Implementing Reusable IP: The Repository & The Library

by

Thomas Zimmermann
Peter Schindler
System on a Chip Design Technology Division
Motorola Semiconductor Products Sector

Today's system designers are utilizing System-on-Chip (SoC) design methods to meet ever shorter design cycles demanded by the marketplace. Reuse of Intellectual Property (IP) has emerged as one of the key strategies available to both IP providers and their corresponding IP consumers. Studies of reuse have clearly shown its value, particularly when the IP was originally created for reuse. The Motorola Semiconductor Products Sector (SPS) has implemented sector wide Semiconductor Reuse Standards (SRS) that follow the semiconductor industry's Virtual Socket Interface Alliance (VSIA) structure.

The IP Repository lies at the core of the IP Reuse Infrastructure. It consists of a centralized meta data (search criteria) database and distributed vaults maintained by each local IP provider organization. A librarian, local to each provider organization, provides the interface between Provider and the Repository. The Web enabled Reuse Infrastructure serves both IP consumers and IP providers. The IP Repository creates an interface between Providers who have created reusable IP and Consumers who want to obtain and reuse IP blocks. The Web enabled feature allows both consumers and providers to search, select, upload, and download IP on demand as appropriate.

IP Providers require only a connection to the Internet, a standard Web browser, and SRS compliant IP which they desire to make available to the public. Each organization that participates in the IP Repository selects its librarian The Consumer requires only a connection to the Internet, a standard Web browser, and an SoC requirement for reusable IP.

Market Drivers

SoC delivers specific solutions for targeted applications. Each SoC design contains an entire system. An SoC design typically includes processors, embedded memory to store system programs and data, communication peripherals, and analog interfaces. SoC products, to be complete, require embedded software. An SoC could be a general purpose IC differentiated only by software such as device drivers, real-time operating systems, application programming interfaces, and application software. All of these various types of IP can be found in an effective Repository.

Both chip manufacturers (IP providers) and systems developers (IP consumers) have recognized the need to implement reuse of their semiconductor designs. Ultimately, industry participants will cooperate to implement IP reuse standards (such as VSIA) and incentivize providers to deposit the IP in an industry-wide Repository. A reuse standard, such as Motorola's SRS, is a necessary precursor for widespread and effective implementation of reuse.

IP created solely for a targeted application requires additional design effort so it can be integrated and reused by other system integrators (IP consumers). Reusable IP design requires a well-documented, parameterized general solution at a high level of abstraction. The reusable IP can then be easily adapted to meet various functional and process technology requirements in subsequent applications.

Once created, the reusable design must be made publicly available to the IP consumer in an IP Repository. With such a repository in place, it's conceivable that in a platform-based design with 18 different modules, existing IP could be reused in as many as 15 or 16 modules. Actual new design may involve only a few modules.

Many local organizations have successfully implemented modest levels of reuse. For IP providers, the concept is simple. By investing slightly more in upfront development costs to make an IP block conform to reuse standards, IP providers can optimize return on investment. For IP consumers faced with short design cycle times, it makes sense to search out existing, available, and reusable IP design blocks. There's no profit for a system integrator who supports a "Not Invented Here" syndrome among the design team.

Optimized reuse conditions means that IP created for reuse and compliant with reuse standards is deposited in an effective IP Repository. The deposited IP must be supported with an extensive database of search criteria (meta data). In a recent industry study by Collett International, current non-optimized reuse requires 41% of the original design effort to implement. This study carefully investigated reuse implementation under optimized conditions and found that reuse implementation could be reduced to as little as 7% of the original design effort. Thus, the need for the core IP infrastructure, the IP Repository.

Repository Architecture

The IP Repository's client-server architecture is composed of the Motorola Intranet, the IP data vaults, an IP meta data database server hosting an Oracle relational database, a web application server, a configuration management system (CM), and the Web-based Graphical User Interface (GUI). As seen in overview, this architecture results in a distributed IP Vault network. Each local IP provider maintains local access and control of their IP. However, to the IP consumer, all of the data provided by the various organizations participating in the repository are available for search seamlessly.

The Oracle relational database drives the system. It stores all the meta data, which serves as the foundation for IP consumers to search for their required IP. All of the IP data is stored in IP vault servers. This database also controls access to the IP vaults. The database accesses the vaults only for uploading new or revised IP or for an IP consumer to download selected IP. When an IP consumer undertakes a search for IP in the system, the database also generates the dynamic contents of consecutive web pages, which forms the graphical user interface. From the interface, a web application server provides the communications link to the IP consumer in the form of the search results delivered on a standard browser.

The multiple vault servers contain the actual IP and use a configuration management (CM) system to support IP version control. The vault server can store IP in any required data structure, thus different types of IP can be stored on the server. Meta data for each IP in the vault provides the search parameters by which IP consumers can locate their desired IP. The IP Repository's function centers on the relational database's interaction with the meta data for each IP deposited in the vaults. Without the meta data properly and accurately entered in the database, the IP consumer would be helpless to locate a desired block of IP.

Each IP module is categorized. For example, it might be a DMA controller, a memory IP, a documentation IP, or a software IP. For each IP library, the librarian maintains a list of allowed IP categories or types. These IP types are assembled into hierarchies, called "taxonomy trees". An IP consumer can use each level of the hierarchy as the starting point for a data query. The query will traverse down the whole tree hierarchy and retrieve all IP found.

Fortunately, the librarian need not be a database expert. It is far more crucial that the librarian understand the design particulars for her or his local organization. To facilitate a librarian with local design expertise, the IP Repository system developers designed a Java based GUI Module Librarian Interface (MLI). Each librarian uses the MLI to create, update, or maintain meta data for all new, existing, and revised IP. The librarian can do this remotely, but the meta data is stored centrally. This methodology in the system always ensures data integrity.

The Librarian and the IP Repository

The IP Repository provides for the storage and retrieval for reusable IP design blocks. Although conceived primarily for silicon IP, IP can also include documentation files, driver application software, or other complex data structures. Because of the flexibility of the architecture, the IP Repository is generic and can be configured to store and retrieve different types of data. The primary purpose of the IP Repository is to facilitate, support, and enable an efficient System-on-Chip design and integration process to reduce development cycle time.

The process begins with an IP provider who owns one or more design blocks that comply with the SRS. The designer uploads the IP to the librarian who loads it into the IP vault and IP meta data database. The meta data database consists of a list of properties provided by the IP creator. Properties include the IP provider's name, organization, and technology. Additional properties identify the IP block by functionality, performance, and quality. The librarian monitors each incoming IP block to assure the quality of the meta data.

IP meta data is classified into two types, static and dynamic. Static fields include mandatory standard information required for all IP such as name, functional description, relationships to other IP components, status, or revision information. Dynamic fields are represented by IP Properties and their respective values include such fields as Target Library, Reuse Count, Prior Uses, Transistor Count, Maximum Speed, etc. The dynamic fields assigned to each design block depend on the IP type.

The librarian must maintain and update all fields. For instance, one key property, the Reuse Count, reflects the quality of an individual IP design block. The Reuse Count tells a prospective IP consumer how often a design block has been reused on a chip. Presumably, the higher the count, the better the design quality of that IP block. The librarian must also manage the Associated Change Request system. This feedback device incorporates IP consumer feedback into the meta data database. Thus, a consumer can note prior problems encountered during implementation and any requests for changes to the IP.

Meta data is maintained and stored in an Oracle relational database. The database operates on a client-server architecture. Each participating organization worldwide selects its own local librarian. Perhaps it's a fact of human nature, but IP creators do like to maintain local control of their reusable IP. This essential system feature provides a comfort factor for the IP provider and facilitates participation by qualified provider organizations.

The librarian for each IP provider accesses, maintains, creates, and updates meta data in the database using the Module Librarian Interface (MLI), the GUI interface mentioned above. Each organization that participates in the IP repository selects a librarian with design expertise in that organization's IP. Thus, the librarian can intelligently interact with the IP creator(s) in his or her own organization to create an optimized properties list for the meta data database.

Uploading an IP to the IP Repository

Ideally, public access to an IP block begins when the provider submits reusable, SRS compliant IP data and the accompanying meta data to the librarian. In practice, however, a block of IP designed for reuse requires several months to be certified as reusable. Therefore, the librarian can upload IP into the system that has not been certified as SRS compliant. This IP receives an "uncertified" tag (meta data) clearly visible to the consumer. The consumer knows that the IP is not certified yet and may not even be fully compliant for reuse. An uncertified upload avoids the certification bottle neck which would prevent reuse of a design block, even if it is not fully compliant. Although not necessarily a rigorous application of the SRS standard, this pragmatic strategy makes a lot of sense and operates well in practice. The combination of the IP data and the IP meta data constitutes a complete entry into the IP Repository. The IP provider is responsible for submitting the IP data in the correct directory structure class.

When the librarian receives the IP data from the provider, the librarian checks to verify that the meta data is included with the IP data. The librarian also applies a set of tools packaged as part of the MLI which performs some SRS compliance checking prior to upload. These "sanity" checks make sure that the data structure is correct and that all required elements of the IP are present. The librarian does not check any code coverage. After completing these checks satisfactorily, the librarian uploads the IP data to the appropriate IP vault.

The librarian also loads the corresponding properties, description, list of key words, contact person or organization, and any view properties into the IP Repository's meta data database. Once the meta data has been added to the database, the librarian changes the access control data to make the IP accessible to IP consumers. Then the librarian performs one more check to verify that both the IP data and its meta data uploaded correctly.

IP consumers can expect IP from three different kinds of sources in the Repository; star IP, standards-based or commodity IP, and market specific IP. Examples of star IP include vendor-specific embedded processors such as Motorola's M*Core, Star*Core, Coldfire, DSP, or PowerPC processors. Standards-based IP examples include Ethernet, Firewire, IRDA, USB Ports, and serial interfaces. Market specific IP examples include MPEG, VoCoder, FLEX (pager protocols), and CDMA/TDMA (cell phone protocols).

Downloading IP from the IP Repository

The IP Consumer first must have an SoC design need for reusable IP. Using appropriate search criteria in a standard web browser, the consumer searches the IP meta data in the massive global database. Using its powerful web based search and distribution capabilities, the relational database locates, selects, and retrieves the desired IP.

The IP Consumer can combine different search methods to locate the desired design block. While conducting the search, the user can initiate a keyword and category (taxonomy) based search. Or the IP Consumer can implement filtering capabilities based on pre-selected meta data to locate the requested design block. Criteria could include a high level functional description or the maximum speed of a requested IP. In response to the search request, the browser queries the meta data database server and dynamically displays the search result on the IP Consumer's web browser.

Once the IP Consumer has selected a candidate for download, the consumer simply downloads via the web browser. The system, through the relational database, establishes a connection to the associated IP vault to initiate the download. This operation is transparent to the user. The IP Consumer doesn't know where the actual IP is stored and has no need for direct access to the storage vault. This also provides a security level to protect the IP from unnecessary or inappropriate access. The IP vault server completes the download process by distributing the IP data via a defined secure protocol to the IP consumer. The consumer can then make the downloaded IP available to the target design system for SoC integration and post-processing.

Summary

Implementing an efficient reuse structure offers the opportunity to drive the cost of reusing IP down from the current levels of 41% to 7%. With system developers (IP consumers) already under increasing pressure from shorter cycle times, the push for efficient reuse becomes critical to achieve cost-effectiveness. The IP Repository described here provides an infrastructure for reuse, a large step towards the promised land of efficient, reusable IP.

The IP Repository features a Web enabled interface so IP providers can upload reusable design blocks and IP consumers can search, select, and download their desired IP. In effect, the IP Repository is a central, one-stop shopping location developed to store and share IP across the globe via the Internet.

# # #

Word Count: 2490