Capítulo 1

Introducción

Gran parte del desarrollo tecnológico llevado a cabo en las últimas décadas se ha conseguido gracias al incremento de la capacidad de computación de los ordenadores modernos. Paralelamente este desarrollo ha sido favorecido por la mejora y el escalado continuo de las tecnologías microlectrónicas (la ley de Moore que establece que la velocidad de procesado de un ordenador se duplica cada año y medio). Además, se han diseñado procesos tecnológicos que permiten la fabricación a gran escala de los productos electrónicos, lo que ha provocado que el precio de éstos sea lo suficientemente bajo como para que su penetración en el mercado llegue a las cotas que se conocen hoy en día. No obstante, aún quedan retos pendientes y situaciones donde las ideas que han permitido todo este proyecto no son eficientes.

Tareas que hace unos años parecían ciencia ficción son hoy por hoy, hasta cierto punto, triviales y requieren un tiempo de procesado ínfimo. Esto es especialmente notable en aplicaciones eminentemente secuenciales, como por ejemplo los cálculos matemáticos complejos. No obstante, estos esquemas secuenciales presentan limitaciones cuando la fuente de información de entrada tiene un carácter esencialmente paralelo. Por ejemplo, hacer una integral en un ordenador no es más que implementar una serie de operaciones ariméticas mediante un algoritmo que guíe los cálculos, que se realizan uno tras otro. Pero, ¿qué pasa si se desea, por ejemplo, hacer el seguimiento de un objeto en tiempo real? El enfoque tradicional establece que habría que capturar imágenes de la escena cada cierto tiempo (limitado por la tecnología y la velocidad del objeto que se desea seguir) y, en cada paso, analizar la imagen para extraer las características esenciales de la misma. Ninguna de estas tareas es trivial y requieren millones de operaciones para obtener poca información.

La pregunta que cabe hacerse es: ¿se puede mejorar aún más la eficiencia de los sistemas de procesado? Parece una tarea difícil, pero en ciertas aplicaciones la respuesta es que sí. La manera de hacerlo es conceptualmente simple: si los seres humanos hacemos estas tareas de forma rápida y eficiente, ¿por qué no imitar el funcionamiento del cerebro? Esta idea da lugar, además de muchos otros, a los sistemas bio-inspirados basados en redes de neuronas, donde el paralelismo es la clave para la mejora de la eficiencia. En vez de tener una unidad de procesado secuencial muy potente, se tienen unidades pequeñas con escasa capacidad (neuronas) pero replicada múltiples

veces, de forma que la capacidad de procesado sea suficiente como para llevar a cabo tareas complejas.

Siguiendo esta idea se pueden implementar procesadores en tiempo real de alta eficiencia para aplicaciones tan dispares como el análisis de vídeo o el reconocimiento de voz: tareas que para un humano son triviales, pero que para las máquinas representan un verdadero reto. La idea de fondo parece sencilla, pero su implementación no lo es tanto. Uno de los grandes problemas cuando se quiere integrar uno de estos sistemas de procesado en un circuito integrado es el masivo paralelismo. Para obtener una eficiencia razonable es necesario que el número de neuronas sea muy alto (del orden de miles o incluso millones), mientras que el área de silicio disponible para su implementación circuital es limitada. Además, esas neuronas tienen una densidad de conectividad muy alta, de forma que, aunque físicamente se pudieran integrar todas en un chip, sería aún más difícil interconectarlas entre sí. De esta forma, parece que lo más razonable es distribuir las neuronas del sistema en varios chips que se comuniquen entre sí cada vez que los cambios en la información de entrada lo requieran.

Por tanto, el escenario planteado está compuesto por una serie de chips que contienen un conjunto de neuronas que generan eventos (pulsos de carga) cada vez que quieren transmitir información hacia otras neuronas que pueden estar en otros chips distintos. Debido a la enorme complejidad del sistema, necesitamos un protocolo que arbitre esta comunicación y que defina cómo debe ser la comunicación entre las neuronas, de forma que la complejidad total del sistema quede reducida. En este sentido, se suele aplicar el protocolo AER (*Address Event Representation*), que permite hacer estas labores de forma eficiente y simple. El uso de AER reduce en gran medida la complejidad, pero obliga a usar buses digitales (que pueden ser hasta de 16 bits) para comunicar los chips. En implementaciones reales, donde se tienen varios chips que se quieren integrar en una misma PCB, esto puede suponer una limitación importante, pues los recursos de rutado son limitados.

Pues bien, el objetivo de este proyecto es explorar una posible solución para este problema, sustituyendo la comunicación en paralelo tradicional por un esquema serie más eficiente a altas velocidades. Existen varios estándares industriales que permiten abordar este problema (ECL, PECL, PCI, ...) y diseñar una interfaz serie eficiente. Para la implementación que aquí se presenta, se ha decidido optar por LVDS (*Low Voltage Differential Signalling*), pues permite alcanzar altas velocidades con bajo consumo y resultar robusto frente a ruido e interferencias, además de propocionar una transmisión serie. LVDS es un estándar industrial de amplio uso y que está encontrando múltiples aplicaciones en los últimos años debido al incremento de las tasas de transmisión entre chips, que han obligado a adoptar nuevos esquemas de transmisión más eficientes, pero al mismo tiempo más complejos.

El sistema LVDS típico está formado por un transmisor que genera los datos en formato paralelo, un serializador que los traduce a un formato serie, un driver que los transmite por el canal, un receptor que interpreta los datos y un deserializador que los convierte de nuevo a formato paralelo. En este proyecto se analizarán todos estos bloques y se realizará un diseño completo de la interfaz a nivel de circuito y una verificación del mismo mediante resultados

de simulación. También se ha tratado de incluir consideraciones adicionales, más allá de las estrictamente necesarias para el diseño de la interfaz, para facilitar tareas posteriores que se deberán abordar: diseño de PCB's para aplicaciones de alta velocidad o un estudio de circuitos LVDS a nivel de transistor con énfasis en los retos de diseño que impone el escalado de las tecnologías microelectrónicas.

La implementación de la circuitería necesaria para realizar la comunicación se ha realizado sobre una tecnología de 90 nm proporcionada por *STMicroelectronics*. Se trata de una tecnología, hasta cierto punto novedosa y puntera, que permite conseguir grandes velocidades de funcionamiento sin grandes costes de diseño. Al escalar la tecnología, es posible integrar más transistores en la misma área de silicio, lo que nos va a permitir integrar los circuitos de comunicaciones de una forma compacta, de manera que no se afecte al área disponible para el resto de los bloques que puedan existir en el chip. Por todo esto, el proceso ha supuesto un reto adicional de diseño pues no se tiene experiencia sobre ella y no se conocen sus particularidades. Además, las tecnologías submicrométricas imponen una serie de limitaciones al diseño, sobre todo analógico, que no existían con tecnologías anteriores.

El trabajo se ha estructurado en 8 capítulos contando con este de introducción. En el capítulo 2 se hace una revisión de los sistemas de comunicación basados en eventos que utilizan el protocolo AER como base para la comunicación. Se comentan los aspectos más relevantes, sus ventajas e inconvenientes y se presta especial atención a las consideraciones de diseño de los generadores y decodificadores AER. Esto es necesario para abordar el diseño de la interfaz, pues resulta imprescindible saber cómo funcionan estos circuitos para poder diseñar a continuación una interfaz con ellos que sea lo suficientemente robusta. Por otro lado, se trata también de colocar al lector frente al entorno en el que van a funcionar los circuitos diseñados y frente a las limitaciones que este entorno impone en el diseño.

A continuación, en el capítulo 3, se realiza un completo estudio de las interfaces LVDS a partir del análisis del estándar industrial que las define. Se presentan tanto características eléctricas y funcionales, como consideraciones de diseño y métodos de análisis y prueba de las mismas. Para completar el conocimiento acerca de los drivers y receptores LVDS, se ha incorporado, en el capítulo 4, un análisis de la circuitería implicada en la capa física de una interfaz serie. Se presta especial atención a los circuitos concretos que se van a usar para el diseño y que pertencen a una librería estándar de la tecnología usada para la implementación.

En el capítulo 5 se presenta una descripción detallada de la estructura y diseño a nivel esquemático de los circuitos utilizados para la serialización y la deserialización de los datos AER, que sirven como entrada para nuestro sistema. Para su diseño, se ha tratado de dividir el sistema en bloques funcionales, lo suficientemente pequeños y sencillos, como para que la comprensión del mismo sea fácil y la metodología de diseño utilizada sea lo menos proclive a errores posible.

El capítulo 6 se centra en la implementación física sobre el silicio de los circuitos presentados en el capítulo 5. Se describe tanto el estilo de layout utilizado como los resultados tras el proceso de diseño de cada una de las partes que componen la interfaz.

Una vez realizado el diseño, es necesario validarlo antes de su fabricación. Para ello se llevaron a cabo una serie de simulaciones, cuyos resultados se muestran en el capítulo 7. Así mismo, se describen también todas las técnicas utilizadas para mejorar la precisión de las simulaciones mediante la inclusión de efectos no ideales que pueden afectar al funcionamiento de la interfaz.

Por último, en el capítulo 8 se recopilan y presentan las conclusiones más significativas, así como las tareas y líneas de investigación que quedan abiertas tras este proyecto.

Capítulo 2

Comunicación por impulsos entre neuronas: el protocolo AER

2.1. Introducción

El desarrollo de los sistemas informáticos para el procesado de la información ha sido espectacular en los últimos años debido, en gran parte, a la mejora en los componentes hardware de los mismos. Sin embargo, aún existen ciertas tareas para las que los ordenadores no son lo suficientemente eficientes. A medida que los sistemas de procesado mejoran, las tareas que se les encomiendan se complican y se hacen más sofisticadas. Esto ha puesto de manifiesto que hay tareas en las que los sistemas artificiales son mejores que en otras. Si se trata de hacer complicados y engorrosos cálculos matemáticos, la velocidad de procesado es dificilmente superable. En cambio, a la hora de procesar imágenes, reconocer patrones, tomar decisiones, reconocer la voz humana,... los PC's no son tan eficientes como lo es el cerebro humano.

La razón de la ineficiencia de los sistemas informáticos tradicionales para realizar estas tareas, que para un humano serían muy sencillas, radica en la propia estructura de los mismos. Un PC está pensado para almacenar un programa y ejecutar un algoritmo que consiste en una serie de órdenes ejecutadas de forma secuencial. Los cálculos matemáticos se hacen en base a algoritmos, lo que hace que los ordenadores sean tan eficientes al procesarlos. Esto no es así para tareas en las que hay implícito un alto grado de paralelismo, como por ejemplo el reconocimiento de un objeto, donde hay que analizar simultáneamente muchos puntos de la imagen.

Así pues, parece necesario optimizar los sistemas artificiales para que se adapten mejor a la realización de ciertas tareas masivamente paralelas. Este hecho se ha afrontado desde varias perspectivas, pero la que hasta ahora ha dado mejor resultado ha sido la de las redes neuronales. La idea básica es construir un hardware que de alguna forma emule el comportamiento del cerebro humano para conseguir así las ventajas que éste tiene para, por ejemplo, el procesado de la información visual. Analizando la estructura interna del cerebro, llaman la atención dos factores esenciales: el elevado número de unidades de procesado de las que dispone (disponemos del orden de 10^{10} neuronas [2]) y la gran cantidad de conexiones que hay entre ellas (las llamadas