Technical Support

Other benefits:
- Hot line technical support
- Industrial grade tools access
- Free access to all membership products, tools, and services
- Participation in OCP-IP events, work groups, and member meetings
- Ability to associate with key industry leaders in the EDA/SoC community
- The ability to contribute to OCP enhancements and access specifications in both draft and adopted form
- Free training material and tutorials for both the OCP Standard and industrial grade checkers and software tools

Membership applications, and the OCP Specifications are available at the website shown below.

For more information, contact Ian at:

**OCP International Partnership**
5440 SW Westgate Drive, Suite 217
Portland, Oregon 97221 USA
Tel: 503-291-2560 • Fax: 503-297-1090
E-mail: admin@ocpip.org • Website: www.ocpip.org

---

**Advancing Transaction Level Modeling: Linking the OSCI and OCP-IP Worlds at Transaction Level**

*By James A. Colgan, Sonics Inc. and Pete Hardee, CoWare, Inc.*

**Abstract:** The growing need for Transaction Level Modeling (TLM) standards that can link together SoC architecture and software development at levels of abstractions higher than RTL has stimulated CoWare and Sonics to collaborate on a cohesive methodology that addresses both SoC designer and software developer needs. This paper outlines the two prevalent industry approaches from OCP-IP (Open Core Protocol – International Partnership) and OSCI (Open SystemC Initiative), and then describes how two founders of both of these organizations are collaborating to provide real solutions to the industry. CoWare is an OCP-IP Sponsor member and founder of OSCI. Sonics is a co-founder of OCP-IP and contributor to the OCP-IP System Level Design Working Group.

Available at: [www.opensystems-publishing.com/whitepapers](http://www.opensystems-publishing.com/whitepapers)