

Capítulo 3:

Convertidor Pipelined

- 3.1 Esquema general y artículos de partida**
- 3.2 Encoder: descripción de funcionamiento global**
- 3.3 Encoder: descripción de componentes**
 - 3.3.1 Árbol de Wallace de 5 bits**
 - 3.3.2 Registros de almacenamiento**
 - 3.3.3 Sumadores con acarreo anticipado**

Convertidor Pipelined

En este capítulo se describe el montaje elegido para la realización de un convertidor analógico-digital basado en una estructura pipelined y su implementación en tecnologías submicrométricas.

En primer lugar comentaremos el esquema general del convertidor, cuya idea básica ha sido sacada de una serie de artículos de partida, así como las modificaciones que han debido de realizarse y su funcionamiento general. Posteriormente, detallaremos el diseño de algunas de sus partes que han sido implementadas tanto en la tecnología de 130 nm como en la de 90 nm.

3.1 Esquema general y artículos de partida

El diseño del convertidor pipelined escogido se basa principalmente en un artículo de J. Kanan, F. Kaess y M. Declercq titulado “*A 640 mW high accuracy 8-bit 1 GHz flash ADC encoder*”. En él, se describe un decodificador (encoder) que permite pasar de código termométrico a binario y que puede ser utilizado dentro de una arquitectura pipelined para así conseguir una velocidad de funcionamiento de hasta 1 GHz.

El encoder descrito en dicho artículo puede considerarse la base principal del convertidor analógico-digital (ADC) a realizar. La idea básica del esquema general es utilizar una redundancia de comparadores a la entrada del encoder, y luego usar un sistema de calibrado que nos permita seleccionar aquellos que mayor precisión muestren. Esto es una solución a los problemas de matching que suelen presentar los convertidores analógico-digitales al tomar los rangos a la entrada de los comparadores, que reciben la señal analógica proveniente de una escalera resistiva. Habitualmente, éstas presentan una gran dificultad para ser apareadas de forma precisa debido a sus niveles de tolerancia. Este efecto genera imprecisión, lo que añadido a posibles efectos de *offset* de los comparadores redundan en que la precisión del convertidor no sea todo lo buena que se desearía.

Uno de las características básicas que definen el comportamiento y características de un convertidor analógico-digital es la resolución a la salida. Esto se corresponde con el número de bits que utilizamos para representar los diferentes de señal de forma digital a la salida del mismo. Para obtener la resolución deseada en un ADC, en general deberemos tener en la entrada un número de niveles que vendrá fijado por los bits que queremos alcanzar a la salida. Así, para una resolución de N bits a la salida se requieren $2^N - 1$ niveles de entrada. En el caso del diseño a implementar, se decidió que fuera únicamente de 6 bits a la salida en lugar de los 8 que se especifican en el artículo. En ese caso, para la resolución de 6 bits a la salida, necesitaremos disponer de $2^6 - 1 = 63$ niveles. Por tanto, en teoría son necesarios para realizar la conversión de forma adecuada un total de 63 comparadores.

No obstante, el esquema descrito en este artículo presenta una alternativa a las imprecisiones comentadas anteriormente, y que se derivan del propio proceso de fabricación de los componentes. Así, se consigue una mayor precisión mediante la utilización de esquemas basados en redundancia y selección de los componentes a usar en el proceso de conversión. Al usar esta estructura basada en la redundancia de comparadores, el diseño se simplifica porque los comparadores no requieren tener unas características tan exigentes de *offset*. La elección del número de bloques redundantes se realiza en base a algoritmos matemáticos que demuestran que con una redundancia de 4 ya puede conseguirse precisión suficiente. Por tanto, en el convertidor a realizar, la entrada del encoder estará compuesta por 4 bloques similares de los que posteriormente se utilizarán sólo los que sean necesarios.

El ADC a diseñar presenta una estructura basada en la redundancia de etapas de entrada (comparadores). Así, establece cuatro bloques distintos y redundantes de comparadores con un rango común de entrada, y luego seleccionar de entre cada uno de ellos los que resulten más precisos. Aunque el rango de entrada sea común, cada bloque de comparadores estará acompañado de su propia escalera resistiva para establecer los niveles de comparación. La selección de los comparadores se realiza al comienzo del

funcionamiento del convertidor mediante un esquema de calibrado al margen del esquema del propio ADC. Por ello, tendrá lugar mediante un proceso totalmente independiente del posterior funcionamiento del encoder para realizar la conversión analógico-digital.

Los comparadores elegidos durante el calibrado serán los que mayor precisión presenten al cambiar frente a los niveles de la señal a la entrada lo más cerca posible de los umbrales ideales. Así, vamos a encontrarnos con que no tendrá por qué mantenerse la ordenación teórica de niveles sucesivos, e incluso podrían intercambiarse unos con otros de mayor precisión para los distintos bloques. Por tanto, todas las conversiones que realice el ADC serán hechas por los comparadores elegidos para tener la mayor precisión posible. Es por ello por lo que este esquema de calibración sólo se necesita que entre en funcionamiento una vez, ya que tan sólo se utiliza para elegir los comparadores necesarios que requiere la resolución del ADC, que en el caso de este diseño será de 6 bits. El análisis detallado del funcionamiento del encoder se recoge en el apartado siguiente.

3.2 Encoder: descripción de funcionamiento global

El encoder es el bloque fundamental del ADC, puesto que se encarga de realizar la conversión de uno a otro código de señales binarias. La conversión de tensión analógica a digital la realizan los bloques formados por la escalera resistiva más los comparadores, a la salida de los cuales se obtiene ya una señal digital. No obstante, esta serie de señales digitales no pueden ser expresadas directamente y deben ser convertidas a otro código interpretable a la salida. En nuestro caso, debido al esquema de funcionamiento elegido, lo que necesitamos es pasar de un código termométrico, basado en el número de unos digitales existentes, a un código binario tradicional.

El diseño a implementar se basa en un esquema de redundancia y selección, por lo que se requerirá un número mayor de componentes de los estrictamente necesarios. Como se ha comentado anteriormente, para una

resolución de 6 bits se requieren 63 niveles a la entrada. Éste deberá ser por tanto el número final escogido que dé como resultado del esquema de calibración para realizar las funciones de conversión.

Inicialmente, la redundancia en los comparadores estaba prevista realizarse mediante cuatro bloques de 63 comparadores, es decir, que así el encoder debería tener 252 entradas digitales. Estas irán cableadas al encoder, cuya labor es la de realizar la conversión a código binario de una serie de unos “desordenados” a la entrada. Se trata por tanto de una conversión de código termométrico a binario, con ciertas particularidades. Debido a las características del esquema utilizado, lo que tendremos a la entrada del encoder será una serie de unos lógicos localizados aleatoriamente que se correspondan con un valor concreto de tensión analógica.

El esquema propuesto en el artículo para la realización del encoder en un convertidor analógico-digital de 8 bits es el siguiente:

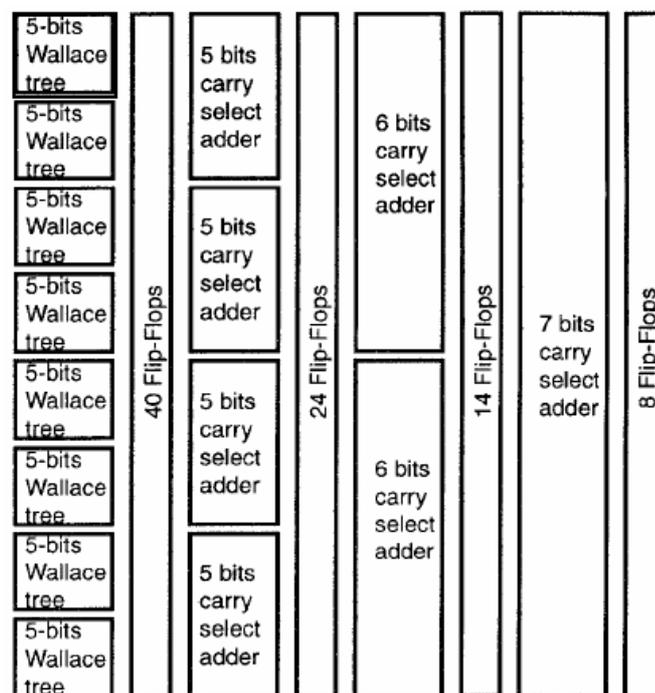


Figura 3.1- Esquema general del encoder para ADC de 8 bits

En dicho esquema podemos apreciar los tres componentes básicos del encoder, que serán comentados en mayor detalle en el apartado 3.2 y que son los siguientes:

- Árboles de Wallace de 5 bits (Wallace tree)
- Registros de almacenamiento (flip-flops)
- Sumadores con selección de acarreo

Aunque serán explicados más adelante de forma exhaustiva, describimos ahora su funcionalidad para así entender mejor el funcionamiento del esquema propuesto.

Las primeras etapas (árboles de Wallace) realizan la función de contar el número de unos lógicos existentes a la entrada del encoder. Esto es necesario puesto que el esquema selecciona los comparadores más precisos, y ello implica que no tienen por qué estar ordenados según ningún patrón concreto. Esta estructura está documentada como la más rápida para efectuar operaciones de cuenta de unos lógicos, de ahí su elección para utilizarse en este esquema. Su estructura y diseño se verán más adelante.

Las etapas intermedias (flip-flops) son registros de almacenamiento que únicamente sirven para mantener las salidas de una etapa y entregarlas a la etapa posterior en el siguiente ciclo de reloj. Con ello se consigue realizar operaciones en paralelo, es decir, que dotan al circuito de la denominada estructura *pipelined*. Con ello, la frecuencia de funcionamiento efectiva que se consigue es mayor a partir de un retraso base de un cierto número de ciclos de reloj igual al número de etapas de procesado de señal que tiene el esquema.

Los sumadores lo que hacen es realizar las sumas parciales de los resultados que les llegan de etapas anteriores, para obtener finalmente como resultado el número de unos a la entrada.

Es decir, que el proceso completo a realizar sería el siguiente: Los árboles de Wallace cuentan el número de unos que existen a su entrada y

pasan dicho resultado a la etapa de sumadores sucesiva a través de los registros. Los sumadores van realizando sumas parciales de unos y otros componentes a su entrada, pasando el resultado a la siguiente etapa también a través de etapas de flip-flops. Así, en un intervalo de cuatro ciclos de reloj, se consigue llegar a un punto en que se unifica el resultado, que será por tanto el número de unos a la entrada, que será expresado ya en un código binario fácilmente interpretable como salida del convertidor analógico-digital.

No obstante, este esquema mostrado puede simplificarse en el caso de nuestro diseño, puesto que para él sólo se requieren 6 bits a la salida. Teniendo en cuenta esa condición, podemos simplificar las etapas posteriores reduciendo tanto el número de registros intermedios como la complejidad de los sumadores en las etapas finales. Con ello, el esquema que en realidad quedaría implementado para este caso sería el siguiente:

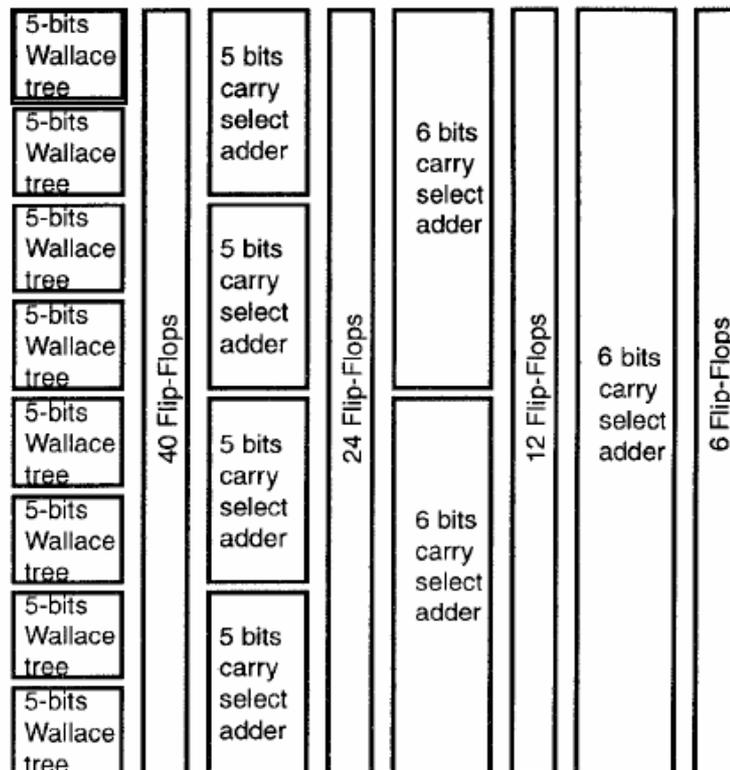


Figura 3.2.- Esquema del encoder para convertidor de 6 bits

Como veremos posteriormente, cada árbol de Wallace de 5 bits dispone de 31 entradas, lo que hace que esta estructura sólo pueda disponer de 248 entradas, cuando en teoría para la redundancia nos harían falta 252 (4×63). El problema que presenta esto es que en teoría debería doblarse todo el esquema sólo para obtener esas 4 entradas, por lo que en principio se opta por esta configuración perdiendo esos cuatro niveles.

Sin embargo, existe aún la posibilidad de que ocurra otro efecto indeseado que este diseño no tiene en cuenta. Si por ejemplo todos los comparadores de un bloque tuvieran el *offset* desplazado hacia un mismo sentido del rango (bien arriba o bien abajo), estaríamos perdiendo una serie de niveles de conversión en la escalera resistiva y por tanto sería imposible realizar la conversión de un modo aceptable. Por ello, se decide implementar las escaleras resistivas con una serie de resistencias adicionales por encima y por debajo del rango ideal. Idealmente, se opta por la solución de colocar 10 resistencias adicionales por encima y debajo del rango necesario.

No obstante, después de numerosos cálculos y aproximaciones para lograr un diseño más o menos óptimo teniendo en cuenta el número de entradas disponibles y el de niveles requeridos, se llega a la conclusión de que este número es inviable en cada escalera resistiva. Para no incrementar dramáticamente la electrónica utilizada, se necesita que el número de entradas total sean un múltiplo del número entero de entradas a un árbol de Wallace, o al menos muy próximo a ello para realizar la configuración de un modo efectivo. Así, finalmente se opta por un diseño de 315 entradas, algo menos del pensado inicialmente, basado en usar diez árboles de Wallace y cablear una entrada más a la etapa posterior por cada dos de ellos.

Con esta configuración, quedaría una redundancia de 4 bloques con su rango ampliado, tres de ellos de 79 y uno de 78 niveles (por cada nivel necesitaremos un comparador, y el número de resistencias de la escalera es uno más que el número de comparadores utilizados). Con ello, el número es algo menor de lo que idealmente deseable, pero se cree suficiente puesto que la hipótesis considerada es un caso extremo más improbable además en caso

de la utilización de esquemas con redundancia. Es decir, las etapas de entrada que realizan la conversión de tensiones analógicas y van después al encoder presentan el esquema siguiente:

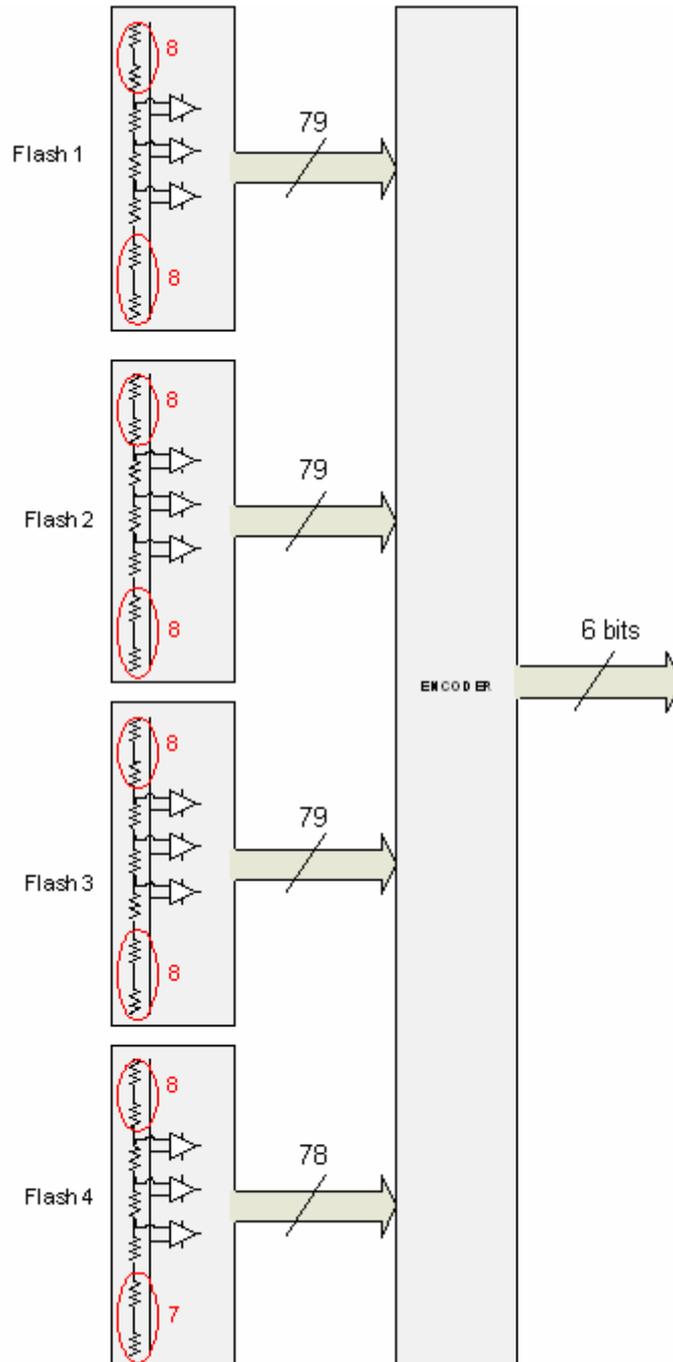


Figura 3.3- Esquema de bloques redundantes a la entrada para el diseño final del ADC de 6 bits

Por tanto, con esta configuración a la entrada, el esquema definitivo que se implementa para el encoder vuelve a verse modificado y es ahora el que se muestra a continuación:

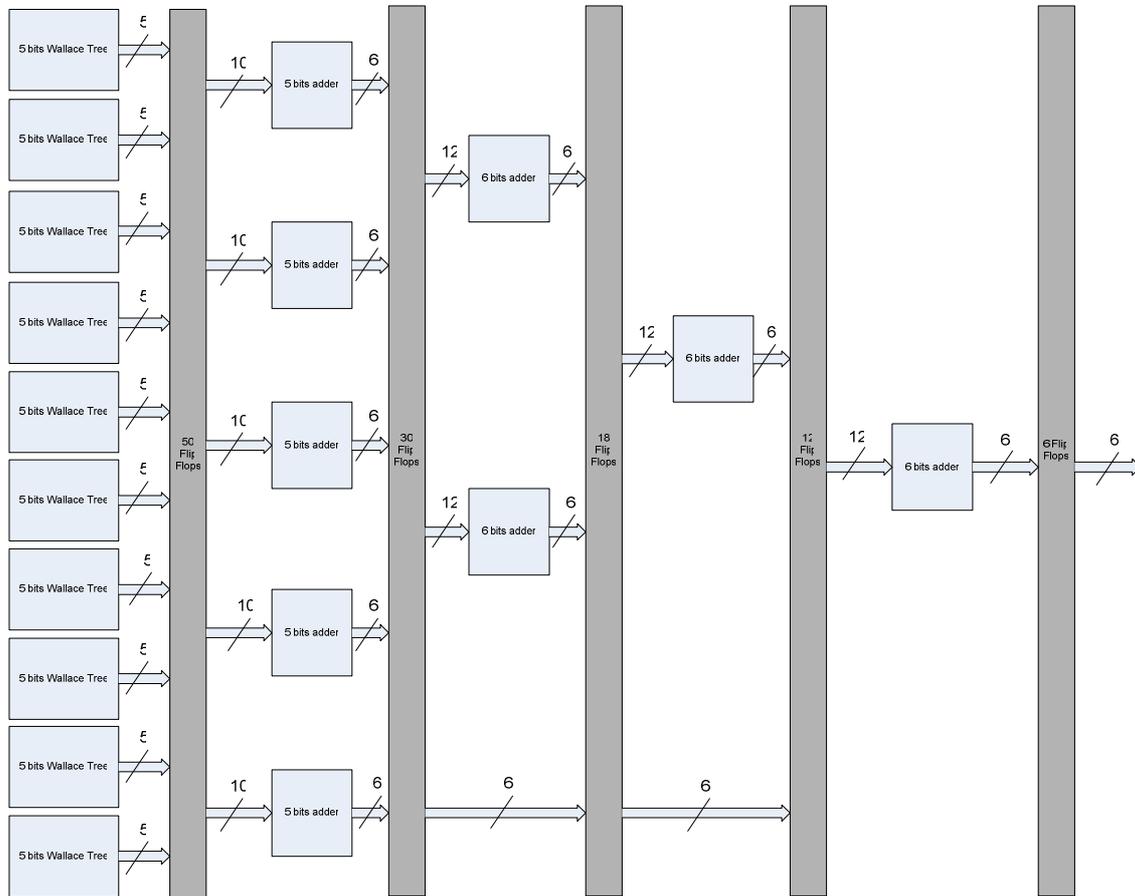


Figura 3.4- Diseño final del encoder pipelined para ADC

Como puede apreciarse, se produce un incremento en la electrónica utilizada, hasta el punto de que resulta necesario aumentar el número de registros y añadir una etapa más al encoder. Esto provoca que el resultado final del ADC ahora tenga un retraso de 5 ciclos de reloj en lugar de los 4 que tenía inicialmente.

Una vez descrito el funcionamiento global del esquema, vamos a pasar a describir en detalle la funcionalidad y características de los componentes básicos del encoder en el siguiente apartado.

3.3 Encoder: descripción de componentes

Como se ha comentado anteriormente, el encoder diseñado consta de tres componentes básicos: árboles de Wallace de 5 bits, registros de almacenamiento flip-flops y sumadores con acarreo.

En los sucesivos apartados se describen las características básicas de cada uno de los bloques, así como su funcionalidad en el esquema y las razones por las que han sido elegidos frente a otras alternativas.

3.3.1 Árbol de Wallace de 5 bits

Constituye el bloque inicial de entrada al sistema. Su elección se justifica en base a que resultan los esquemas más rápidos para realizar la cuenta de unos lógicos. A pesar de que existen otras configuraciones que realizan la conversión de código binario a termométrico con mayor velocidad, éstas siempre requieren que los unos lógicos a la entrada estén ordenados. Debido al principio de funcionamiento del convertidor, esto en nuestro caso no tendrá por qué ser así, por lo que las entradas en general están distribuidas de forma aleatoria. Por tanto, los árboles de Wallace constituyen la opción más eficiente para realizar la conversión de uno a otro código dentro del encoder.

Siguiendo las directrices del artículo mencionado anteriormente, los árboles de Wallace implementados constan de salidas de 5 bits. Esta es la configuración más eficiente, puesto que se forman a partir de árboles de 4 bits sumando las salidas de ambos. Este diseño no existía documentado como tal y fue realizado por Rafael Ibáñez Hueso en el Proyecto Fin de Carrera "*Diseño de un convertidor analógico-digital con esquema de autocalibración*", donde queda recogido en mayor detalle.

El esquema de un árbol de Wallace de 4 bits está compuesto de una serie de celdas elementales que se combinan según la configuración mostrada en la siguiente figura:

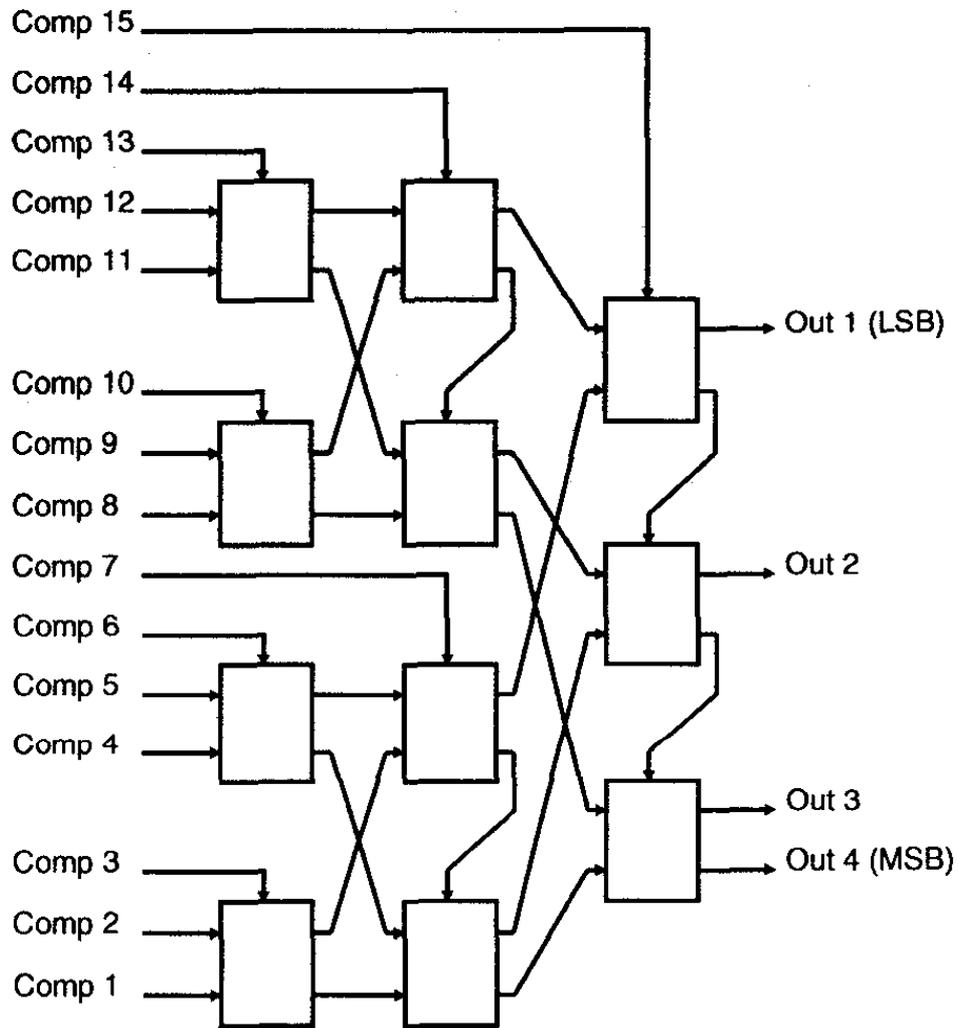


Figura 3.5.- Esquema de un árbol de Wallace de 4 bits

Cada celda unidad responde a una función lógica dada por una relación entre sus entradas y salidas que se caracteriza por expresar a la salida en código binario el número de unos lógicos que existen a la entrada. Por tanto, aplicando diseño mediante mapas de Karnaugh, se obtiene el siguiente esquema para cada celda elemental:

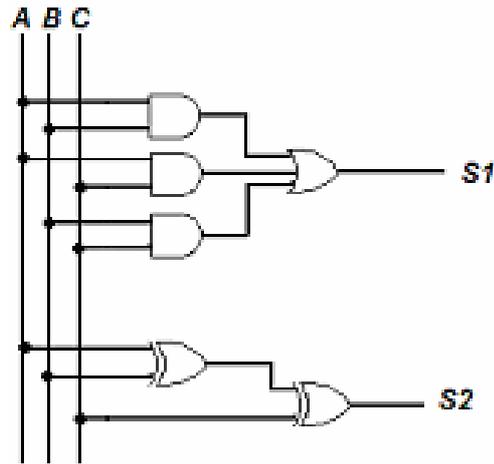


Figura 3.6. - Esquema de una celda unidad de árbol de Wallace

A partir de los árboles de Wallace de 4 bits, es relativamente sencillo extender el montaje a un árbol de Wallace de 5 bits. Para ello, es necesario utilizar dos árboles de 4 bits y cablear las salidas de ambos a nuevas celdas unidad, siguiendo el esquema de cableado siguiente:

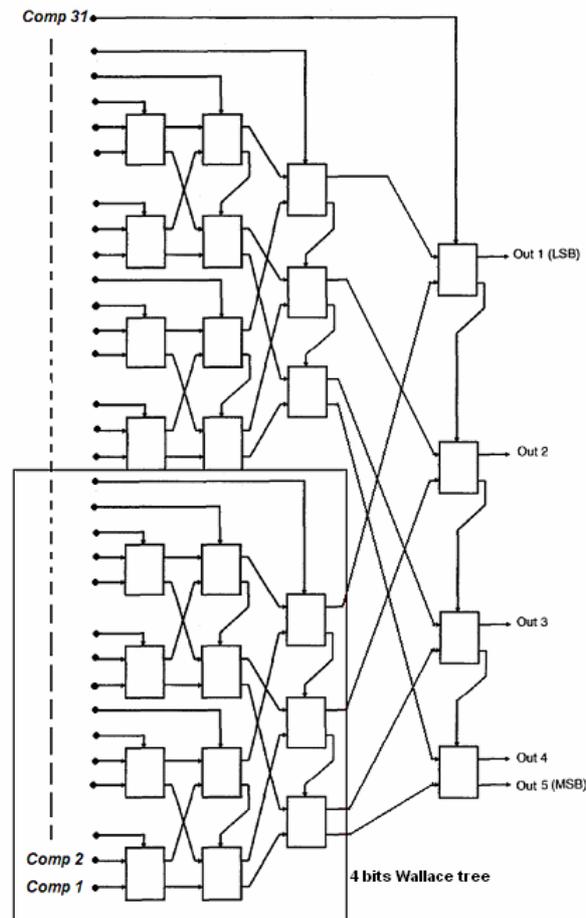


Figura 3.7. - Esquema de árbol de Wallace de 5 bits

3.3.2 Registros de almacenamiento

Las etapas intermedias formadas por registros constituyen únicamente una interconexión para el correcto funcionamiento del esquema pipelined. Simplemente, se trata de un número concreto de registros colocados en paralelo (tantos como salidas tenga la anterior etapa) que se encargan de mantener ese valor y pasarlo a la entrada del siguiente bloque combinacional en el próximo ciclo de reloj. Otorgan el comportamiento síncrono al esquema.

En el diseño final, hay cinco etapas intermedias de registros que interconectan los bloques funcionales del encoder entre sí. Cada una de ellas está compuesta de 50, 30, 18, 12 y 6 flip-flops respectivamente.

3.3.3 Sumadores con selección de acarreo

Estas etapas se encargan de realizar operaciones de suma de las salidas de la etapa anterior proporcionadas por registros de almacenamiento. Se dividen en varios bloques funcionales sucesivos para aprovechar al máximo la estructura pipelined.

En cada etapa se implementan sumadores de 6 bits, como se describe en el esquema de la figura 3.4 (apartado 3.2). En la primera etapa, las entradas provienen de los árboles de Wallace y poseen 5 bits, por lo que es posible realizar la suma correctamente con 6 bits para el resultado. En las etapas sucesivas, a pesar de que haya entradas de 6 bits y salidas del mismo número, no puede producirse desbordamiento. Esto se justifica debido al propio principio de funcionamiento del ADC, puesto que el máximo número de comparadores activos a la entrada será 63, que es perfectamente representable mediante 6 bits en código binario.

Los sumadores empleados se basan en un esquema de sumadores en paralelo con acarreo anticipado. Su elección se basa en que son los sumadores con mayor velocidad de cálculo, aunque a cambio pueda producirse

un mayor coste de utilización de área. El principio de funcionamiento de este esquema se basa en sumadores trabajando en paralelo que realizan las operaciones de suma de cada bit tanto para acarreo positivo como negativo, y una vez calculado el mismo en la etapa posterior, este sirve para seleccionar el resultado válido en la siguiente. Así, el esquema genérico en el que se basa el diseño de cada uno de los sumadores de 6 bits utilizados sería el siguiente:

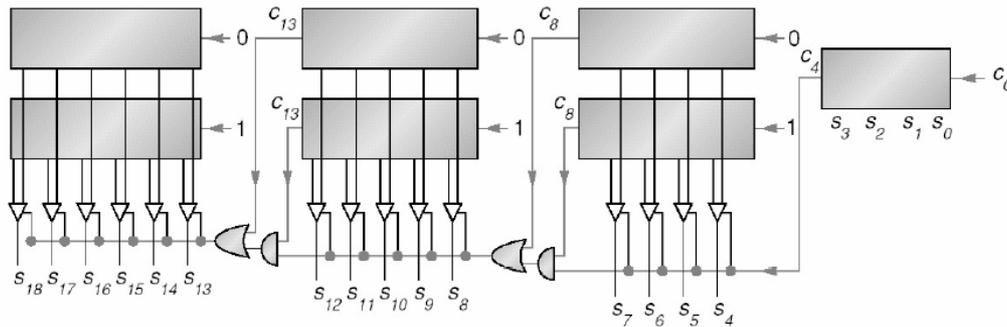


Figura 3.8. - Esquema genérico de sumadores con selección de acarreo

El diseño y las simulaciones del comportamiento de todos estos diseños se detallan en el capítulo 4 que se presenta a continuación.