

## **5.1 Introducción**

En la conexión entre la CPU y la memoria se han de cumplir las especificaciones temporales. Para ello no únicamente se han de contemplar que los tiempos de la memoria sean compatibles con los de la CPU, también se han de considerar los tiempos añadidos por los circuitos de decodificación de direcciones y cualquier otra lógica añadida.

Es importante tener en cuenta que los tiempos no son nunca valores exactos, siempre se especifican en forma de valores mínimos, típicos y/o máximos.

Cuando de un tiempo se proporciona únicamente el valor máximo, se ha de tener en cuenta que a veces el caso peor es el valor mínimo. En ese caso, a falta de especificación de valor mínimo, se ha de emplear un valor mínimo de cero.

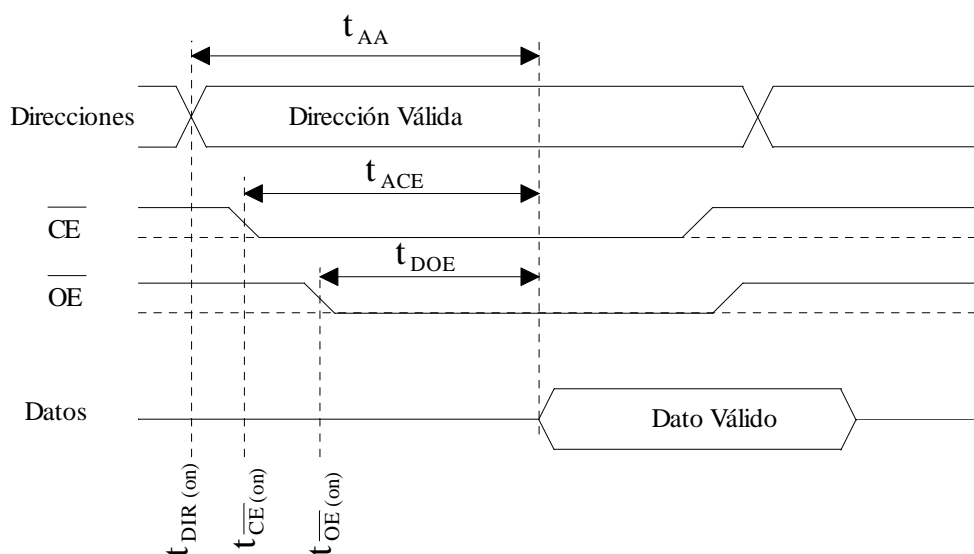
## **5.2 Tiempos en lectura**

En una memoria se define como tiempo de acceso en lectura el tiempo necesario para que la memoria ponga en el bus de datos el dato requerido a partir de los valores del bus de direcciones.

Este tiempo se puede dar a partir de tres instantes:

- $t_{AA}$  : Tiempo de acceso desde el cambio de las direcciones
- $t_{ACE}$  : Tiempo de acceso desde el cambio de la señal  $\overline{CE}^*$  ( $CS^*$ )
- $t_{DOE}$  : Tiempo de acceso desde el cambio de la señal  $\overline{OE}^*$ .

Los nombres que dan a estos tiempos los catálogos de las memorias pueden ser distintos.



Lo normal es que, para estos tiempos, el *datasheet* de la memoria especifique únicamente los tiempos máximos. Caso de que no se proporcionen valores mínimos, estos se han de considerar nulos.

Estos tiempos se han de cumplir todos ellos al mismo tiempo, eso significa que no habrá datos válidos hasta que no haya transcurrido el tiempo especificado desde la transición de cada una de las tres señales.

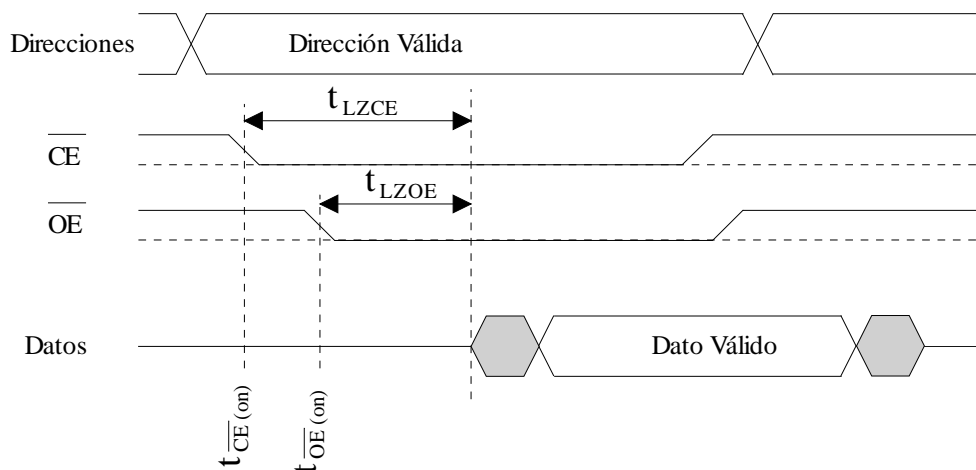
Si llamamos  $t_{DIR}$ ,  $t_{CE}$  y  $t_{OE}$  a los tiempos en que cambian las direcciones, la señal  $CE^*$  y la señal  $OE^*$  respectivamente, entonces, el instante máximo en que hay datos válidos se puede especificar como:

$$t_{DATO(max)} = \max \{ t_{DIR(on)} + t_{AA(max)} , t_{CE(on)} + t_{ACE(max)} , t_{OE(on)} + t_{DOE(max)} \}$$

### Alta y baja impedancia

La memoria, en estado inactivo, debe dejar el bus de datos en estado de alta impedancia para que otros dispositivos puedan hacer uso del bus.

Al final del tiempo de acceso deber haber pasado al estado de baja impedancia con un dato correcto en el bus. Sin embargo, es normal que el paso a baja impedancia se de antes de que el dato sea válido.



Los tiempos de control de bus especifican el tiempo que pasa desde el cambio de una señal de control hasta que el bus de datos pasa a baja impedancia:

- $T_{LZCE}$  : Tiempo *Low-Z* desde el cambio de la señal  $CE^*$  ( $CS^*$ )
- $T_{LZOE}$  : Tiempo *Low-Z* desde el cambio de la señal  $OE^*$ .

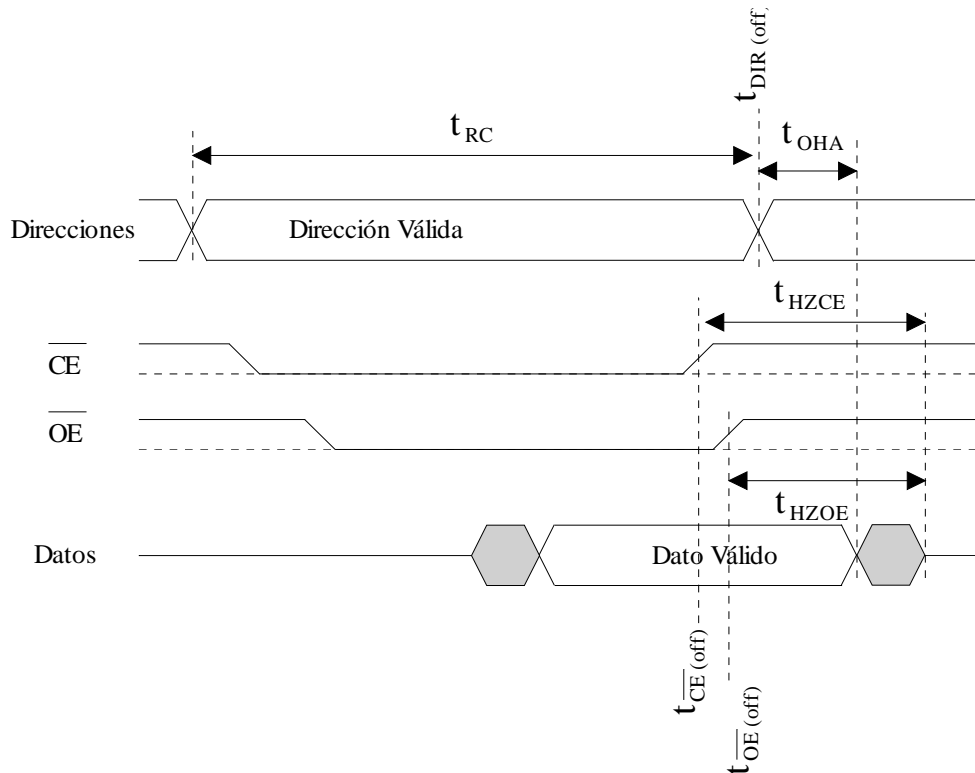
Nuevamente, como en el caso de los tiempos de acceso, **ambos** tiempos se han de satisfacer para que el bus pase a baja impedancia.

A partir de estos tiempos se puede determinar el tiempo mínimo en el que el bus deja de estar en alta impedancia:

$$t_{Low-Z(min)} = \max \{ t_{CE(on)} + t_{LZCE(min)} , t_{OE(on)} + t_{LZOE(min)} \}$$

### Finalización del ciclo

El ciclo de lectura se acaba cuando se desactivan sus señales de control: Direcciones, CE\* y OE\*.



Se pueden definir tiempos también respecto a la desactivación de las señales de control.

- $t_{OHA}$  : Tiempo (hold) desde cambio de direcciones a dato no válido.
- $t_{HZCE}$  : Tiempo desde desactivación de CE\* hasta High-Z.
- $t_{HZOE}$  : Tiempo desde desactivación de OE\* hasta High-Z.

El dato puede desaparecer debido a la desactivación de cualquiera de las señales, por tanto, el tiempo mínimo para que el dato deje de estar presente en el bus será, por tanto:

$$t_{DATO-OFF(min)} = \min \{ t_{DIR(off)} + t_{OHA(min)} , t_{CE(off)} + t_{HZCE(min)} , t_{OE(off)} + t_{HZOE(min)} \}$$

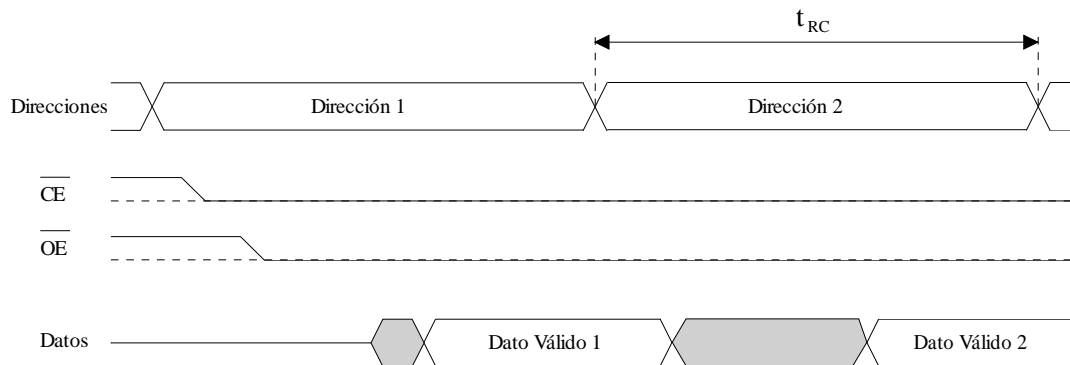
También puede interesar conocer el tiempo máximo antes de que el bus pase a estar nuevamente en alta impedancia. En este caso tendremos:

$$t_{BUS-HighZ(max)} = \min \{ t_{CE(off)} + t_{HZCE(max)} , t_{OE(off)} + t_{HZOE(max)} \}$$

Para memorias de más de 8 bits puede haber señales de entrada de *strobe* que permitan acceder sólo a alguno de los bytes del bus de datos. En estos casos, también habrá tiempos referidos a la activación y desactivación de estas señales.

Una vez vistos todos los tiempos se puede hablar también de un tiempo de ciclo de lectura  $t_{RC}$  que une todos los tiempos anteriores.

Si se hacen accesos de lectura consecutivos a diferentes posiciones de memoria del mismo chip, entonces no hace falta alzar las señales CS\* y OE\* antes de empezar un nuevo ciclo.



En definitiva, en lectura, la CPU da una orden, y la memoria responde a ella con los retardos descritos anteriormente. La validez o no del ciclo de lectura depende de que la CPU tenga la información en el tiempo requerido por ésta.

### 5.3 Tiempos en escritura

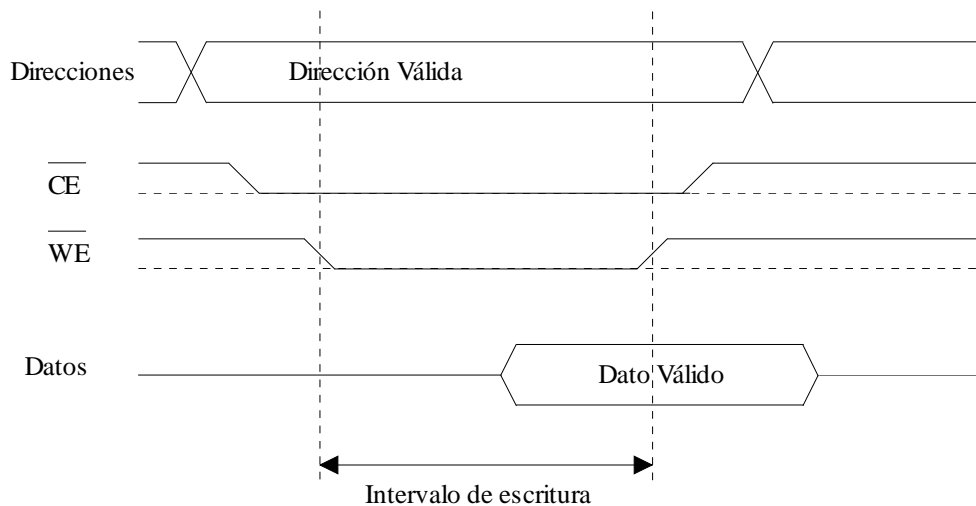
Los tiempos de lectura están presentes en todas las memorias tanto si son de lectura (ROM,PROM) como si son de lectura/escritura (RAM).

Los tiempos de escritura, lógicamente, sólo tienen sentido en las memorias RAM.

Los tiempos en escritura guardan algunas similitudes con los de lectura, pero hay dos diferencias importantes:

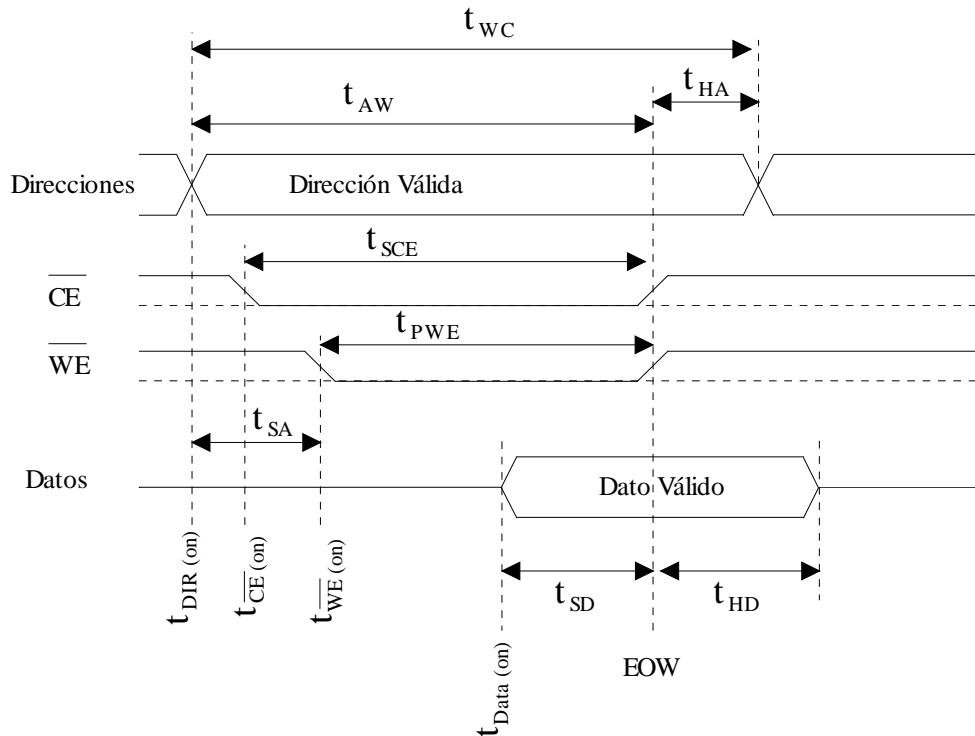
- La señal WE\* substituye a la señal OE\*. Dependiendo de la memoria, el estado de OE\* ha de ser inactivo o es indiferente mientras WE\* se halle activo. Para que se active la escritura tanto CS\* como WE\* deben hallarse activos.
- Los datos no los genera la propia memoria RAM sino la CPU. Hay un instante EOW (End of Write) en el que se acaba la escritura, pero la información debe de permanecer válida en el bus de datos durante un tiempo mínimo tanto antes como después de este instante.

A diferencia de la lectura, la validez de la escritura no se evalúa en la CPU sino en la propia memoria.



El tiempo disponible para realizar la escritura desde el momento en que la CPU activa todas las señales de control hasta el momento en que la escritura debe haber acabado se denomina **Intervalo de Escritura**.

Podemos observar el cronograma en escritura de manera detallada:



Podemos apreciar 6 instantes controlados por la CPU que son importantes para la temporización:

$t_{DIR(on)}$	Tiempo en que las direcciones son válidas.
$t_{CE(on)}$	Tiempo en que $CE^*$ se activa.
$t_{WE(on)}$	Tiempo en que $WE^*$ se activa.
$t_{Data(on)}$	Tiempo en que el dato es válido.
$t_{CE(off)}$	Tiempo en que $CE^*$ se desactiva.
$t_{WE(off)}$	Tiempo en que $WE^*$ se desactiva.

El tiempo en que acaba la escritura (EOW) es aquel en el que se desactiva alguna de las señales de control de la escritura  $WE^*$ ,  $CE^*$ . Por tanto, el instante EOW puede definirse como:

$$t_{EOW} = \min \{ t_{WE(off)} , t_{CE(off)} \}$$

En el cronograma estos dos tiempos se han mostrado iguales y coincidentes con  $t_{EOW}$ , pero en un caso práctico  $t_{WE(off)}$  y  $t_{CE(off)}$  pueden ser distintos y  $t_{EOW}$  coincidirá con el primero que ocurra de los dos.

$t_{EOW}$  tendrá cotas máximas y mínimas que podrán definirse:

$$t_{EOW(min)} = \min \{ t_{WE(off)(min)} , t_{CE(off)(min)} \}$$

$$t_{EOW(max)} = \min \{ t_{WE(off)(max)} , t_{CE(off)(max)} \}$$

Desde el cambio de las direcciones y la activación de WE\* y CE\* han de pasar unos tiempos (mínimos), específicos de la memoria, hasta el instante EOW:

$t_{AW}$	Tiempo desde que las direcciones son válidas.
$t_{SW}$	Tiempo desde la activación de CE*.
$t_{PWE}$	Tiempo desde la activación de WE*.

De ello se tendrán que cumplir las inecuaciones:

$$t_{AW} \leq \min(t_{EOW} - t_{DIR(on)}) \Rightarrow t_{AW} \leq t_{EOW(min)} - t_{DIR(on)(max)}$$

$$t_{SCE} \leq \min(t_{EOW} - t_{CE(on)}) \Rightarrow t_{SCE} \leq t_{EOW(min)} - t_{CE(on)(max)}$$

$$t_{PWE} \leq \min(t_{EOW} - t_{WE(on)}) \Rightarrow t_{PWE} \leq t_{EOW(min)} - t_{WE(on)(max)}$$

*Todos los tiempos han de ser menores que una determinada cota. Ello implica usar valores mínimos para los términos positivos y máximos para los negativos.*

Si la memoria requiere que OE\* sea alta, también habrá un tiempo  $t_{OEW}$  mínimo desde que OE\* se desactivó.

La escritura de la memoria concluye en el instante EOW, pero las direcciones y los datos deben ser válidos durante un tiempo antes y después de este instante:

$t_{SD}$	Tiempo de setup de los datos.
$t_{HD}$	Tiempo de hold de los datos.

De ello:

$$t_{SD} \leq \min(t_{EOW} - t_{Data(on)}) \Rightarrow t_{SD} \leq t_{EOW(min)} - t_{Data(on)(max)}$$

$$t_{HD} \leq \min(t_{Data(off)} - t_{EOW}) \Rightarrow t_{HD} \leq t_{Data(off)(min)} - t_{EOW(max)}$$

También hay unos requisitos respecto de las direcciones, que deben estar activas antes de que se activen las señales de control y deben permanecer activas también después del instante EOW:

$t_{SA}$	Tiempo de setup de las direcciones.
$t_{HA}$	Tiempo de hold de las direcciones.

De ello:

$$t_{AS} \leq \min(t_{WE(on)} - t_{DIR(on)}) \Rightarrow t_{AS} \leq t_{WE(on)(min)} - t_{DIR(on)(max)}$$

$$t_{HA} \leq \min(t_{DIR(off)} - t_{EOW}) \Rightarrow t_{HA} \leq t_{DIR(off)(min)} - t_{EOW(max)}$$

Finalmente, de manera similar al caso de las lecturas, hay un tiempo para el ciclo total de escritura que es  $t_{WC}$ .

La definición de los tiempos puede variar de una memoria a otra, e incluso pueden aparecer señales distintas. Por tanto, las definiciones anteriores pueden considerarse una referencia pero no son, necesariamente, las únicas que deben considerarse.

Igual que en el caso de las lecturas, se pueden hacer escrituras consecutivas en una memoria sin desactivar entre ellas la señal  $CE^*$ . No obstante, a diferencia de las lecturas, siempre tiene que haber un instante EOW de finalización de escritura antes y después del cual los datos deben ser válidos. Por tanto, la señal  $WE^*$  debe activarse y desactivarse para cada dato escrito.

## **5.4 Evaluación de tiempos en lectura**

Los apartados anteriores explican las condiciones a cumplir desde el punto de vista de la memoria. En este apartado y en el siguiente se estudiará si una CPU es compatible con las especificaciones de una memoria dada. El resultado de la evaluación será **OK** si se cumplen todos los tiempos y **KO** si se incumple alguno.

En lectura, la CPU da una orden y la memoria responde a ella con sus retardos. La evaluación de los tiempos normalmente consiste en comprobar que la memoria haya tenido tiempo suficiente de entregar los datos en el tiempo requerido.

Para comprobar que una memoria cumple con las especificaciones temporales en lectura, se ha de tomar el cronograma de tiempos de la CPU.

A partir de ello se han de obtener los tiempos en los que cada señal se activa. Estos tiempos pueden guardar relación todos con una referencia inicial, o pueden referirse a los ciclos de reloj. En este último caso se deberá considerar la frecuencia de reloj para hacer los cálculos.

Los tiempos importantes suelen ser:

$t_{CPU-DIR(on)}$	Tiempo en que el bus de direcciones es válido.
$t_{CPU-DIR(off)}$	Tiempo en que el bus de direcciones deja de ser válido.
$t_{R/W^*(H)}$	Tiempo en el que la señal $R/W^*$ pasa a modo lectura (alta).
$t_{R/W^*(L)}$	Tiempo en el que la señal $R/W^*$ puede pasar a escritura (baja).
$T_{strobe(on)}$	Tiempo en que se activa el <i>strobe</i> para el bus.
$T_{strobe(off)}$	Tiempo en que se desactiva el <i>strobe</i> para el bus.
$t_{CPU-lectura}$	Instante de lectura de la CPU.
$t_{DS}$	Tiempo de setup para el dato.
$t_{DH}$	Tiempo de hold para el dato.

De todos estos tiempos, dependiendo de lo que se desee calcular, puede ser interesante conocer valores máximos o mínimos.



El objetivo principal es garantizar que el dato está presente en el bus desde antes del tiempo de setup a después del tiempo de hold. Ambos referidos al tiempo en que la CPU lee el bus de datos.

A partir de los tiempos de la CPU, se ha de saber cuando la información llega a la memoria. Ello incluye los tiempos de buffers, decodificadores de direcciones y cualquier otra lógica.

De ello podemos hallar los instantes referidos a la memoria:

$$\begin{aligned} &t_{\text{DIR}(\text{on})}, t_{\text{DIR}(\text{off})} \\ &t_{\text{CE}(\text{on})}, t_{\text{CE}(\text{off})} \\ &t_{\text{OE}(\text{on})}, t_{\text{OE}(\text{off})} \end{aligned}$$

Una vez conocidos los tiempos referidos a la memoria, se han de usar los tiempos de la memoria para hallar el máximo tiempo que puede pasar para que el dato esté disponible  $t_{\text{DATO}(\text{max})}$  tal y como se explicó en 5.2.

Se deberá cumplir, el tiempo de *setup* para la CPU:

$$t_{\text{CPU-DS}(\text{max})} < t_{\text{CPU-Lectura}(\text{min})} - t_{\text{DATO}(\text{max})}$$

Si el tiempo anterior no se cumple, normalmente se pueden añadir un número  $n$  ciclos de espera hasta que se cumpla:

$$t_{\text{CPU-DS}(\text{max})} < t_{\text{CPU-Lectura}(\text{min})} - t_{\text{DATO}(\text{max})} + n \cdot T$$

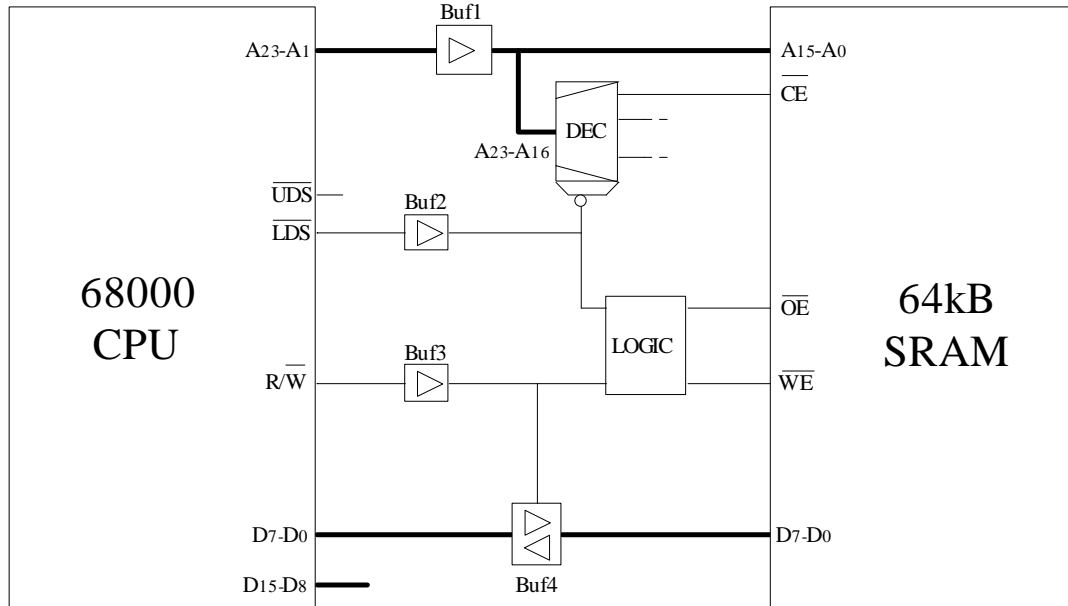
Adicionalmente, deberá cumplirse el tiempo de hold de la CPU, por tanto:

$$t_{\text{CPU-DH}(\text{max})} < t_{\text{DATO-OFF}(\text{min})} - t_{\text{CPU-Lectura}(\text{max})}$$

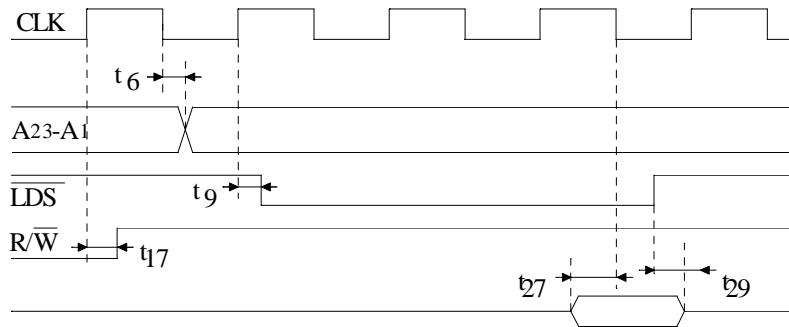
### Ejemplo:

Deseamos conectar un  $\mu P$  de la familia 68000 a dos chips de RAM de 64kB. Por simplicidad, en la figura sólo se muestra el chip que corresponde a las direcciones pares (controlado por LDS\*).

Por simplicidad se ha retirado del esquema la gestión de la señal DTACK\*.



La siguiente figura muestra los tiempos dependientes de la CPU.



De ello podemos saber los tiempos en que las señales llegarán a la SRAM. Tomaremos como referencia de tiempo 0 el primer flanco de subida de CLK.

$$t_{DIR(on)} = \frac{T}{2} + t_6 + t_{Buf1}$$

$$t_{CE(on)} = \left\{ \frac{T}{2} + t_6 + t_{Buf1} + t_{DEC(i/o)}, T + t_9 + t_{Buf2} + t_{DEC(sel)} \right\}$$

$$t_{OE(on)} = \left\{ T + t_9 + t_{Buf2}, t_{17} + t_{Buf3} \right\} + t_{LOGIC}$$

Como puede observarse, las señales CE\* y OE\* dependen de dos caminos distintos por lo que hay dos posibles tiempos. Dependiendo del caso tomaremos el valor máximo o el mínimo de ambos valores.

A partir de los tiempos de la SRAM podemos saber cuando estará disponible el dato, tal y como se describió en el apartado 5.2, pero añadiendo el tiempo asociado al bufer 4.

$$t_{DATO} = \max\{t_{DIR(on)} + t_{AA}, t_{CE(on)} + t_{ACE}, t_{OE(on)} + t_{DOE}\} + t_{Buf4}$$

Para que el tiempo anterior sea correcto, no obstante, se ha de cumplir que el tiempo de selección del buffer no condicione el retardo por lo que se deberá cumplir:

$$t_{DATO} > t_{17} + t_{Buf3} + t_{Buf4(sel)}$$

Una vez que sabemos cuando estará disponible el dato, podemos hallar si se cumple el tiempo de set-up ( $t_{27}$ ).

$$t_{27} < 3,5T - t_{DATO} \Rightarrow t_{27} < 3,5T - t_{DATO(max)}$$

Dado que la restricción para  $t_{DATO}$  es de máximo, entonces, para calcular  $t_{DATO(max)}$  emplearemos:

$$\begin{aligned} t_{DIR(on)MAX} &= \frac{T}{2} + t_{6MAX} + t_{Buf1MAX} \\ t_{CE(on)MAX} &= \max\left\{\frac{T}{2} + t_{6MAX} + t_{Buf1MAX} + t_{DEC(i/o)MAX}, T + t_{9MAX} + t_{Buf2MAX} + t_{DEC(sel)MAX}\right\} \\ t_{OE(on)MAX} &= \max\left\{T + t_{9MAX} + t_{Buf2MAX}, t_{17MAX} + t_{Buf3MAX}\right\} + t_{LOGICMAX} \\ t_{DATO(max)} &= \max\left\{t_{DIR(on)MAX} + t_{AA(max)}, t_{CE(on)MAX} + t_{ACE(max)}, t_{OE(on)MAX} + t_{DOE(max)}\right\} + t_{Buf4MAX} \end{aligned}$$

El cronograma también muestra que se ha de mantener un tiempo de hold  $t_{29}$  desde que la CPU desactiva LDS\*.

La señal LDS\* da lugar a cambios en las señales CE\* y OE\*.

$$t_{CE(off)} = t_{LDS(off)} + t_{Buf2} + t_{DEC(sel)}$$

$$t_{OE(off)} = t_{LDS(off)} + t_{Buf2} + t_{LOGIC}$$

A su vez, las especificaciones de la SRAM deben indicar el tiempo desde que se desactivan estas señales hasta que el dato deja de ser válido (paso a alta impedancia).

Tal y como vimos en 5.2:

$$t_{BUS-HighZ} = \min\{t_{CE(off)} + t_{HZCE}, t_{OE(off)} + t_{HZOE}\}$$

de ello, y añadiendo el tiempo asociado al bufer 4, tendremos:

$$t_{DATO(off)} = \min\{t_{LDS(off)} + t_{Buf2} + t_{DEC(sel)} + t_{HZCE}, t_{LDS(off)} + t_{Buf2} + t_{LOGIC} + t_{HZOE}\} + t_{Buf4}$$

A partir de ello podemos evaluar si se cumple el tiempo de hold  $t_{29}$ :

$$t_{29} < t_{DATO(off)} - t_{LDS(off)}$$

Substituyendo el valor de  $t_{DATO(off)}$  tendremos:

$$t_{29} < \min\{t_{Buf2} + t_{DEC(sel)} + t_{HZCE} , t_{Buf2} + t_{LOGIC} + t_{HZOE}\} + t_{Buf4}$$

Dado que se trata de un mínimo, los valores positivos deberán ser tomados como mínimos:

$$t_{29} < \min\{t_{Buf2\ MIN} + t_{DEC(sel)\ MIN} + t_{HZCE\ MIN} , t_{Buf2\ MIN} + t_{LOGIC\ MIN} + t_{HZOE\ MIN}\} + t_{Buf4\ MIN}$$

Podemos dar por buena la temporización (OK) si se cumple esta especificación para  $t_{29}$  y la anteriormente descrita para  $t_{27}$ .

## **5.5 Evaluación de tiempos en escritura**

Igual que en el caso de lectura, para comprobar que una memoria cumple con las especificaciones temporales en lectura, se ha de tomar el cronograma de tiempos de la CPU.

Nuevamente se han de tomar los tiempos de las acciones de la CPU:

$t_{\text{CPU-DIR}(\text{on})}$	Tiempo en que el bus de direcciones es válido.
$t_{\text{CPU-DIR}(\text{off})}$	Tiempo en que el bus de direcciones deja de ser válido.
$t_{\text{R/W}^*(\text{L})}$	Tiempo en el que la señal R/W* pasa a modo escritura (baja).
$t_{\text{R/W}^*(\text{H})}$	Tiempo en el que la señal R/W* puede pasar a escritura (alta).
$T_{\text{strobe}(\text{on})}$	Tiempo en que se activa el <i>strobe</i> para el bus.
$T_{\text{strobe}(\text{off})}$	Tiempo en que se activa el <i>strobe</i> para el bus.

El objetivo principal es garantizar que el dato está presente en el bus desde antes del tiempo de setup a después del tiempo de hold, pero en este caso de la memoria. Ambos referidos al tiempo en que la memoria lee el bus de datos (EOW).

A partir de los tiempos de la CPU, se ha de saber cuando la información llega a la memoria. Ello incluye los tiempos de buffers, decodificadores de direcciones y cualquier otra lógica.

De ello podemos hallar los instantes referidos a la memoria:

$t_{\text{DIR}(\text{on})}$ ,  $t_{\text{DIR}(\text{off})}$   
 $t_{\text{CE}(\text{on})}$ ,  $t_{\text{CE}(\text{off})}$   
 $t_{\text{WE}(\text{on})}$ ,  $t_{\text{WE}(\text{off})}$   
 $t_{\text{Data}(\text{on})}$ ,  $t_{\text{Data}(\text{off})}$

Una vez conocidos los tiempos referidos a la memoria, se han de usar los tiempos de la memoria para hallar el tiempo  $t_{\text{EOW}}$ .

Se deberá cumplir, el tiempo de setup para la memoria:

$$t_{\text{MEM-DS}(\text{max})} < t_{\text{EOW}(\text{min})} - t_{\text{Data}(\text{on})(\text{max})}$$

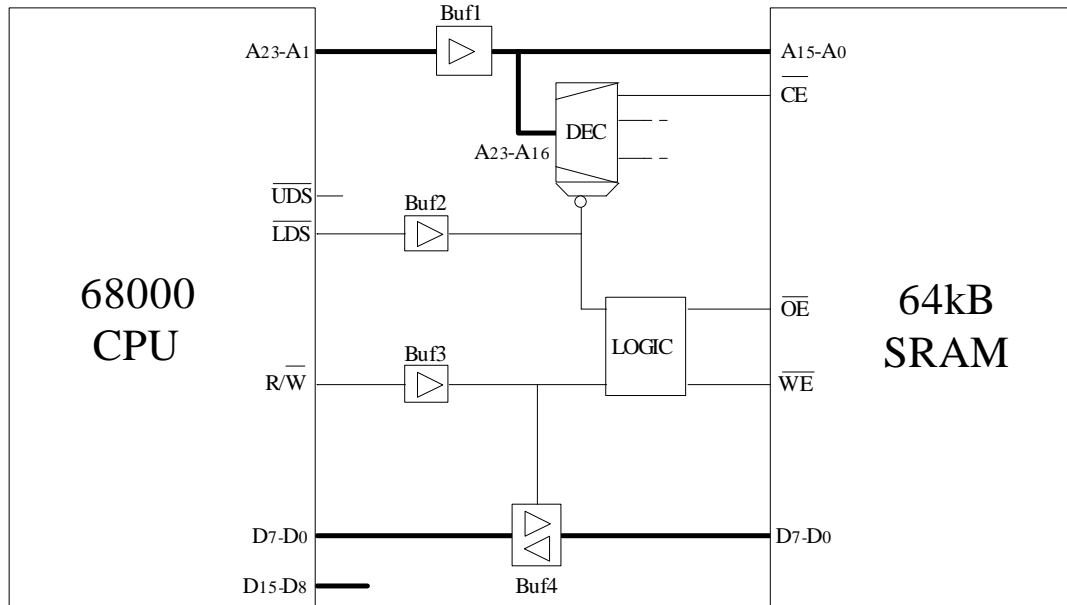
Adicionalmente, deberá cumplirse el tiempo de hold de la memoria, por tanto:

$$t_{\text{MEM-DH}(\text{max})} < t_{\text{Data}(\text{off})(\text{min})} - t_{\text{EOW}(\text{max})}$$

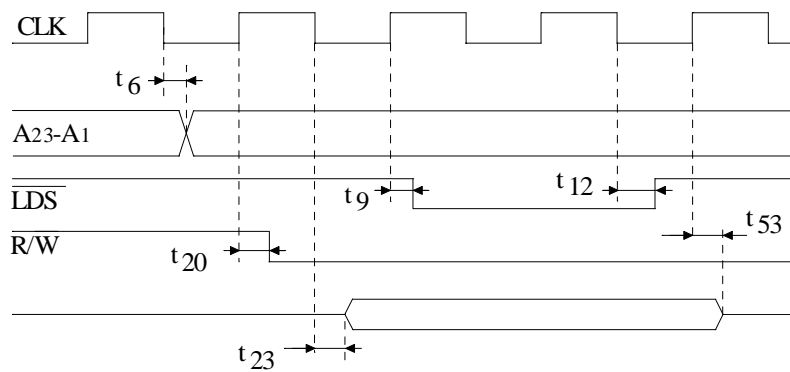
También tendrán que cumplirse los otros tiempos especificados.

### Ejemplo:

Deseamos verificar los tiempos de escritura del mismo ejemplo descrito para los tiempos de lectura.



La siguiente figura muestra los tiempos dependientes de la CPU para el ciclo de escritura.



De ello podemos saber los tiempos en que las señales llegarán a la SRAM. Tomaremos como referencia de tiempo 0 el primer flanco de subida de CLK.

$$\begin{aligned}
t_{DIR(on)} &= \frac{T}{2} + t_6 + t_{Buf1} \\
t_{CE(on)} &= \left\{ \frac{T}{2} + t_6 + t_{Buf1} + t_{DEC(i/o)} , 2T + t_9 + t_{Buf2} + t_{DEC(sel)} \right\} \\
t_{WE(on)} &= \left\{ 2T + t_9 + t_{Buf2} , T + t_{20} + t_{Buf3} \right\} + t_{LOGIC} \\
t_{DATA(on)} &= \left\{ 1,5T + t_{23} + t_{Buf4} , T + t_{20} + t_{Buf3} + t_{Buf4(sel)} \right\} \\
t_{DATA(off)} &= 4T + t_{53} + t_{Buf4} \\
t_{WE(off)} &= 3,5T + t_{12} + t_{Buf2} + t_{LOGIC} \\
t_{CE(off)} &= 3,5T + t_{12} + t_{Buf2} + t_{DEC(sel)}
\end{aligned}$$

Debemos hallar de lo anterior el tiempo EOW:

$$\begin{aligned}
t_{EOW} &= \min \{ t_{WE(off)} , t_{CE(off)} \} \\
t_{EOW} &= \min \{ 3,5T + t_{12} + t_{Buf2} + t_{LOGIC} , 3,5T + t_{12} + t_{Buf2} + t_{DEC(sel)} \}
\end{aligned}$$

La expresión anterior se puede simplificar:

$$t_{EOW} = 3,5T + t_{12} + t_{Buf2} + \min \{ t_{LOGIC} , t_{DEC(sel)} \}$$

A partir de los tiempos anteriores se han de verificar los tiempos de setup y hold de las direcciones:

Setup:  $t_{SD} < t_{EOW(min)} - t_{Data(on)(max)}$

$$\begin{aligned}
t_{SD} &< 3,5T + t_{12 MIN} + t_{Buf2 MIN} + \min \{ t_{LOGIC MIN} , t_{DEC(sel) MIN} \} \\
&\quad - \max \{ 1,5T + t_{23 MAX} + t_{Buf4 MAX} , T + t_{20 MAX} + t_{Buf3 MAX} + t_{Buf4(sel) MAX} \}
\end{aligned}$$

Hold:  $t_{HD} < t_{Data(off)(min)} - t_{EOW(max)}$

$$t_{HD} < 4T + t_{53 MIN} + t_{Buf4 MIN} - 3,5T + t_{12 MAX} + t_{Buf2 MAX} + \min \{ t_{LOGIC MAX} , t_{DEC(sel) MAX} \}$$

Igual que se han comprobado estos tiempos podrían comprobarse los otros 5 tiempos descritos para el ciclo de escritura. Para ello basta sustituir los tiempos requeridos dentro de cada expresión:

Tiempo suficiente desde el cambio de direcciones:

$$t_{AW} \leq t_{EOW(min)} - t_{DIR(on)(max)}$$

Tiempo suficiente desde la activación de CE\*:

$$t_{SCE} \leq t_{EOW(min)} - t_{CE(on)(max)}$$

Tiempo suficiente desde la activación de WE\*:

$$t_{PWE} \leq t_{EOW(\min)} - t_{WE(on)(\max)}$$

Tiempo suficiente desde que las direcciones son válidas (setup):

$$t_{SA} \leq t_{WE(on)(\min)} - t_{DIR(on)(\max)}$$

Tiempo suficiente de permanencia de las direcciones:

$$t_{HA} \leq t_{DIR(off)(\min)} - t_{EOW(\max)}$$

*Este último tiempo se cumple siempre en este ejemplo porque el cronograma de la CPU no indica la desactivación de las direcciones.*