OCP International Partnership

Open Core Protocol

Overview

The Open Core Protocol (OCP) establishes the first *openly licensed, core-centric protocol* to meet contemporary system-level integration challenges. OCP comprehensively defines an efficient, bus-independent, configurable and highly scalable interface for on-chip subsystem communications. With broad industry support and collaboration, OCP International Partnership (OCP-IP) now offers the 3.0 version specification that further extends capabilities in increasingly important areas such as system-level cache coherency and power management. OCP data transfer models range from simple request-grant handshaking through pipelined request-response to complex out-of-order operations.

Legacy IP cores are readily adapted to OCP, while new implementations may take full advantage of advanced features: designers select only those features and signals encompassing a core's specific data, control and test configuration. Core definition using OCP encapsulates a complete system integration description enabling *core and test bench reuse without rework*. Not only does OCP provide clear delineation of design responsibilities for core authors and System-on-Chip (SoC) integrators, but also institutes a key partitioning formalism for verification engineers and automation software.

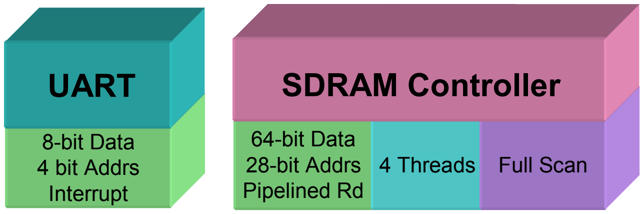
Highlights

The OCP promotes IP core reusability and reduces design time, design risk and manufacturing costs for SoC designs. It focuses exclusively on IP core interfacing without pre-empting interconnect topology or other application-specific integration choices.

- Enables IP core creation to be independent of system architecture and application domain
- Describes all inter-core communications
- Optimizes die area by configuring into the OCP interface only those features needed by the core
- Specified timing categories assure core interoperability
- Facilitates rapid, plug-and-play IP integration

Advantages

- The *de facto* open standard with industry wide support
- Eliminates the ongoing task of interface protocol (re)definition, verification, documentation and support
- Readily adapts to support new core capabilities
- Testbench portability simplifies (re)verification
- Limits test suite modifications for core enhancements
- Interfaces to any bus structure or on-chip network
- Delivers industry-standard flexibility and reuse
- Point-to-point protocol can directly interface two cores



Complete spectrum of core signaling can be handled by a single protocol – the OCP

Capabilities

The OCP captures all core characteristics without restricting system arbitration, address map, etc.

- Small set of mandatory signals, with a wide range of optional signals
- Synchronous, unidirectional signaling allows simplified implementation, integration and timing analysis
- Configurable address and data word width
- Structured method for inclusion of sideband signals: high-level flow control, interrupts, power control, device configuration registers, test modes, etc.
- Transfers may be *pipelined* to any depth for increased throughput
- Optional burst transfers for higher efficiency
- Multiple concurrent transfers use *thread identifiers* for out-of-order completion
- *Connection identifiers* provide end-to-end traffic identification for differential quality of service, etc.
- Synchronization primitives include atomic test-set, lazy synchronization, non-posted write commands
- System-wide hardware cache coherence enabled through additional coherence transactions
- Optional core disconnect scheme to support power management techniques

The Open Core Protocol Specification is available at:
www.ocpip.org

Open Core Protocol key features

Basic OCP interoperability

- Master/slave interface with unidirectional signals
- Driven and sampled by the rising edge of the OCP clock
- Fully synchronous, no multi-cycle timing paths
- All signals are strictly point-to-point (except clock and reset)
- Simple request/acknowledge protocol
  - Supports data transfer on every clock cycle
  - Allows master or slave to control transfer rate
- Core-specific data and address bus definition including:
  - Byte and non-byte-oriented data busses
  - Read-only and write-only interfaces
  - In-band data tagging (parity, EDC, etc.)
  - In-band command tagging (protocol extensions, etc.)
- Pipelined or blocking commands including non-posted write
- Security access permissions can be part of any request
- Well-defined linguistic format for core characteristics, interfaces (signals, timing and configuration) and performance
- Timing category specifications
  - Level 2 - tightest, highest performance interface timing
  - Level 1 - conservative timing for effortless integration
  - Level 0 - protocol without timing specified (especially useful for simulation/verification tools)

Simple Extensions performance

- Bursts group related transfers into a complete transaction
- Burst transactions supported:
  - Sequential (precise or indefinite length)
  - Streaming (e.g. FIFO)
  - Core-specific (e.g. cache lines)
  - Controlled atomicity in breaking up long bursts
  - Two-dimensional block sequences
- Pipelined (cmd/add ahead of data) writes
- Aligned or random byte enable orders
- Read and write data flow control
- Address space definition for multi-address-segment targets
- Single-request/multiple-data or one command per data phase

Sideband Extensions custom signaling

- Core-specific, user defined signals:
  - System event signals (e.g. interrupts, error notification)
  - Two synchronous resets defined: master-to-slave and/or slave-to-master
  - Data transfer coordination (e.g. high-level flow control)
- Debug and Test Interface Extensions
  - Support structured full or partial scan test environments
  - Scan pertains to internal scan techniques for a pre-designed hard-core or end user-inserted into a soft-core
  - Clock Controls are used for scan testing and debug, including multiple clock domains
  - IEEE 1149 supports cores with a JTAG Test Access Port
  - JTAG- and Enhanced-JTAG-based debug for MIPS®, ARM®, TI® DSP, SPARC™ and others

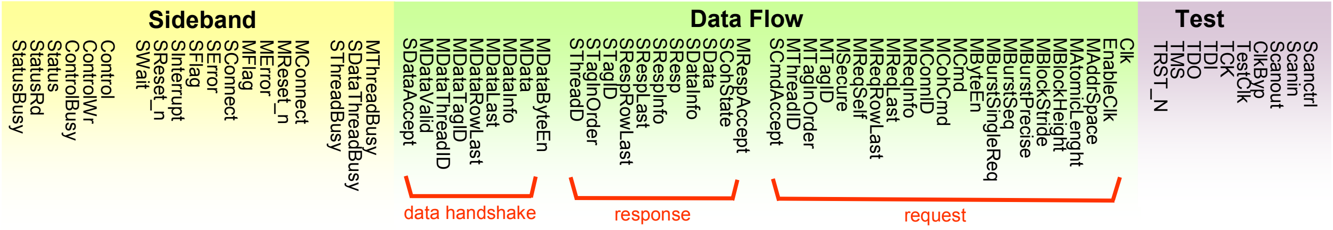
Complex Extensions concurrency support

- Thread identifiers enable:
  - Interleaved burst transactions
  - Out-of-order transaction completion
  - Differential quality of service
- Rigorous thread flow control definition guarantees non-blocking
- Connection identifiers enable:
  - End-to-end system initiator identification
  - Service priority management by system targets
- Tagging provides shared flow control for out-of-order transactions

CoreCreator®

OCP-IP offers its members the use of an EDA tool, CoreCreator II, to automate the tasks of building, simulating, verifying and packaging OCP-compatible cores. CoreCreator II also includes Verification IP to generate and respond to OCP stimulus, and a protocol checker to ensure compliance with the OCP specification. IP core products can be fully componentized by consolidating core models, timing parameters, synthesis scripts, verification suites and test vectors.

Complete Set of OCP Signals



About OCP-IP®

Formed in 2001, OCP-IP is a non-profit corporation promoting, supporting and delivering the only openly licensed, core-centric protocol comprehensively fulfilling integration requirements of heterogeneous multicore systems. The Open Core Protocol (OCP) facilitates IP core reusability and reduces design time, risk, and manufacturing costs for all SoC and electronic designs by providing a comprehensive supporting infrastructure. For additional background and membership information, visit www.OCPIP.org.

OCP-IP Association, Inc. 3855 SW 153rd Drive, Beaverton, OR 97006 USA
Tel: 1-503-619-0560 Fax: 1-503-644-6708 E-mail: admin@ocpip.org URL: www.ocpip.org