

Antecedentes

El procesador es un componente electrónico que interpreta instrucciones y procesa los datos contenidos en los programas. Son conocidos también como CPU (Central Processing Unit) y es uno de los componentes necesarios dentro de una computadora, junto con la unidad de memoria y los dispositivos de entrada/salida.

Los primeros procesadores fueron diseñados a la medida, como parte de una computadora más grande, sin embargo este método de diseño para una aplicación en particular ha desaparecido y se ha sustituido por el desarrollo de procesadores baratos y estandarizados adaptados para uno o muchos propósitos. La popularización de los circuitos integrados permitió el diseño y fabricación de procesadores más complejos y en espacios menores.

Desde la aparición en la década de los '70s de los microprocesadores de Intel, este tipo de procesadores tomó el control del mercado, desplazando casi totalmente al resto de las alternativas de implementación de una unidad central de proceso en esa época. Actualmente compañías como AMD han ganado bastante mercado en este rubro.

Los microprocesadores deben cumplir con ciertas capacidades, la primera es leer y escribir información en la memoria. Esto es decisivo ya que las instrucciones del programa que ejecuta el microprocesador y los datos sobre los cuales trabaja están almacenados en esa memoria. La otra capacidad es reconocer y ejecutar una serie de comandos o instrucciones proporcionados por los programas. La tercera capacidad es decirle a otras partes del sistema lo que deben de hacer, para que el procesador pueda dirigir la operación a realizar. En pocas palabras los circuitos de control del microprocesador tienen la función de decodificar y ejecutar

el programa, que en sí es un conjunto de instrucciones para el procesamiento de los datos.

Por otro lado, un coprocesador es un procesador utilizado como suplemento de las funciones principales del procesador principal. Las operaciones ejecutadas por un coprocesador pueden ser aritméticas, procesamiento gráfico, procesamiento de señales, criptografía, entre muchas otras.

La principal razón para construir coprocesadores es para evitar que el procesador principal realice tareas intensivas, acelerando así el rendimiento del sistema. Debido a que los coprocesadores son más especializados, las tareas para las que son diseñados se realizan más eficientemente.

Los coprocesadores no son procesadores de propósito general, requieren de un procesador principal que lea las respuestas del coprocesador y maneje todas las operaciones junto con las funciones del procesador.

Actualmente los coprocesadores se pueden implementar principalmente de tres formas, dependiendo de su fin:

- Empleando un microprocesador adicional que cumpla con los requerimientos del coprocesador que se desea diseñar, que se comunique con el procesador principal y siga un programa preestablecido para poder llevar a cabo las tareas para las cuales es requerido.
- Por medio de un DSP (*Digital Signal Processor*), en el caso de tareas que requieran una velocidad bastante alta, o un nivel de procesamiento matemático bastante elevado.
- Usando lógica programable. En este caso, se pueden construir coprocesadores a la medida y hechos para casos específicos de acuerdo a las necesidades del sistema.

Lógica programable y HDLs

Al escoger lógica programable como base para implementar un coprocesador, uno tiene mayor libertad sobre el número de entradas y salidas, líneas de control y cualquier puerto que se vaya a ocupar para comunicarse con el procesador principal o con otros elementos externos que se requieran. Esta libertad, comparada con microprocesadores, es mayor ya que en los μ Ps se tiene preestablecido el número de puertos de entrada/salida y el número de bits en cada puerto. No hay que olvidar, sin embargo, que en los dispositivos lógicos programables también se tienen limitaciones en cuanto a número de bits por bus, dependiendo de su arquitectura interna, y de los pines disponibles ya que siempre se tienen algunos preestablecidos.

Ahora, existen diferentes formas de diseño para implementar un coprocesador mediante lógica programable; la más arcaica y de un bajo nivel de abstracción sería mediante diagramas lógicos, empleando la herramienta software pertinente. El usuario crea su esquemático, o diagrama RTL empleando desde compuertas básicas hasta bloques parametrizados y realizando las conexiones pertinentes se puede obtener el sistema requerido, sin embargo dicha captura no es estandarizada, por lo que no existe posibilidad de portabilidad. En otras palabras, no es 100% seguro el traslado del diseño de un dispositivo de un fabricante a otro.

Por esta razón, se crearon los HDLs (Hardware Description Languages), que permiten documentar interconexiones y el comportamiento de un circuito sin necesidad de utilizar diagramas esquemáticos.

El flujo de diseño empleando HDLs suele ser típico:

1. Definir las tareas que tiene que hacer el circuito
2. Escribir el programa usando un HDL.
3. Comprobación de sintaxis y simulación de la descripción.

4. Programación del dispositivo y comprobación del funcionamiento.

Los rasgos comunes entre todos los HDLs suelen ser la independencia del hardware y la modularidad o jerarquía. Esta última se refiere a que una vez hecho un diseño, éste se puede usar dentro de otro diseño más complicado y con otro dispositivo compatible. Por ejemplo, en el caso del objetivo de esta tesis, que se refiere al diseño de un multiplicador, éste puede ser usado para un diseño más complejo que involucre cálculos en los que se requiera una multiplicación.

Actualmente los principales HDLs son SystemC, Verilog y VHDL. El primero está basado en un conjunto de librerías implementadas en C++. Los otros dos son lenguajes de descripción de hardware de alto nivel y son populares para dispositivos complejos, sin embargo, pueden ser empleados también para diseños de baja complejidad.

Verilog es un lenguaje de descripción de hardware (HDL), desarrollado más o menos al mismo tiempo que VHDL, por Gateway Design Automation y se convirtió en un estándar IEEE en 1995. Las bases sintácticas de VHDL están basadas en ADA (lenguaje de programación orientada a objetos), mientras que las de Verilog están basadas en C. Debido a esto, algunos consideran a Verilog más fácil de aprender (o menos intimidante). VHDL y Verilog mantienen más o menos el 50% del mercado cada una. Ambos lenguajes pueden satisfacer los requerimientos de diseño digital. Si se conoce alguno de los dos lenguajes, no es difícil hacer la transición de uno a otro.

Por estas razones, se empleó en este proyecto el VHDL como lenguaje para describir el coprocesador dentro de un dispositivo de lógica programable. En los capítulos siguientes se hablará sobre el VHDL, así como de dispositivos lógicos programables, desde los más básicos hasta algunos más complejos. Posteriormente se hablará sobre microprocesadores, en especial de PICs, ya que en este proyecto se empleará uno de ellos como procesador principal del sistema.