

## *Capítulo 6.*

# *ESQUEMA DE AUTOCALIBRACIÓN DIGITAL*

- 6.1 ARQUITECTURA EXTERNA*
- 6.2 ARQUITECTURA INTERNA*
- 6.3 FUNCIONAMIENTO*

## 6.1 ARQUITECTURA EXTERNA

En este capítulo nos centraremos en desarrollar el esquema de calibración, que será como ya hemos mencionado anteriormente la pieza clave del convertidor A/D que estamos desarrollando a lo largo de este proyecto.

De los cuatro bloques de comparadores que tenemos en nuestro convertidor (cuatro debido a la redundancia), deberán escogerse los 63 cuyo offset es menor, esta tarea será realizada por el esquema de calibración digital. Esos 63 comparadores serán por tanto, aquellos cuya transición se realice más cerca del umbral ideal. Según el artículo de Conor Donovan y Michael P. Flynn referenciado en el capítulo 3 de este proyecto<sup>1</sup>, se puede escoger cualquier comparador por encima o por debajo del umbral. La idea que vamos a desarrollar por tanto, consistirá en coger el comparador que cambie más cerca del nivel de referencia ideal.

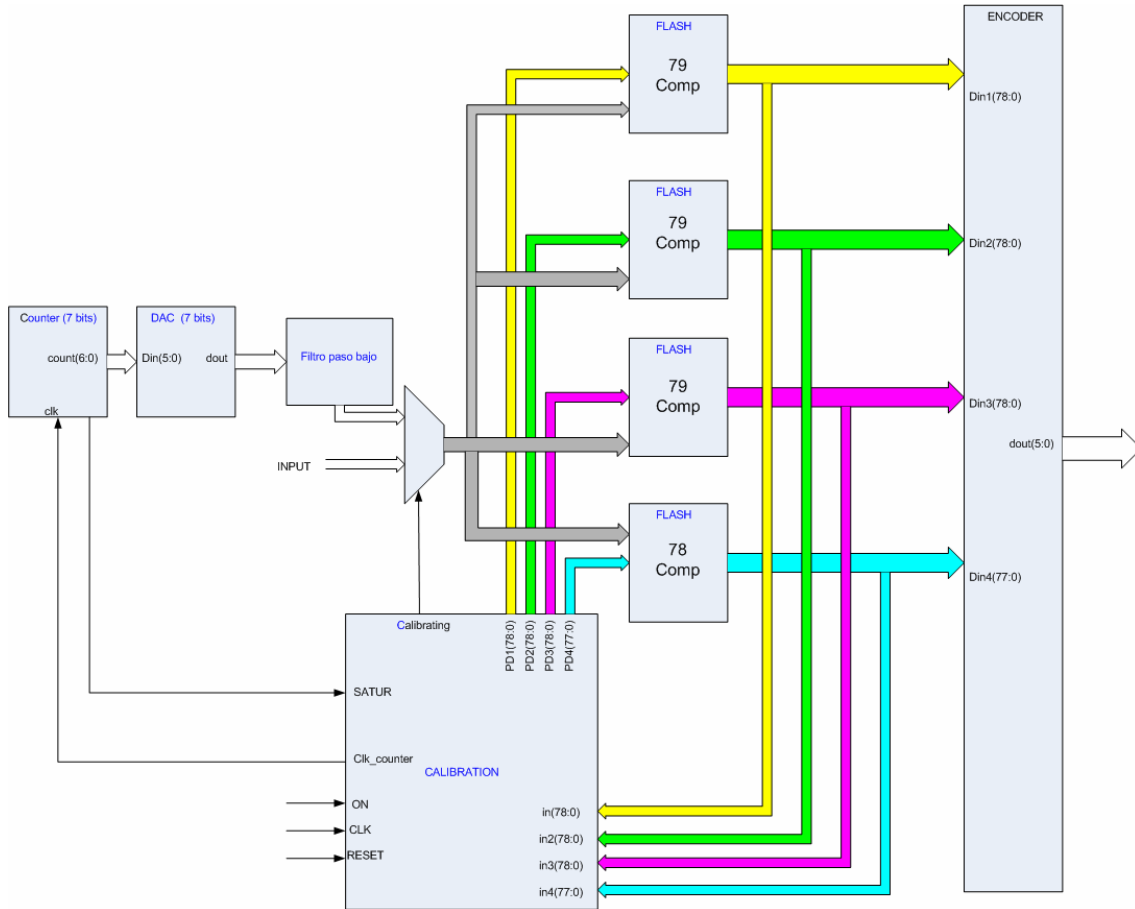
En primer lugar definiremos las señales externas que podrá tener nuestro bloque de calibración para poder controlar al resto de los elementos del circuito. El esquema para nuestro bloque de calibración será el mostrado en la figura 6.1. Como podemos ver en la figura, el esquema principal de calibración constará primeramente de varias señales externas que interaccionarán con el resto de bloques del circuito para tomar resultados o actuar sobre ellos. El significado de cada una de estas señales externas se detalla a continuación:

1. ON: Será la encargada de activar el esquema de calibración y por tanto comenzar el ciclo de calibración. Se activará cuando se encienda por primera vez nuestro convertidor A/D.
2. CLK: Señal de reloj de nuestro esquema de calibración. Como ya hemos mencionado anteriormente en otros capítulos, la frecuencia de operación de nuestro esquema de calibración puede ser mucho más baja que la frecuencia de operación de nuestro convertidor ya que la fase de calibración tiene lugar sólo una vez, al encender nuestro convertidor, y puede tardar unos cuantos ciclos en finalizar, sin importar la velocidad.
3. RESET: Será la encargada de llevar el bloque de calibración a un estado conocido al principio de la fase de calibración.
4. Clk counter: Señal de reloj generada por el bloque de calibración digital y que será la usada por el contador del generador de rampa. La cuanta del contador del generador de rampa avanzará con los flancos de subida de esta señal.

Calibrating: Señal que se activará cuando el ciclo de calibración haya finalizado cambiando la selección del multiplexor. En la fase de calibración, esta señal esta a “0” con lo que la señal de entrada que ven los comparadores es la rampa analógica de calibración. Una vez finalizado el ciclo de calibración, esta señal se activará a “1”, dando paso a la señal de entrada analógica a convertir.

---

<sup>1</sup> “A Digital 6-bit ADC in 0.25- $\mu$ m CMOS”, IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 37, NO. 3, MARCH 2002



**Figura 6.1. Detalles del convertidor A/D con esquema de calibración**

5. **PD**: Señales de activación o desactivación del Power-down de los comparadores. Servirán para desactivar los comparadores que no se han elegido durante el proceso de calibración y mantener encendidos a los elegidos. Con esta señal a "1" el comparador permanece encendido, mientras que si ponemos la señal a "0" el comparador se apaga mostrando un "0" a su salida. Habrá tantas señales de PD como comparadores.
6. **IN**: Señales de muestreo de las salidas de los comparadores. Estas señales no son más que las salidas de cada uno de los comparadores. Su utilidad consiste en poder conocer en todo momento la salida de cada comparador, para saber el instante exacto en el que se producen las transiciones de nivel bajo a nivel alto de nuestros comparadores, para según la cercanía de la transición con el nivel de referencia, escoger unos u otros comparadores.
7. **SATUR**: Señal que indicará el fin de la cuenta del contador del generador de rampa.

## 6.2 ARQUITECTURA INTERNA

Una vez que tenemos claras las señales externas que interaccionarán con el resto de bloques del circuito, pasaremos a describir detalladamente la arquitectura interna de nuestro esquema para conseguir un correcto control de esas señales de interacción. En la figura 6.2, podemos ver detalladamente la arquitectura interna de nuestro esquema de calibración. Los bloques más importantes de los que se compone dicho esquema son los siguientes:

1. Banco de Latches especiales: Es un conjunto formado por 315 latches (tantos como comparadores). Estos latches son unos latches especiales de diseño propio y que tendrán un comportamiento particular. Cada uno de esos latches es como el mostrado en la figura 6.3.

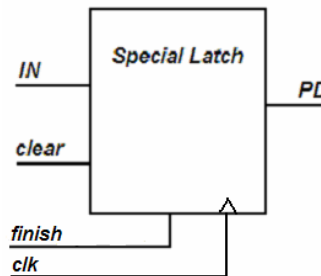


Figura 6.3. Latch especial

Como podemos ver, tenemos una señal *IN* que será la entrada al latch. Ahí irán conectadas las salidas de los comparadores del flash, cada comparador irá conectado a un latch a su salida. Tenemos también una señal de salida *PD*, que irá conectada a los *Power-Down* de los comparadores, concretamente cada latch tendrá su señal de salida *PD* conectada al *Power-Down* del comparador que tiene conectado a su entrada. Tendremos también dos señales de habilitación (*finish* y *clear*) y el reloj. El funcionamiento del latch se describe a continuación:

Action	Input	Output
Initially		PD = 0
Change of the input	IN = 1	PD = 1
Change of the input	IN = 0	PD = 1
Clear	Clear = 1	PD = 1
Change of the input	IN = 1 and Clear = 1	PD = 1
End of the calibration cycle	Finish = 1 and Clear = 1	PD = 0
End of the calibration cycle	Finish = 1 and Clear = 0	PD = 1

Figura 6.4. Funcionamiento del latch especial

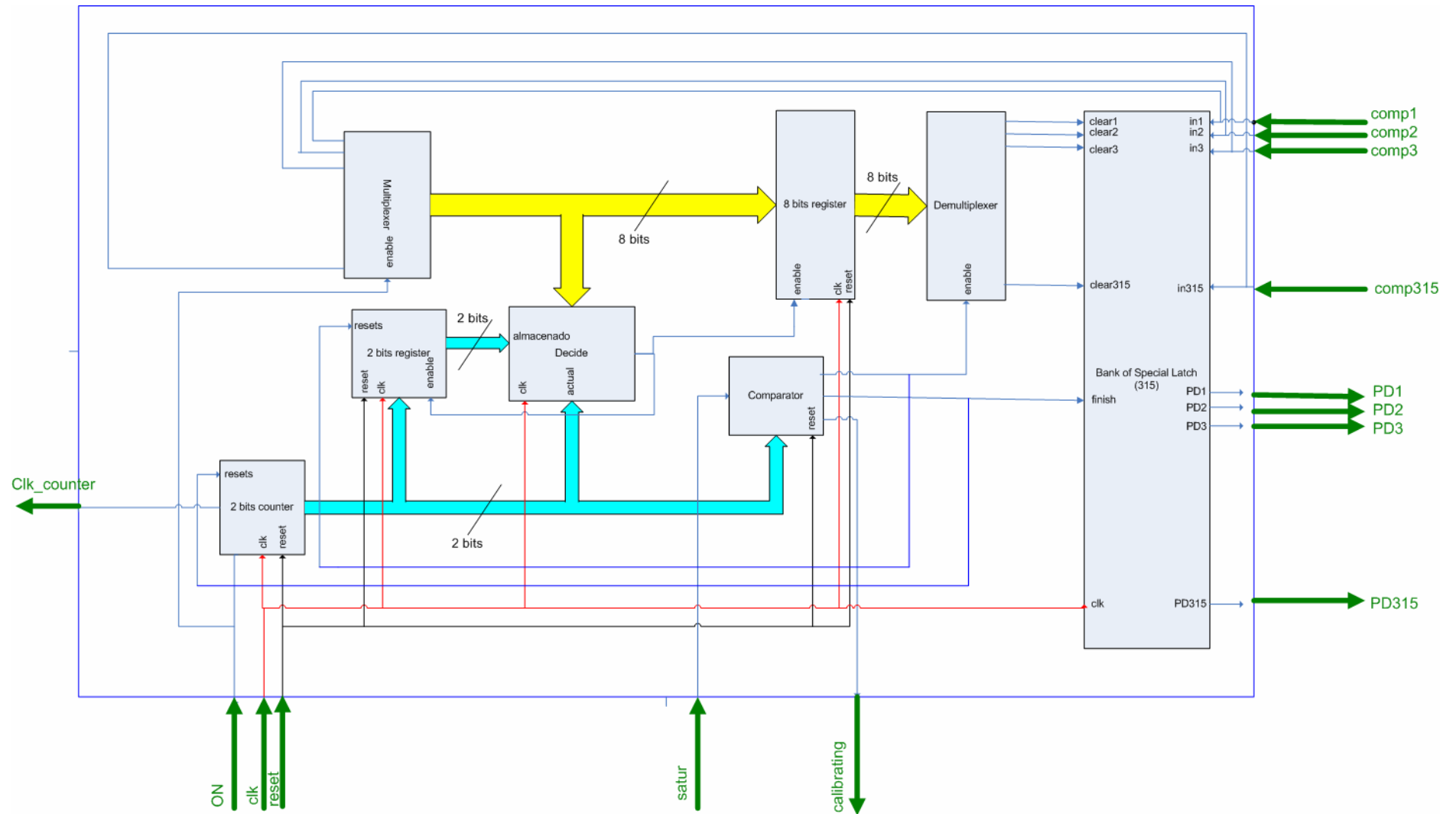


Figura 6.2. Arquitectura interna del bloque de calibración digital

Como podemos ver resumido en la tabla, el comportamiento de los latches es el siguiente. Inicialmente por defecto en estado de reset tenemos  $PD=0$ , es decir, todos los comparadores habilitados y  $IN=0$ . Cuando se produce un cambio en la salida de un comparador tendremos  $IN=1$ , con lo que  $PD$  pasará a  $1$ , deshabilitándose el comparador. Como el comparador se ha deshabilitado, su salida pasará a ser  $0$  de nuevo con lo que  $IN=0$ , pero ahora sin embargo  $PD$  sigue permaneciendo a  $1$  indicando que ese comparador permanecerá deshabilitado. En el momento que termina un ciclo de calibración, se activará el *clear* de algún comparador, para dejarlo como elegido. Cuando  $clear=1$ ,  $PD$  pasará a  $0$  dejando habilitado el comparador porque este será uno de los elegidos, pero ya, una vez que ha venido la señal *clear* nunca más se volverá a apagar este comparador, es decir, si ahora viniese  $IN=1$  otra vez,  $PD$  seguiría permaneciendo a  $0$  y no pasaría a  $1$  como ocurría al principio. Este comparador al que se le ha activado el *clear* será uno de los 63 elegidos para la conversión. Esta misma operación se repetirá en cada uno de los 63 niveles de referencia que tenemos. Cuando hemos terminado los 63 niveles de referencia,  $finish=1$ . Con esto, a todos los comparadores para los que nunca se haya activado su *clear* se pondrá  $PD=1$ , deshabilitándolos para el resto de su vida. Por el contrario, cuando se recibe *finish*, para los comparadores que hayan recibido alguna vez un *clear* se dejará su  $PD=0$ , indicando que son definitivamente los elegidos.

2. Multiplexor: A él irán conectadas también las salidas de los 315 comparadores. Se encargará de dar a su salida el número en 8 bits, del comparador que cambia en cada momento.
3. Demultiplexor: Realizará la tarea inversa al multiplexor comentado anteriormente. A su entrada tendrá el número de 8 bits de un comparador y a su salida obtendremos la activación de la señal *clear* del latch especial referente a ese comparador. Recordemos que se colocará un latch especial por cada comparador en el banco de latches.
4. Contador de 2 bits: Es un contador de 2 bits con la particularidad que estará continuamente realizando cuentas ascendentes y descendentes desde 0 a 3, empezando por la descendente, es decir contará: 3-2-1-0-0-1-2-3-2..... .
5. Decide: Es un bloque de decisión de dos entradas y una salida, que consiste en que, a su entrada tendrá el número de comparador que cambia en cada momento, así como el valor actual del contador de 2 bits y el valor almacenado del contador de 2 bits para el que se produjo el último cambio de un comparador. Si el número de comparador es distinto de cero comparará el valor actual del contador de 2 bits con el valor almacenado, si el actual es menor que el almacenado se activará la señal de salida.

6. Comparador: Activará las señales *finish* de los latches especiales y la señal *calibrating* una vez que el contador del generador de rampa haya saturado y el contador de 2 bits haya llegado a 3. La señal *calibrating* se activará un ciclo después que la señal *finish*. Cada vez que el reloj de 2 bits llegue a 3, este bloque se encargará de activar la señal *enable* del demultiplexor para activar el *clear* correspondiente al comparador con el que nos vamos a quedar para ese valor de referencia. También dará un reset al registro de 2 bits para que no haya nada almacenado al empezar el nuevo valor de referencia.
7. Registros de 8 y 2 bits: Servirán para almacenar el número de comparador elegido para cada nivel de referencia y el número del contador de 2 bits en el que se ha producido su cambio respectivamente.

El funcionamiento del bloque de calibración basado en estos bloques es bastante intuitivo, y lo detallaremos a continuación.

Cuando encendemos nuestro convertidor, se activará la señal *ON*, activando un contador de 2 bits. Este contador de 2 bits realizará cuentas ascendentes y descendentes, de forma que el paso por cero coincida con un flanco de subida del reloj *clk\_counter*, que producirá un paso en la cuenta del contador de 7 bits del generador de rampa (ver figura 6.5).

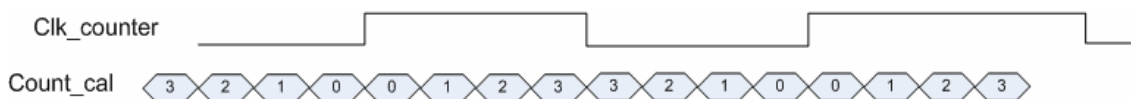


Figura 6.5. Funcionamiento del contador de 2 bits

El objetivo será escoger los comparadores que cambien lo más cerca posible del flanco de subida de *Clk\_counter*, es decir, para cada referencia de tensión (cada ciclo de *Clk\_counter*), cogeremos el comparador que ha cambiado con un valor mínimo en *Count\_cal*, es decir, aquel que ha cambiado más cerca del valor 0 del contador de 2 bits. Esto asegurará que nos quedemos con el comparador de menor offset. Para elegir esto, tenemos un bloque que se llama “*decide*” que tendrá a su entrada el número del comparador que cambia en cada momento, de manera que, cuando ese número sea distinto de 0 (será cero en estado de reset), comparará el valor actual del contador de 2 bits con el valor de este contador almacenado para el último cambio de un comparador. Si el valor actual es menor que el valor almacenado, querrá decir que el comparador que ha cambiado ahora, lo ha hecho en un valor del contador de 2 bits más cercano de 0, con lo cual tendrá menor offset que el comparador que cambió antes, por esto se almacenará el nuevo valor del contador en el registro de 2 bits y el nuevo valor de comparador en el registro de 8 bits. Cuando termina un ciclo de *Clk\_counter*, quiere decir que pasaremos a elegir comparador para un nuevo valor de referencia. Por tanto, cuando el contador de 2 bits llega a tres en la cuenta ascendente, se terminará la

elección de comparador para un valor de referencia, por esto, cuando el contador de 2 bits llegue a tres en una cuenta ascendente, el bloque “*comparador*” hace tres cosas:

1. Habilita el demultiplexor para que se active el *clear* del comparador almacenado en el registro de 8 bits. Eso quiere decir que ese comparador será el seleccionado para ese nivel de referencia que ha sido testeado (será uno de los 63 comparadores elegidos).
2. Si la señal *satur* esta a “1”, resetea el contador de 2 bits, un ciclo de reloj después pone a “1” la señal *finish* de los comparadores y otro ciclo de reloj después pondrá la señal *calibrating* a “1” indicando que el proceso de calibración ha finalizado ya que ese era el último valor de referencia (indicado por la señal *satur*).

Cuando esto acaba, el contador de 2 bits reanuda una nueva cuenta descendente de manera que estamos ante la búsqueda de comparador para un nuevo nivel de referencia con lo que todo el proceso se vuelve a repetir.

Cuando terminen los 63 niveles de referencia, se activará la señal *satur* del bloque comparador. Un ciclo después de esto se activará la señal *finish* indicando a los latches que el proceso de calibración ha finalizado, con lo cual se deshabilitarán todos los comparadores menos los que han sido elegidos para cada nivel de referencia (los que han recibido un *clear* en algún momento). Un ciclo después de estos se activará la señal *calibrating* indicando que el proceso de calibración ha finalizado y que el convertidor está listo para convertir, con lo cual ahora la señal de entrada a los comparadores pasará a ser la entrada analógica del convertidor.



## 6.3 FUNCIONAMIENTO

El esquema completo de calibración, ha sido montado y simulado en VHDL. A continuación pasaremos a mostrar algunos ejemplos de simulaciones que nos muestran el correcto funcionamiento del diseño. En la figura 6.6 podemos ver el nombre dado a los principales buses y señales que intervienen en el esquema con objeto de que sea más fácil la interpretación de las gráficas obtenidas por parte del lector.

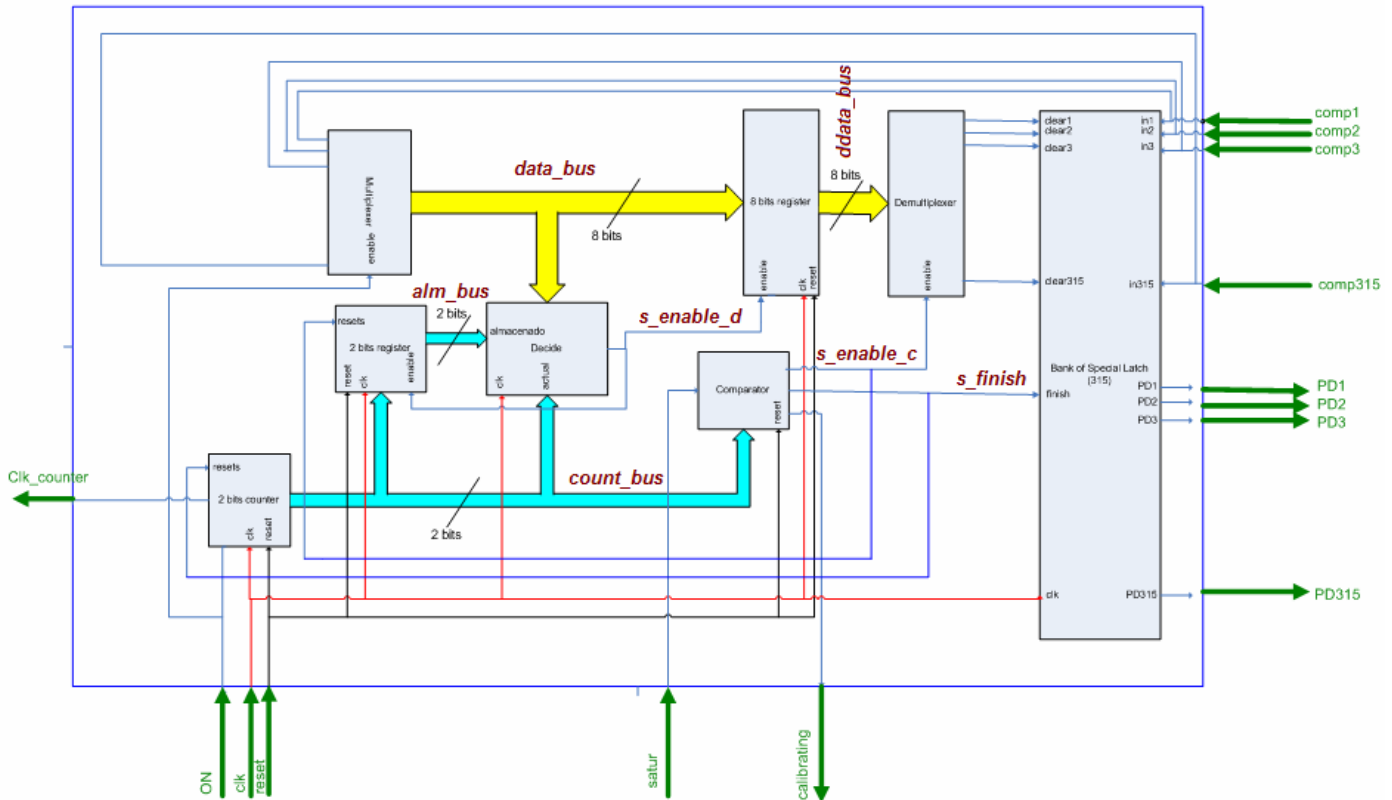


Figura 6.6. Señales en el esquema de calibración

La primera prueba consiste en probar el funcionamiento normal del esquema cambiando las salidas de los comparadores, es decir, cambiando (comp1....comp315). El resultado se muestra en la figura 6.7. Como podemos observar, en la señal **data\_bus** tenemos el número de comparador que cambia en cada momento. Como podemos ver, el comparador que ha cambiado más cerca del umbral es el comparador número 5, ya que ha cambiado en el valor más bajo de **count\_bus**. Por tanto, el valor 5 será el que debe de almacenarse en el registro de 8 bits. Para ello se activa **s\_enable\_d** y ya tenemos el número de comparador en la señal **ddata\_bus**. También se dará un reset al registro de 2 bits al final del ciclo para este valor de referencia con lo que la salida del registro de 2 bits debe de pasar a 0 para que no haya nada almacenado para el siguiente valor de referencia (**alm\_bus**). Como vemos en la figura, cuando el contador de 2 bits termina su cuenta ascendente, se activará el enable del demultiplexor (**s\_enable\_c**) activándose el clear (**s\_clear**) del comparador almacenado en el registro de 8 bits, en este caso el

comparador 5, indicando que este comparador será el elegido para este nivel de referencia.

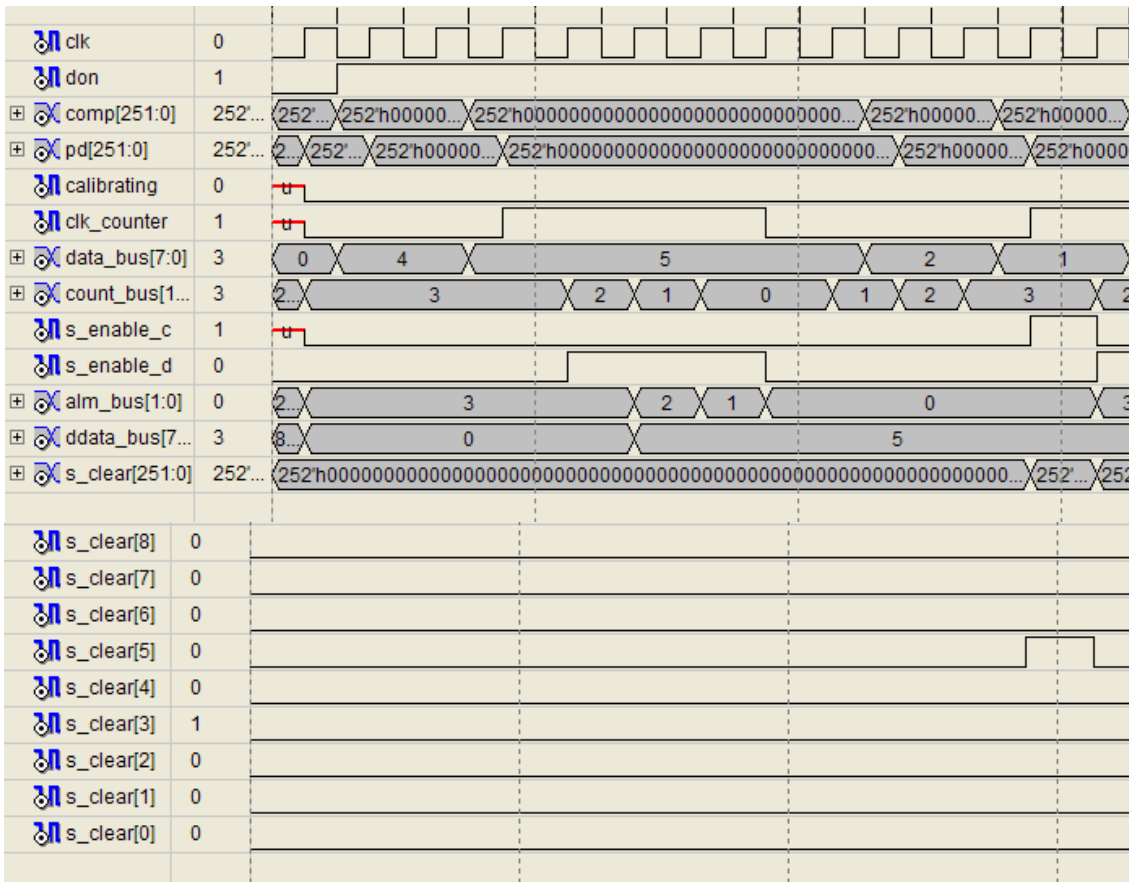


Figura 6.7. Fase de selección de comparadores

Cuando la rampa del generador de rampa llega a su fin, éste mandara la señal *satur* al bloque de calibración indicando que la fase de calibración ha finalizado. Después de esto el bloque de calibración se esperará a que el contador de 2 bits llegue a 3, contando ascendientemente y una vez que ha llegado concluye el proceso de calibración activando la señal *s\_finish*, que desactivará todos los comparadores no seleccionados durante la fase de calibración dejando activados los que sí hayan sido seleccionados. Dos ciclos después se activará la señal *calibrating* indicando que el proceso de calibración ha concluido y dando paso ahora a la señal analógica a convertir.

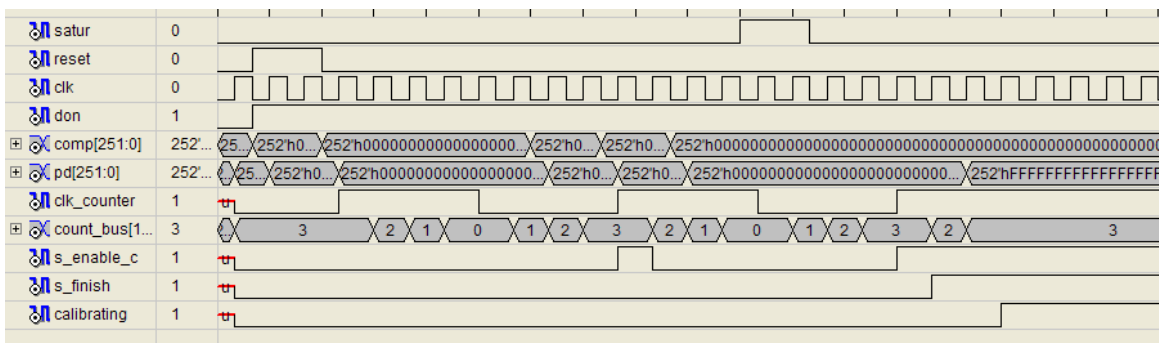


Figura 6.8. Finalización de la fase de calibración