

INTRODUCCIÓN

Los **HDLs** o *Hardware Description Languages* surgieron en paralelo con las herramientas de captura de esquemáticos. Sus orígenes se remontan a los primeros años de la década de 1970, tratando de resolver algunos de los problemas que se presentaban al modelar circuitos electrónicos con software tradicional que, por sus características y ejecución secuencial, no permitía considerar aspectos fundamentales en cualquier circuito electrónico como el de la concurrencia de señales. Así pues, una de las características básicas de los **HDLs** es modelar o describir circuitos y las señales presentes en ellos de manera concurrente, sin que el orden de las sentencias tenga trascendencia en la descripción.

Al final de los años 70, para progresar en todos los aspectos relacionados con el desarrollo y fabricación de circuitos integrados de tecnología avanzada, el Gobierno de los EE.UU. inició un programa denominado VHSIC (*Very High Speed Integrated Circuit*) en el que se puso de manifiesto la necesidad de disponer de un nuevo lenguaje para definir o describir y simular los dispositivos, ya que las herramientas de entonces trabajaban a nivel de puertas, siendo, por tanto, inadecuadas para su aplicación a diseños en los que se contemplaban cientos de miles de puertas.

El desarrollo del VHDL (*VHSIC Hardware Description Language*), junto con sus requerimientos básicos, se plantean en 1981 dentro de una conferencia de trabajo del programa VHSIC, en la que colaboran representantes del Gobierno, Universidades e industrias representativas del sector. El proyecto es iniciado en 1983 por el Gobierno de los EE.UU. al emitir la solicitud de propuestas para desarrollo del lenguaje y de las herramientas necesarias para su aplicación. En 1986, después de varias revisiones, el IEEE asume la tarea de revisar, modificar y estandarizar el VHDL en base a una aceptación amplia y consensuada por los grupos interesados en el lenguaje. El lanzamiento en Diciembre de 1987 como estándar **IEEE 1076** fue la base para su adopción por los principales fabricantes de sistemas y herramientas CAD. El posterior desarrollo de herramientas para síntesis basadas en el VHDL, unido a su capacidad de descripción jerárquica desde el nivel de sistemas hasta el de puertas y su progresiva aceptación han convertido al VHDL en un estándar *de facto* a nivel mundial, cubriendo desde los aspectos de descripción y simulación, para los que inicialmente se pensó, hasta los relacionados con síntesis y desarrollo de sistemas basados en ASICs - *Application Specific Integrated Circuits* - y FPGAs - *Field Programmable Gate Arrays* .

El VHDL es considerado, más que un lenguaje, un conjunto de recursos con los que cada usuario puede construirse un lenguaje a su medida. En efecto, la posibilidad de modelar circuitos con niveles de abstracción que van desde el nivel de puertas a niveles algorítmicos, la definición de tipos, operadores y subprogramas propios, el uso de empaquetamientos que también definen los usuarios a su medida, etc. hacen del VHDL un excelente lenguaje para la descripción hardware, que efectivamente, en muchas aplicaciones se usa con limitación de sus recursos, o soportando solo una parte de sus sentencias o

posibilidades de modelado. Este es el caso, por ejemplo, del *Boundary Scan Description Language* -BSDL- subconjunto del VHDL utilizado por numerosos fabricantes de dispositivos para descripción de sus componentes en relación con el standard **IEEE 1149** para Pruebas de circuitos y sistemas. Otro tanto sucede prácticamente con todas las herramientas de síntesis e incluso en algunas de simulación, en las que solamente una parte de las sentencias está soportado y, lo que es un cierto problema, no siempre es el mismo conjunto ni los formatos o las inferencias para síntesis son los mismos.

En cuanto al aprendizaje del VHDL, como en cualquier lenguaje o idioma, no es preciso ser un experto “académico” para poder expresarse en VHDL y realizar modelos que sean aceptados por el compilador poco tiempo después de conocer los conceptos básicos. El primer pequeño problema es que, como tantas otras cosas, hay que aprenderlo secuencialmente mientras que para utilizarlo bien deberían aprenderse muchos conceptos concurrentemente, expresándose con términos que pronto verá el lector. Esto hace algo problemático el orden en presentar conceptos y ejemplos. Al parecer, la opinión de diversos autores no es coincidente, según se deduce de la ordenación de temas en distintos libros sobre VHDL. En cualquier caso, la práctica con un programa de simulación y, tal vez, releer entre compilación y compilación, ayudará a ver cómo se pueden realizar modelos más eficientes, concepto poco claro, que debiera entenderse como en cualquier proyecto por la calidad o robustez del modelo, seguido del coste y del tiempo para su desarrollo. Sin duda, todos esos parámetros se optimizan con un conocimiento general de todas las posibilidades del lenguaje.

Esta publicación titulada “*Desarrollo de modelos en VHDL*”, sin hacer una cobertura amplia y detallada con muchas páginas más, pretende ser algo más que una simple introducción al lenguaje “a complementar con los mensajes de error que dará el compilador” que, sin duda, también enseñan, pero no son el mejor método para aprender. En su redacción se han considerado los conceptos fundamentales del VHDL : los necesarios para que el contenido de las secciones 6 , 7 y 8 pueda seguirse fácilmente, permitiendo desarrollar los modelos que, a nivel académico, serán suficientes para la realización de Proyectos o diseños sobre sistemas digitales. También se ha considerado que el aprendizaje y la práctica del VHDL requieren ya bastante esfuerzo de abstracción como para tener que hacerlo en un idioma distinto del materno.

Solo resta desearle al lector que el tema y su tratamiento le resulten interesantes.

El Autor

Málaga, Septiembre de 1996.