

# El diseño basado en modelos acelera la verificación funcional

El uso del diseño basado en modelos permite que el equipo de desarrollo recorra el proceso sistemáticamente. También minimiza posteriores quebraderos de cabeza con relación a la verificación.

David Lidrbauch, RTC Group

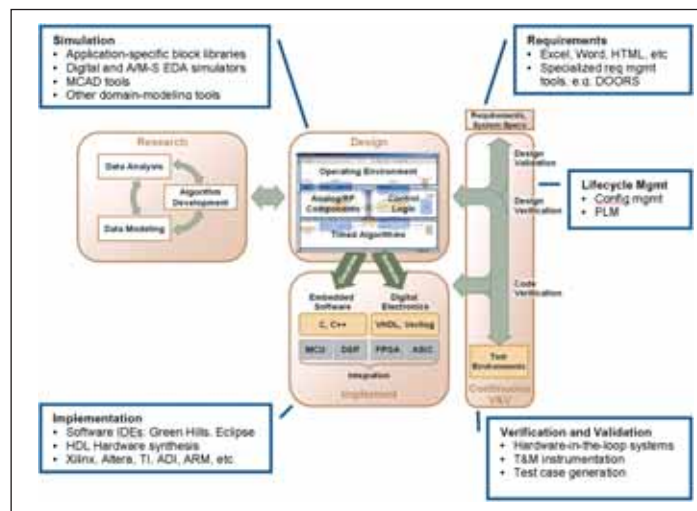


Fig. 1

La verificación funcional consume entre un 50% y un 65% del tiempo de desarrollo y del presupuesto de los proyectos de tecnología SoC (del inglés system-on-chip, sistemas en un chip). Con herramientas fragmentadas y metodologías de desarrollo basadas en el scripting manual, colecciones de herramientas de verificación dispares, incompatibilidades entre herramientas con lenguajes de transacciones casi estándares y fuentes de bases de datos no coincidentes, la verificación funcional puede convertirse en un proyecto que desvíe la atención de los objetivos de desarrollo principales. En lugar de optimizar y depurar los diseños de producto, los ingenieros pierden tiempo depurando su flujo de trabajo y compensando las lagunas existentes en la cadena de su herramienta de verificación.

La verificación funcional de nivel de chip debe examinarse desde una perspectiva más amplia. Gary Smith, vicepresidente de Gartner Dataquest, identificó este problema de forma clara y concisa en su artículo antes del DAC de 2006: "It's the software, stupid". Él y Daya Nadamuni, vicepresidente de investigación de Gartner Dataquest, resaltaron la función crítica e integral del software en los proyectos de SoC de hoy en día. "Puedes tener elegantes algoritmos, un silicio de primer paso y una propiedad intelectual de lujo", afirmó Nadamuni, pero sin software, el producto no va a ninguna parte".[1]

La mayoría de los dispositivos portátiles actua-

les ilustran esa perspectiva más amplia, como lo hacen también los desafíos de desarrollo en el ámbito de la radio definida por software (SDR) y sus hermanas: el sistema conjunto de radio táctica (JTRS) y la radio cognitiva. Debido a la inteligencia de reconfiguración in situ que demandan estos sistemas [2], se podrían llamar la "tormenta perfecta" de los retos del diseño de hardware o software en colaboración.

La pregunta que hay que formular es: "¿Es cada tarea un gasto o una inversión?"

## Diseño basado en modelos

En el diseño basado en modelos, el modelo de sistema es una especificación ejecutable que se convierte en el patrón oro de las actividades de diseño y verificación. Durante la fase de diseño, los distintos equipos de diseño, incluidos los ingenieros de sistemas, software y hardware, comparten este modelo. Cada equipo usa el modelo para elaborar sus componentes, al tiempo que garantiza que el objetivo del diseño se mantenga. La figura 1 muestra un flujo de trabajo basado en modelos típico.

El diseño basado en modelos es decisivo a la hora de combinar los esfuerzos de la verificación y el diseño. El modelo del sistema y su entorno operativo, menos el componente que se está diseñando, se convierten en banco de pruebas y entorno de análisis de cada componente. Este enfoque compensa la dificultad de verificar los

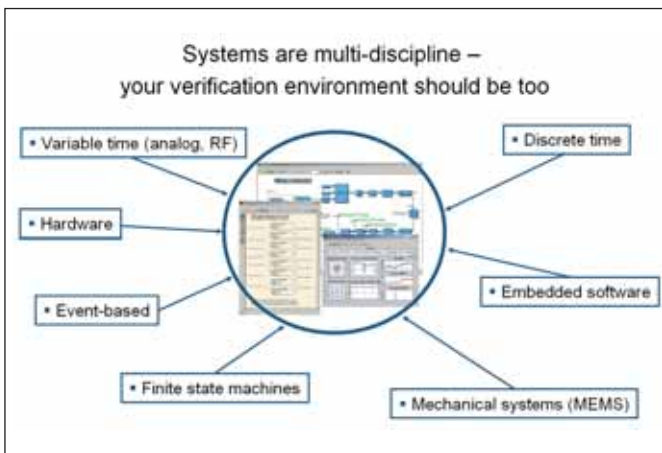


Fig. 2

principales sistemas del mercado que carecen de interfaces bien definidas basadas en estándares entre los componentes internos y los subsistemas. Asimismo, elimina el esfuerzo improductivo que con frecuencia se consume para crear pruebas para bloques e interfaces por separado.

Un ingeniero que verifica HDL o software embebido suele preguntarse: “¿Será mi implementación lo suficientemente buena?” La pregunta crítica que le sigue debe ser: “¿Qué es suficientemente bueno?” La respuesta a esta última pregunta ya se encuentra en la propia especificación de algoritmos creada por los ingenieros de algoritmos o sistemas.

Con el uso de la co-simulación, el equipo de verificación utiliza esa especificación ejecutable como marco de pruebas para garantizar la correcta funcionalidad del sistema. La co-simulación permite a los ingenieros de algoritmos, así como a los ingenieros de hardware y software trabajar en sus propios entornos especializados, sin embargo deja claro a todo el equipo sus decisiones de diseño e implementación. El esfuerzo original relacionado con el diseño de la especificación de algoritmos ejecutable pasa de ser un gasto a ser una inversión que se rentabiliza continuamente.

### Adopción gradual del diseño basado en modelos: verificación de bloques de componentes

Para lograr una gran productividad inicial, el desarrollo en el nivel de bloques de componentes puede llevarse a cabo de forma independiente y paralela. Los algoritmos pueden examinarse y optimizarse por separado, aunque las dependencias entre componentes quedarán sin probar.

Los enfoques del tipo “divide y vencerás” funcionaban bien en el pasado para la verificación funcional porque los elementos fundamentales de los sistemas estaban separados de forma natural y eran independientes. Por el contrario, debido a la convergencia funcional propia de los productos actuales, los bloques analógicos, digitales, de interfaz y de procesador en los SoC presentan una interdependencia extrema. La veri-

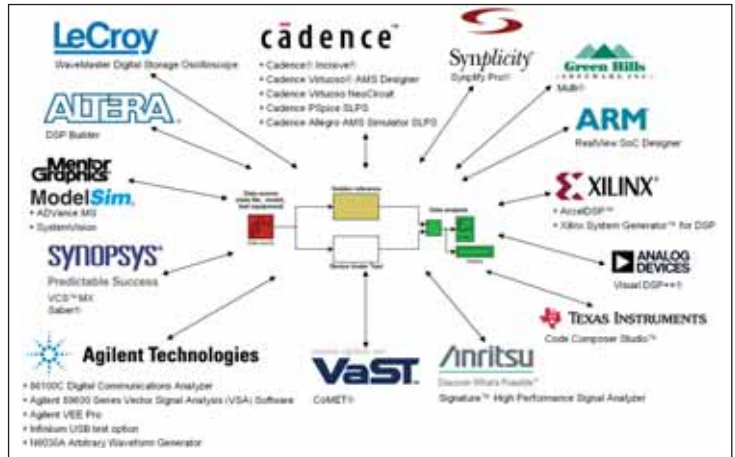


Fig. 3

ficación íntegra de cualquiera de los bloques requiere explorar por completo sus funciones independientes y sus interacciones con otras partes del diseño. Es en la verificación de las interacciones que atraviesan las fronteras de los dominios donde falla el enfoque “divide y vencerás”. La figura 2 muestra algunas de las dificultades en cuestión.

### Verificación del diseño en el contexto del sistema

Ninguna parte de la verificación en el nivel de bloques de componentes puede examinar las interfaces y dependencias entre los distintos dominios y desvelar problemas que pudieran aparecer una vez que el producto se encuentre en uso real. La fusión de los bloques de algoritmos con el resto del sistema es decisiva para comprobar el diseño.

Broadcom usó el diseño basado en modelos para acelerar el desarrollo de la gama SPINNER de sus productos semiconductores WCDMA para dispositivos móviles 3G y simplificó así el reto de convertir la complejidad total del sistema WCDMA en un elemento de hardware que funcionase. Broadcom utilizó MATLAB y Simulink para modelar varios subsistemas, incluidos la estación base, un modelo de canales inalámbricos y el microteléfono. Cada uno de los subsistemas sirvió de marco de pruebas para los demás. Fueron capaces de parcelar rápidamente el modelo de sistema en subsistemas de la arquitectura para permitir que los diseñadores del hardware implementaran cada parte con mayor facilidad.

La evaluación de la implementación óptima de los fragmentos de un sistema suele funcionar mejor después de haber definido y validado los algoritmos. Puede ser necesario explorar la arquitectura del sistema para identificar la implementación de destino más adecuada.

Por ejemplo, en la Rice University de Houston, Brogioli, Radosavljevic y Cavallaro exploraron las ventajas e inconvenientes de la parcelación en una implementación de DSP/FPGA para teléfonos móviles 3,5G HSPDA y detectaron mejoras del 90% con la parcelación apropiada. [3]

Este proceso de parcelación e implementación a partir del modelo de sistema puede acelerarse espectacularmente. Se puede usar la generación automática de código para generar el código embebido del procesador y el HDL, así como el banco de pruebas para la FPGA. Los enlaces de co-simulación cierran el circuito de verificación para comparar cada implementación con el modelo de sistema de referencia. El modelo de sistema original con el código de análisis desarrollado, las métricas del sistema y otras pruebas pueden reutilizarse como plataforma de verificación.

### Combinación de niveles de abstracción para mayor precisión e integridad

Rara vez el diseño discurre linealmente desde las abstracciones de alto nivel hacia representaciones cada vez más detalladas de todo el diseño. El diseño basado en modelos facilita la combinación de niveles de detalle a voluntad, lo cual agiliza la capacidad de mejorar los detalles reales de la simulación funcional. Ello es debido a que es posible elegir el nivel de abstracción adecuado a cada necesidad con el fin de mantener unas velocidades de simulación de nivel de sistema muy rápidas, al tiempo que se somete a prueba el bloque para verificar su funcionalidad y comprobar las opciones de implementación en el contexto de todo el sistema.

En el artículo titulado "DVB-T System Analysis using System Level Co-simulation," [4] los autores describen cómo la interfaz de co-simulación entre MATLAB, Simulink y Cadence Virtuoso AMS Designer ofrece al diseñador de sistemas una gama más amplia de combinaciones de niveles de abstracción al integrar el diseño del sistema con la simulación de señales mixtas de Virtuoso AMS Designer, lo cual también aporta unos ciclos más rápidos de obtención de resultados y una mejor perspectiva del diseño. Asimismo, describen la dificultad inherente a los modernos flujos de diseño no basado en modelos: "Cada herramienta está optimizada para un nivel de abstracción y un área de aplicación concretos. Aunque existe un cierto solapamiento entre las herramientas, aparecen dificultades cuando se hace necesario analizar efectos que abarcan distintos niveles".

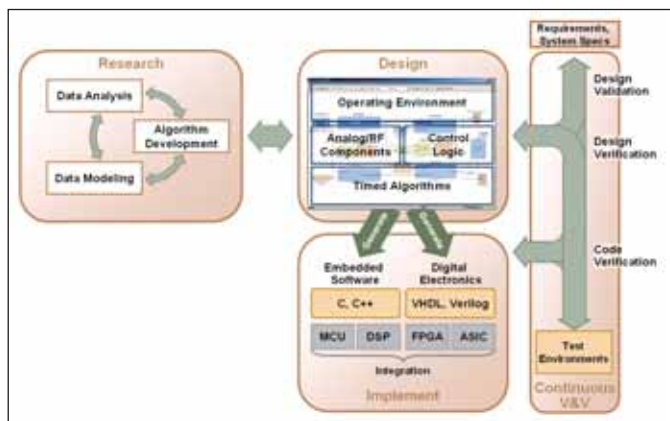


Fig. 4

### El ROI del diseño basado en modelos se multiplica si se aplica a todos los equipos y proyectos

El uso del diseño basado en modelos permite que el equipo de desarrollo recorra el proceso sistemáticamente. El ROI del diseño basado en modelos aumenta sustancialmente si las habilidades especializadas de cada uno de los miembros del equipo se aprovechan durante la verificación de componentes y sistemas por parte de los demás equipos. Este aprovechamiento tiene lugar tanto de forma descendente por el flujo hacia la implementación y la verificación final como de forma ascendente para lograr la mejora del anterior examen del diseño; en definitiva, funciona con el fin de progresar en la comprensión de todas las etapas de desarrollo. El uso conjunto y la reutilización de un modelo de sistema común evita desajustes en el proceso de verificación funcional, mejora la calidad del producto final, reduce los esfuerzos duplicados y malgastados, y perfecciona la totalidad del proceso de desarrollo. El diseño basado en modelos se puede adoptar de manera gradual y las ventajas máximas se logran en el momento en que todos sus elementos se incorporan a todo el proyecto. De este modo, el diseño basado en modelos emerge como inversión que ofrece ventajas en cualquiera de los aspectos del desarrollo de sistemas.

#### Referencias:

- [1] "Dataquest to EDA: 'It's the software, stupid'", Rich Goering, EE Times, 24 de julio de 2006.
- [2] "Next-Generation Design Issues in Communications", Bruce Fette, Mieczyslaw M. Kokar, Mark Cummings. Portable Design, marzo de 2008, pág. 20.
- [3] "Hardware/Software Co-design Methodology and DSP/FPGA Partitioning: A Case Study for Meeting Real-Time Processing Deadlines in 3.5G Mobile Receivers.", Michael Brogioli, Predrag Radosavljevic and Joseph R. Cavallaro. Rice University. <http://www.ece.rice.edu/~rpredrag/doc/MWCAS.pdf>.
- [4] "DVB-T System Analysis using System Level Co-simulation". Ponencia del congreso CDNLive! de 2008. Múnich, Alemania, 28 a 30 de abril. Hans-Werner Groh - Atmel Germany GmbH, Walter Hartong - Cadence Design Systems GmbH, Uwe Eichler - Fraunhofer Institute for Integrated Circuits.

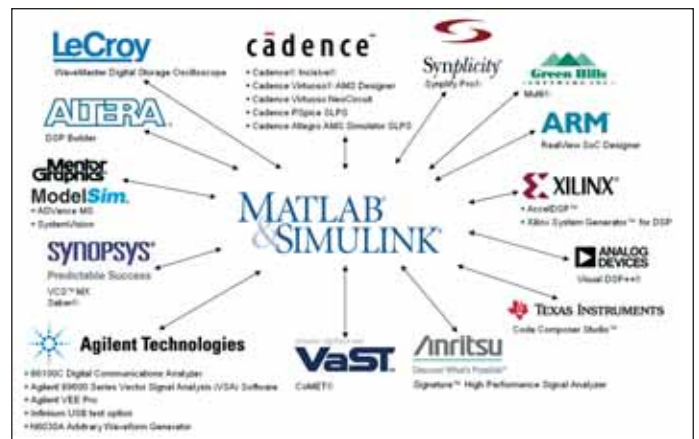


Fig. 5