

CAPÍTULO 3:

MODELO DE FUENTE DE CONTROL CON PENDIENTE SIMPLE.

PRIMERA PARTE:

- 3.1. *Introducción.*
- 3.2. *Modelo de fuente de control con pendiente simple.*
- 3.3. *Proceso de obtención y validación del modelo.*
- 3.4. *Paso cero. Cálculo matemático de las pendientes para V_x .*
 - 3.4.1. *Tensión de salida en función de V_x .*
 - 3.4.1.1. *Característica de subida.*
 - 3.4.1.2. *Característica de bajada.*
 - 3.4.2. *Obtención de t_r y t_f para la fuente de control V_x .*
 - 3.4.2.1. *Obtención de t_r para la fuente de control V_x .*
 - 3.4.2.2. *Obtención de t_f para la fuente de control V_x .*
- 3.5. *Paso 1: cálculo de valores característicos de cada modelo.*
 - 3.5.1. *V_{ol} y V_{oh} .*
 - 3.5.2. *C_{comp} .*
 - 3.5.3. *R_{load} (R_o).*
 - 3.5.4. *R_{lh} .*
 - 3.5.5. *R_{hl} .*
- 1. 3.6. *Paso 2: Funciones en MATLAB para realizar la operaciones.*
- 3.7. *Pasos 3 y 4: Simulaciones PSPICE con distintos modelos.*

SEGUNDA PARTE:

- 3.8. *Solución completa con MATLAB del cálculo de V_x .*
- 3.9. *Evolución en el tiempo de la salida respecto a V_x .*
 - 3.9.1. *Característica de subida.*
 - 3.9.2. *Característica de bajada*
- 3.10. *Funciones en MATLAB para el cálculo de t_r y t_f .*
- 3.11. *Ejemplo: resultados con el modelo DR-1.*

3.1. Introducción.

Como primer paso en el desarrollo de distintas estructuras capaces de modelar de manera general buffers de salida de circuitos integrados, se mostrará en este capítulo un modelo basado en un reducido número de componentes lineales, cuyos valores pueden ser obtenidos a partir de la información proporcionada en ficheros IBIS. Serán descritas las ventajas y limitaciones que esta estructura proporciona.

Se desarrollan dos métodos diferentes para solucionar el mismo problema, por lo que se estructura el capítulo en dos partes diferenciadas.

PRIMERA PARTE

3.2. Modelo de fuente de control con pendiente simple.

Se busca generar un modelo lo más sencillo posible que sea capaz de trabajar con las características más importantes del buffer que está siendo modelado. Estos valores, descritos en detalle más adelante, son los niveles entre los que conmuta el elemento (V_{ol} y V_{oh}), la impedancia de salida del circuito, caracterizada mediante las resistencias equivalentes del driver al conmutar de un nivel digital bajo a alto (R_{lh}) y viceversa (R_{hl}), así como la capacidad parásita de salida del driver (C_{comp}).

La siguiente figura describe la estructura utilizada, que a partir de ahora será referenciada como **modelo de fuente de control con pendiente simple**:

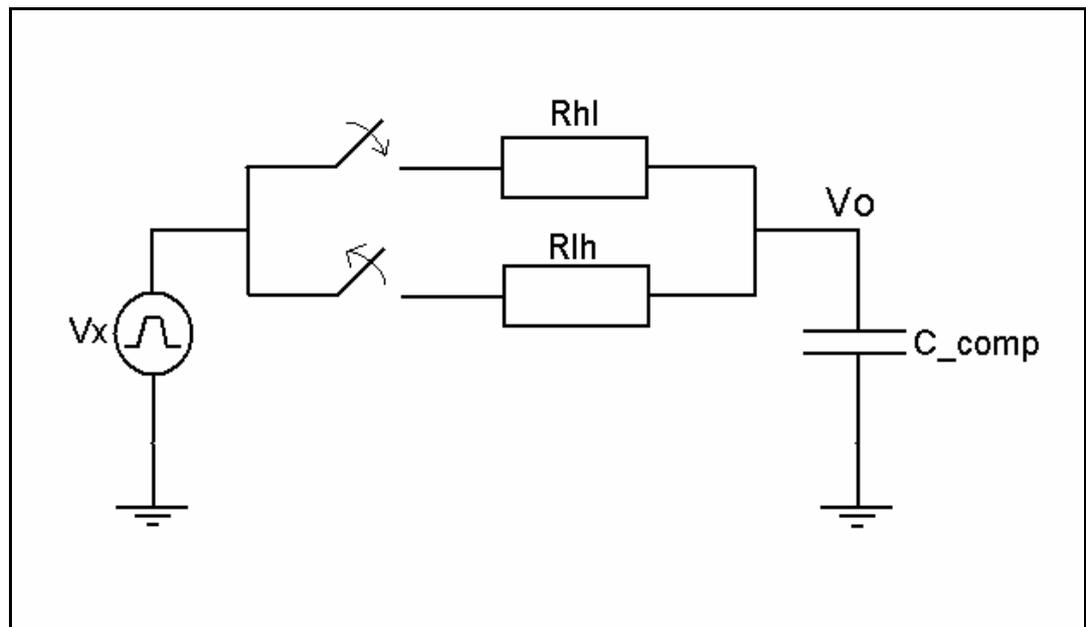


Figura 3.1: Modelo de fuente de control con pendiente simple.

3.3. Proceso de obtención y validación del modelo.

A continuación se exponen los pasos necesarios para conseguir caracterizar el modelo y posteriormente comprobar su validez, que se irán desarrollando a lo largo del capítulo:

0. Ya que el modelo posee una estructura fija y en el sólo cambian los valores de los componentes, es posible plantear un problema matemático sobre él. Consiste en **calcular la pendiente de subida y de bajada que ha de tener la fuente de tensión tipo pulso V_x** , para que a la salida del circuito los respectivos tiempos de subida y bajada correspondan con ciertas especificaciones descritas en el fichero IBIS. La manera en que son descritas las características de subida y bajada en la especificación IBIS son diferentes (carga R_o para subida y carga R_o en serie con fuente de tensión de valor V_{oh} para bajada), por lo que **se han de resolver independientemente dos problemas**, uno para el cálculo de la pendiente de subida y otro para la de bajada. Los interruptores del modelo son utilizados para forzar que el circuito trabaje con la resistencia R_{ih} o R_{hl} , según esté siendo generada la curva de subida o de bajada. Este problema se resuelve de manera general, por lo que sólo hay que hacerlo una vez.
1. A partir de las características del driver descritas en el fichero IBIS calcular los valores de V_{oh} , V_{ol} , R_{ih} , R_{hl} , R_o y C_{comp} .
2. Conociendo la expresión matemática de las pendientes y los valores concretos del resto de elementos del modelo, es posible crear en MATLAB funciones que devuelvan las pendientes de la fuente V_x , o lo que es lo mismo, los tiempos de subida (t_r) y bajada (t_f) de ésta.
3. Cuando el modelo tiene sus valores definidos es posible utilizar algún simulador de circuitos electrónicos (por ejemplo SPICE) para obtener la salida del circuito.

4. Por último, los resultados obtenidos se comparan con la información IBIS con objeto de determinar la calidad del modelo. Una comprobación más rigurosa, superponiendo la salida del modelo con la del circuito real, será realizada en el último capítulo.

En la figura 3.2. se muestra gráficamente el proceso descrito en anteriormente:

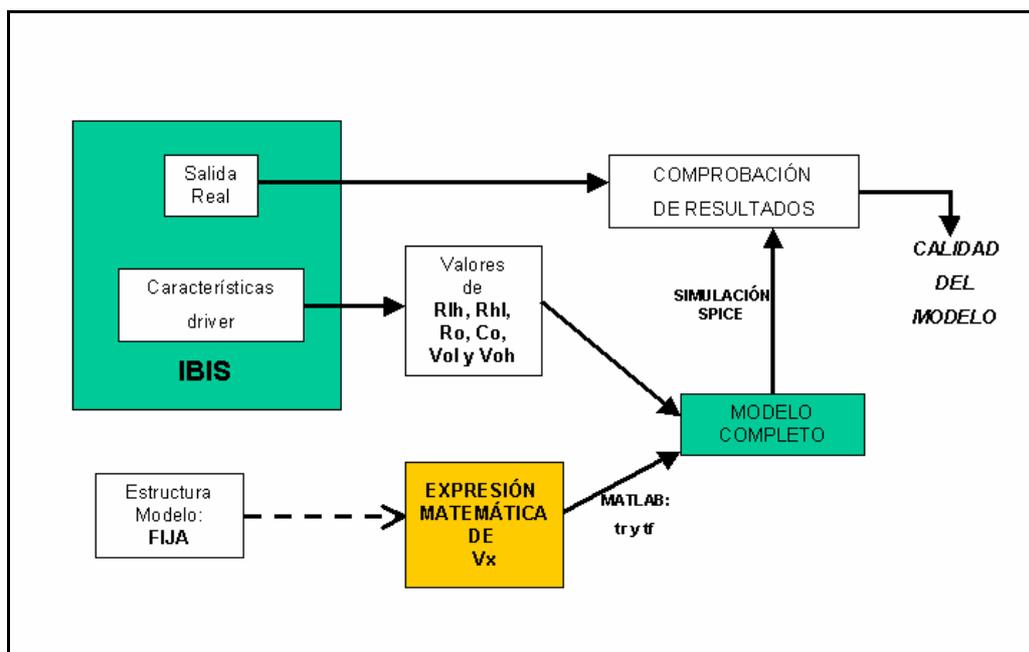


Figura 3.2: Proceso de obtención y validación del modelo.

Los siguientes apartados tienen como objeto explicar en detalle como se realizan cada una de estos pasos.

3.4. Paso 0: Cálculo matemático de las pendientes para V_x

El objetivo es controlar automáticamente el valor de los tiempos de subida y bajada en la fuente V_x , para que a la salida del circuito se emplee un tiempo determinado en variar un nivel de tensión también especificado. Estos valores de tiempo y tensión no serán aleatorios, sino que se relacionarán con la información descrita en la *keyword* **[Ramp]**.

[Ramp] indica el tiempo empleado por el circuito real en pasar de un 20% al 80% de su valor final con una carga y fuente de tensión externa especificada, dándose esta información como una fracción explícita, esto es, sin ser reducida. En esta *keyword* por tanto hay dos informaciones a este respecto:

- dV_r/dt_r : Para la característica de subida (raise).
- dV_r/dt_f : Para la característica de bajada (raise).

El **criterio** que se ha tomado es el de extrapolar los valores indicados en **[Ramp]**, de forma que dividiéndolos por 0.6 obtenemos el tiempo que se emplearía en pasar del valor inicial al valor final, y por lo tanto, una primera aproximación del tiempo que emplea nuestro circuito real en conmutar.

Para comprobar la validez de esta suposición se debe observar, siempre que sea posible, la información dada en las palabras reservadas **[Rising Waveform]** y **[Falling Waveform]** (estas *keywords* no son obligatorias en la norma), pues describen tablas de tiempo frente a voltaje en el circuito real.

De una manera muy general se puede decir que esta aproximación da lugar a tiempos de subida y bajada, de un 0 al 100% del valor final, más rápidos que los del circuito real que está siendo modelado, ya que corrientemente las formas de onda son más suaves cuando se encuentran próximas a los extremos de funcionamiento. Este hecho se comprobará en el último capítulo al comparar las salidas con un driver real.

3.4.1. Tensión de salida en función de V_x .

Se busca en primer lugar conocer la tensión a la salida del driver en función de la fuente de control V_x , para luego despejar y poder controlar V_x dependiendo de los valores de salida que deseemos obtener.

3.4.1.1. Característica de subida.

Al conmutar el driver del nivel inferior al superior, solamente se trabaja con la parte del modelo activa en esta fase, es decir, cuando ASWLH está cerrado. Además, es añadida la resistencia de carga R_o conectada a tierra, tal y como son obtenidos los valores de [Ramp] para la forma de onda de subida.

El circuito que ha de ser resuelto es el siguiente:

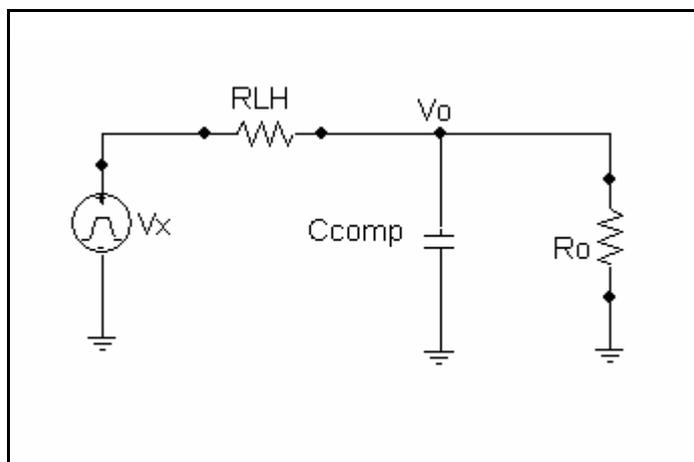


Figura 3.3: Modelo de subida con carga estándar (R_o).

Donde la fuente de control V_x tiene la forma mostrada en la figura 3.4:

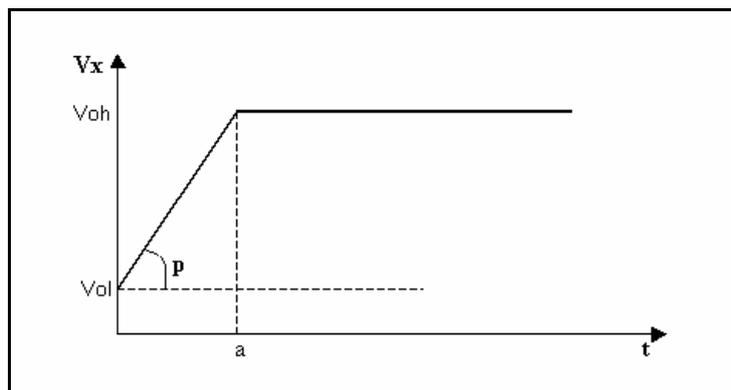


Figura 3.4: Fuente de control V_x para característica de subida.

Las ecuaciones que definen el comportamiento, sin considerar condiciones iniciales, son:

$$V_x(t) = Rlh \cdot it + V_o(t) = Rlh \cdot \left(C \cdot \frac{dV_o(t)}{dt} + \frac{V_o(t)}{R_o} \right) + V_o(t) \xleftrightarrow{\text{Laplace}}$$

$$V_x(s) = Rlh \cdot (C \cdot s \cdot V_o(s) + \frac{1}{R_o} \cdot V_o(s)) + V_o(s) \rightarrow$$

$$V_x(s) = V_o(s) \cdot \left[1 + \frac{Rlh}{R_o} + Rlh \cdot C \cdot s \right] \rightarrow$$

$$V_o(s) = \frac{\frac{1}{Rlh \cdot C} V_x(s)}{\frac{1}{Rlh \cdot C} \cdot \left(1 + \frac{Rlh}{R_o} \right) + s} = \frac{\alpha \cdot V_x(s)}{\beta + s}$$

Por otra parte se calcula la expresión de $V_x(s)$:

$$V_x(t) = p \cdot t \cdot u(t) - p \cdot (t - a) \cdot u(t - a) + Vol \cdot u(t) \xleftrightarrow{\text{Laplace}}$$

$$V_x(s) = \frac{p}{s^2} \cdot (1 + e^{-as}) + \frac{Vol}{s}$$

De esta forma:

$$Vo(s) = \frac{p \cdot \alpha \cdot [1 - e^{-as}]}{s^2 \cdot (\beta + s)} + \frac{\alpha \cdot Vol}{s \cdot (\beta + s)}$$

Tras operar y antitransformar se obtiene el resultado deseado:

$$Vo(t) = \left(\frac{\alpha \cdot Vol}{\beta} - p \cdot \gamma \right) \cdot [1 - e^{-\beta \cdot t}]$$

donde

$$\alpha = \frac{1}{Rlh \cdot C}$$

$$k = 1 + \frac{Rlh}{Ro}$$

$$vo^- = \frac{Vol}{k}$$

$$\beta = \alpha \cdot k$$

$$\gamma = \frac{1}{\beta \cdot k} \cdot (1 - e^{\beta \cdot a})$$

Ya con este resultado se pueden comprobar los valores iniciales y finales que están siendo modelados:

$$Vo(t=0) = 0$$

$$Vo(t \rightarrow \infty) = \frac{Ro}{Ro + Rlh} \cdot Voh$$

Estos valores son correctos :

- el problema se ha planteado sin condiciones iniciales, luego en el instante inicial ha de valer cero. vo^- corresponde al valor que el condensador ha de tener en el momento en que Vx empieza a conmutar. Este valor será

utilizado más adelante para la obtención la pendiente de subida del modelo.

Por su parte, en régimen permanente el condensador estará cargado, por lo que por él no circula corriente; de esta forma se tiene un divisor de tensión con un valor que se corresponde con el valor final obtenido.

3.4.1.2. Característica de bajada.

De la misma forma que en el anterior apartado, en este momento va a ser planteado y resuelto el cálculo de la ecuación de salida del modelo cuando conmuta entre un valor digital alto a uno bajo.

El circuito que ha de ser resuelto se muestra en la figura 3.5, en el que se puede ver que son añadidas la resistencia de carga R_o así como una fuente de alimentación de valor V_{oh} , conectadas tal y como indica la especificación IBIS para el cálculo de los valores de [Ramp].

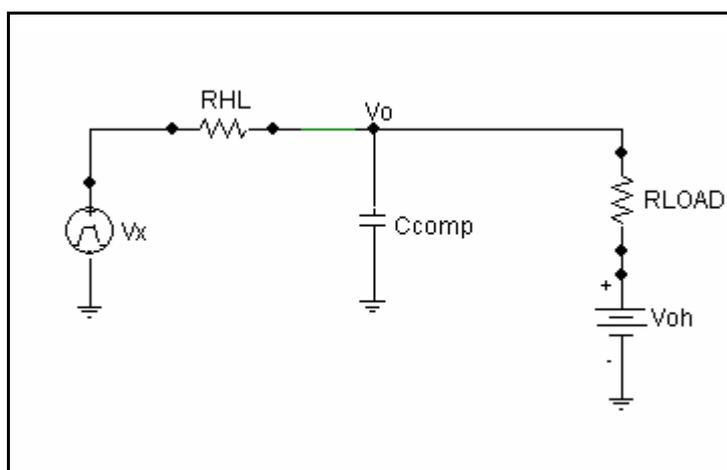


Figura 3.5: Modelo de bajada con carga estándar (R_{load} y V_{oh}).

De nuevo , si las condiciones iniciales no son consideradas las ecuaciones que definen el comportamiento del circuito son:

$$V_{oh} = R_o \cdot \left[\left(\frac{V_o - V_x}{R_{hl}} \right) + C \cdot \frac{dV_o(t)}{dt} \right] + V_o(t) \xrightarrow{\text{Laplace}}$$

$$V_o(s) \cdot \left[R_o \cdot C \cdot s + \left(1 + \frac{R_o}{R_{hl}} \right) \right] - \frac{R_o}{R_{hl}} \cdot V_x(s) = V_{oh}(s)$$

En este caso, la fuente V_x tendrá la siguiente forma indicada en la figura 3.6:

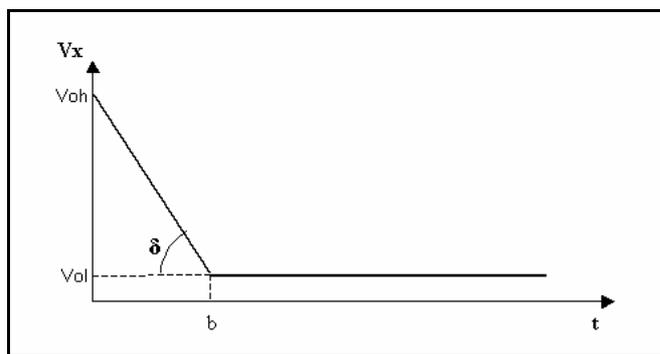


Figura 3.6: Fuente de control V_x para característica de bajada.

Así pues puede ser modelada mediante:

$$V_x(t) = -\delta \cdot t \cdot u(t) + V_{oh} \cdot u(t) + \delta \cdot (t - b) \cdot u(t) \xrightarrow{\text{Laplace}}$$

$$V_x(s) = \frac{\delta \cdot (e^{-b \cdot s} - 1) + s \cdot V_{oh}}{s^2}$$

De esta forma, sustituyendo y operando:

$$V_o(s) = \frac{\Omega \cdot \frac{V_{oh}}{s} + \tau \cdot \frac{\delta \cdot (e^{-b \cdot s} - 1)}{s^2}}{\Omega + s}$$

donde:

$$\tau = \frac{1}{Rhl \cdot C}$$

$$\Omega = \frac{1}{Ro \cdot C} \cdot \left(1 + \frac{Ro}{Rhl}\right)$$

y antitransformando:

$$Vo(t) = [Voh - \aleph] \cdot (1 - e^{-\Omega t})$$

donde:

$$\aleph = \frac{\tau \cdot \delta}{\Omega^2} \cdot (e^{\Omega \cdot b} - 1)$$

Con estos resultados se comprueban el valor inicial y final:

$$Vo(t)=0$$

$$Vo(t \rightarrow \infty) = Voh - \aleph = \frac{(Voh - Vol)}{(Ro - Rhl)} \cdot Rhl + Vol$$

De nuevo, estos resultados son correctos:

- El problema ha sido planteado sin condiciones iniciales, luego en el instante inicial el valor de salida ha de ser cero.

En régimen permanente el condensador ya está cargado, por lo que de nuevo se forma un divisor de tensión que en el caso más general corresponde con la última expresión matemática.

3.4.2. Obtención de t_r y t_f para la fuente de control V_x .

Una vez que se tienen las ecuaciones del comportamiento a la salida del modelo es posible imponer condiciones respecto al tiempo y el valor de la variación de tensión, de forma que a partir de esto sea posible despejar los tiempos de subida y bajada necesarios en la fuente V_x para conseguir estos resultados.

Como será mostrado a continuación, la solución exacta de este problema requeriría resolver una ecuación no algebraica, y debido a la complejidad de ésta, el consiguiente uso de cálculo numérico; por ello se hará uso de la aproximación de la exponencial en serie de Taylor.

3.4.2.1. Obtención de t_r para la fuente de control V_x .

De la diferencia entre el valor final y el inicial será despejado el valor del tiempo de subida. Ha de recordarse que el valor inicial viene dado por v_0^- , por lo que $t=0$ en este caso se considera ese instante, es decir, cuando el condensador ya ha quedado cargado a ese valor.

Si consideramos que:

- $W=dVr/0.6$
- $T=dtr/0.6$

Entonces:

$$\begin{aligned}
V_o(T) = V_o("0") &= \left(\frac{\alpha \cdot Vol}{\beta} - p \cdot \gamma \right) \cdot [1 - e^{-\beta T}] - \frac{Vol}{k} = W \rightarrow \\
&\{ 1 - e^{-\beta T} = L \} \rightarrow \\
\frac{-Vol}{k} \cdot e^{-\beta T} - p \cdot \frac{1}{\beta \cdot k} (1 - e^{\beta a}) \cdot L &= W \rightarrow \\
\left\{ e^x \approx 1 + x + \frac{x^2}{2} + \frac{x^3}{6} \right\} \\
\left\{ p = \frac{V_{oh} - Vol}{a} = \frac{V_d}{a} \right\} &\rightarrow \\
-\frac{Vol}{k} \cdot e^{-\beta T} + \frac{L \cdot V_d}{k} \cdot \left(1 + \frac{\beta \cdot a}{2} + \frac{\beta^2 \cdot a^2}{6} \right) &= W \rightarrow
\end{aligned}$$

$$a = tr = \frac{1}{2 \cdot \beta} \cdot \left(-3 \pm \sqrt{9 - 24 \cdot (1 - Y)} \right)$$

con

$$Y = \frac{W \cdot k + Vol \cdot e^{-\beta T}}{L \cdot V_d}$$

Siendo válida solamente la solución con signo positivo. Queda calculado el valor de tr.

3.4.2.2. Obtención de tf para la fuente de control Vx.

De nuevo haciendo la diferencia entre el valor en que empieza a conmutar el driver, esto es Voh, y el valor de tensión al transcurrir el intervalo de tiempo en que ha de conmutar el circuito, es posible calcular la pendiente de bajada para la fuente Vx; también se vuelven a encontrar los mismos problemas en la resolución de la ecuación que en el apartado anterior :

- W2 = dVf/0.6
- T2 = dtf/0.6

$$V_o(t="0") - V_o(T2) = V_{oh} - \left[V_{oh} - \frac{\tau \cdot \delta}{\Omega^2} \cdot (e^{\Omega b} - 1) \right] \cdot (1 - e^{-\Omega T2}) = W2 \rightarrow$$

$$\left\{ \delta = \frac{Vd}{b} \right\}$$

$$\left\{ L = 1 - e^{-\Omega T2} \right\}$$

$$\left\{ e^x \approx 1 + x + \frac{x^2}{2} + \frac{x^3}{6} \right\}$$

$$V_{oh} \cdot e^{-\Omega T2} + \frac{\tau \cdot Vd}{\Omega} \cdot \left(1 + \frac{\Omega \cdot b}{2} + \frac{\Omega^2 \cdot b^2}{6} \right) \cdot L = W2 \rightarrow$$

$$b = tf = \frac{1}{2\Omega} \cdot \left(-3 \pm \sqrt{9 - 24 \cdot (1 - J)} \right)$$

con

$$J = \frac{(W - V_{oh} \cdot e^{-\Omega T}) \cdot \Omega}{L \cdot \tau \cdot Vd}$$

En este caso el valor válido para tf es el negativo, ya que hemos trabajado con δ como valor positivo y no negativo. En el modelo se usará este valor cambiado de signo.

3.5. Paso 1: cálculo de valores característicos de cada modelo.

3.5.1. Vol y Voh.

Estos valores corresponden a los niveles entre los que conmuta la fuente de tensión y que, observando la estructura del modelo, van a definir también los niveles de salida del *driver* en ausencia de carga. Por ejemplo, trabajando con tecnología CMOS estos dos valores serían típicamente 0v y 5v.

En IBIS no está especificado que la tecnología deba ser explícitamente descrita, sin embargo podemos conseguir estos dos valores :

- Voh corresponde con el valor indicado en el valor típico (typ) de la *keyword* **[Voltage range]**, esto corresponde a la tensión de alimentación del circuito.
- Vol es obtenido como el valor de tensión asociado a la mínima corriente que circula a la salida del circuito cuando estamos caracterizando el comportamiento en *pull-down*. De esta forma, se debe identificar en la *keyword* **[Pulldown]** el mínimo en la columna *I(typ)*, y seleccionar a continuación el valor de *Voltage* que le corresponde.

En la gran mayoría de los circuitos actuales, este nivel es idénticamente cero.

3.5.2. C_comp

Este parámetro se encuentra directamente en la información que proporciona la especificación IBIS, ya que corresponde a la capacidad parásita final que encuentra el *buffer*, sin incluir la capacidad asociada al encapsulado.

Dentro de la *keyword* **[Model]**, se selecciona el valor típico (typ) de C_comp; por simplicidad, a partir de ahora y a lo largo de este capítulo C_comp será denominado simplemente C.

3.5.3. R_load (Ro)

Carga de salida con la que son calculados los valores para [Ramp], [Rising Waveform] y [Falling Waveform]. Por defecto su valor es 50 Ω, y si es distinto este valor se indica en la palabra reservada [Ramp].

3.5.4. Rlh

La obtención de este valor, así como el de Rhl, supone una mayor complejidad ya que es necesario calcularlo de una manera indirecta.

Rlh y Rhl deben corresponder a la resistencia equivalente que en régimen permanente encuentra nuestro driver. Esto muestra la existencia de dos maneras de calcular cada una de las resistencias. Respecto a Rlh :

1. Primero hemos de conocer el valor final de la salida a nivel alto, que lo podemos obtener como $V_{ol} + dV_r/0.6$. Acto seguido se calcula la diferencia entre V_{oh} y este valor, que corresponde a la tensión existente en régimen permanente en el transistor de pullup. A partir de la información de la tabla de **[pullup]**, se busca la corriente que circula para esta tensión (en la mayoría de los casos es necesario extrapolar) y calculando el cociente entre ambos se obtiene el valor de la resistencia Rlh.
2. En régimen permanente se forma un divisor de tensión de valor :

$$V_o(t \rightarrow \infty) = \frac{R_o}{R_o + R_{lh}} \cdot V_{oh}$$

(ver apartado 3.4.1.1). Conociendo el valor final, calculado como en el apartado anterior, es posible despejar Rlh:

$$Rlh = \frac{Ro \cdot Voh}{Vo(t \rightarrow \infty)} - Ro$$

Este último valor es más preciso que el obtenido en 1., resultado de una extrapolación, por lo que será el utilizado. Si estos dos métodos no proporcionan valores muy similares es posible que existan errores en el fichero IBIS con el que estamos trabajando¹, por lo que supone una comprobación siempre interesante.

3.5.5. Rhl

De manera equivalente al cálculo de Rlh, existen dos métodos para calcular Rhl:

1. El valor final del *driver* al conmutar de un nivel alto a uno bajo, con la carga descrita en [Ramp], es $Voh - dVf / 0.6$. En la curva de [Pulldown] se interpola esta tensión para poder obtener la corriente asociada. Dividiendo tensión entre corriente se calcula la resistencia Rhl.
2. Al conmutar del nivel alto al bajo, al alcanzarse el régimen permanente se forma un divisor de tensión cuyo valor final es (ver apartado 3.4.1.2):

$$Vo(t \rightarrow \infty) = Voh - \mathcal{N} = \frac{(Voh - Vol)}{(Ro + Rhl)} \cdot Rhl + Vol$$

¹ Esto no es nada extraño si no se trabaja con ficheros fiables, y muy frecuente con los ficheros IBIS obtenidos en Internet.

De forma que conociendo el valor final es posible despejar R_{hl} :

$$R_{hl} = R_o \cdot \frac{(V_o(t \rightarrow \infty) - V_{ol})}{V_{oh} - V_o(t \rightarrow \infty)}$$

Este segundo valor es más preciso pues no se realiza ninguna interpolación para conseguirlo, por lo que es el utilizado.

De nuevo, si los dos valores no coinciden es muy posible que el fichero IBIS contenga errores.

3.6. Paso 2: Funciones en MATLAB para realizar las operaciones.

Al haber obtenido las expresiones para t_r y t_f las funciones MATLAB resultan muy sencillas, y se limitan a sustituir los valores que se le pasan como entrada en la expresión correspondiente. Las funciones son las siguientes:

function [beta,tr,slopelh] = Np1lh100(Vol,Voh,Rlh,Ro,C,dVr,dtr)

*%This function is used to calculate the slope and the raise time
%for the voltage source Vx. Developed by Mario Palma.*

*alfa=1/(Rlh*C);*

k=1+(Rlh/Ro);

*beta=alfa*k;*

vini=Vol/k;

W=dVr/0.6;

T=dtr/0.6;

*L=1-exp(-beta*T);*

Vd=Voh-Vol;

%Now the slope and raise time are calculated

*Y=(k*W+Vol*exp(-beta*T))/(L*Vd);*

tr1=(-3+(9-24(1-Y))^0.5)/(2*beta);*

tr2=(-3-(9-24(1-Y))^0.5)/(2*beta);*

slopelh1=Vd/tr1;

slopelh2=Vd/tr2;

end;

function [tf,slopehl] = Np1hl100(Vol,Voh,Rhl,Ro,C,dVf,dtf)

*% This function is used to calculate the necessary Vx's falling
 % slope in the linear one-pin general model. Developed by
 % Mario Palma.*

$Tau=1/(Rhl*C);$

$omega=(1/(Ro*C))*(1+Ro/Rhl);$

$T=dtf/0.6;$

$W=dVf/0.6;$

$L=1-exp(-omega*T);$

$Vd=Voh-Vol;$

$v0=Voh-W;$

%Now the slope and the falling time are calculated:

$J=(omega*(W-Voh*exp(-omega*T)))/(L*Tau*Vd);$

$tf1=(-3+(9-24*(1-J))^0.5)/(2*omega);$

$tf2=(-3-(9-24*(1-J))^0.5)/(2*omega);$

$slopehl1=Vd/tf1;$

$slopehl2=Vd/tf2;$

end;

3.7. Pasos 3 y 4: Simulaciones PSPICE con distintos modelos.

Una vez que un modelo está completamente caracterizado mediante el método expuesto, es posible simular su comportamiento mediante SPICE y compararlo con la información IBIS, que describe a la perfección al driver real con el que se trabaja

A continuación se muestran los resultados obtenidos trabajando con diversos ficheros IBIS. El *DR-1a.ibs* se encuentra listado en el **apéndice 1**. El resto ha sido sacado de la página web oficial de IBIS, pudiéndose acceder a ellos gratuitamente; las últimas figuras muestran los resultados gráficos, obtenidos con PSPICE.

CARACTERÍSTICA DE SUBIDA				
Fichero IBIS	Compañía	Modelo	tr (seg.)	Error (A ó B)
<i>art360.ibs</i>	IBM	<i>padiol2</i>	7.7613e-12	A=-2.501%
<i>DR-1a.ibs</i>	SIEMENS	<i>DR-1a</i>	NO FUNCIONA	
<i>80960RD.ibs</i>	INTEL	<i>highobuffer</i>	2.923e-14	0 %

Tabla 3.1: Resultados de simulaciones con ficheros IBIS (característica de subida).

CARACTERÍSTICA DE BAJADA				
Fichero IBIS	Compañía	Modelo	tf (seg.)	Error (A ó B)
<i>64MSS3.ibs</i>	<i>SIEMENS</i>	<i>I/O_DQ</i>	<i>1.0604e-10</i>	<i>A=0.47%</i>
<i>DR-1a.ibs</i>	<i>SIEMENS</i>	<i>DR-1a</i>	<i>2.4565e-10</i>	<i>B=2.19%</i>
<i>80960RD.ibs</i>	<i>INTEL</i>	<i>highobuffer</i>	<i>1.80229e-10</i>	<i>0.01 %</i>

Tabla 3.2: Resultados de simulaciones con ficheros IBIS (característica de bajada).

Los errores cometidos son de dos tipos:

- Error tipo A: Se aplica cuando no se alcanza el valor de tensión deseado en el instante en que se mide, es decir, en $dt_{\text{ideal}}/0.6$. Se define como:

$$A = \frac{V_{\text{obtenido}} - V_{\text{ideal}}}{V_{\text{ideal}}} \cdot 100$$

- Error tipo B: Se utiliza cuando se alcanza el valor final demasiado pronto, por lo que es un error de tiempo. Es definido como:

$$B = \frac{t_{\text{obtenido}} - t_{\text{ideal}}}{t_{\text{ideal}}} \cdot 100$$

Al comprobar los resultados con ficheros IBIS reales se pone de manifiesto que no siempre se obtiene una solución correcta. Esto es debido a que la aproximación que se utiliza para la exponencial hace que en algunos casos el resultado para la pendiente sea inestable y carezca de sentido (observar el

orden de magnitud de t_r y t_f), por lo que esto supone una **limitación al modelo**. Aún así, la resolución del problema de esta forma está justificada ya que expresiones algebraicas aceleran en gran medida el cálculo de las pendientes, hecho que resulta interesante al buscarse un modelo rápido para simulación, además de no requerir una herramienta específica para resolver el problema numérico, en caso de querer ser resuelto de esta forma.

También es interesante destacar que en este modelo con sólo un condensador y una resistencia no es posible forzar que las curvas de salida empiecen en un punto (de tiempo y de tensión), pasen por otro valor intermedio, como podría ser el 20% o el 80% en un determinado tiempo, y consigan un valor final de tensión. Las ecuaciones de por sí imponen, como ya ha sido mostrado, los valores iniciales y finales, de forma que únicamente se puede controlar con este método el tiempo en que estos valores se consiguen. Para controlar el 20% y el 80% será necesario un modelo más complejo.

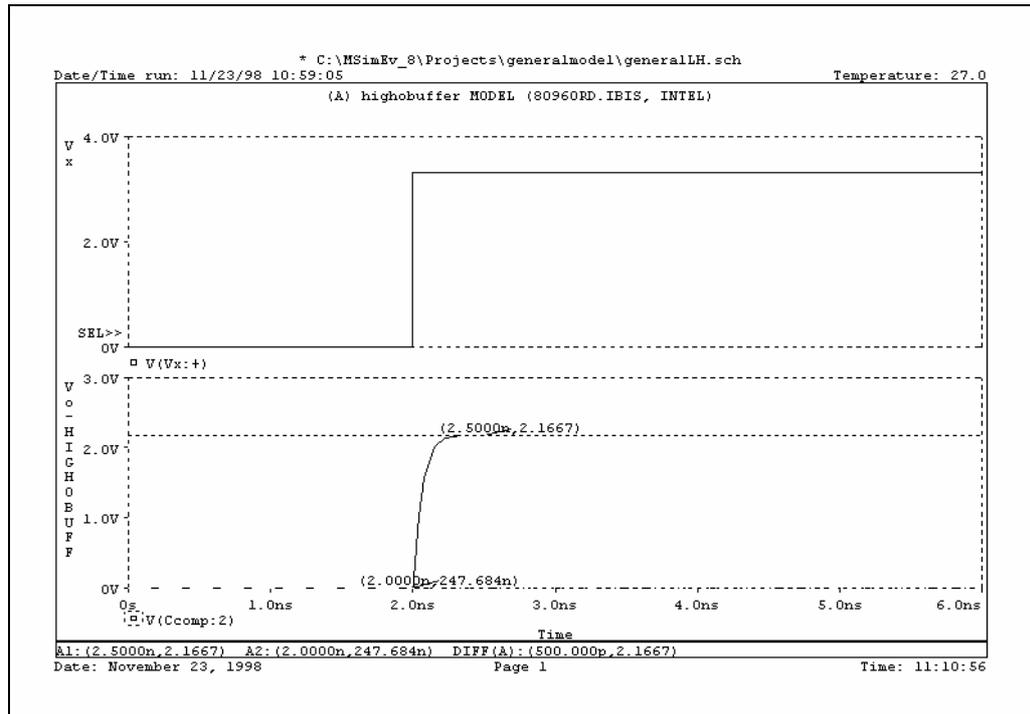


Figura 3.7: Curva de subida para el modelo *highobuffer*, de INTEL.

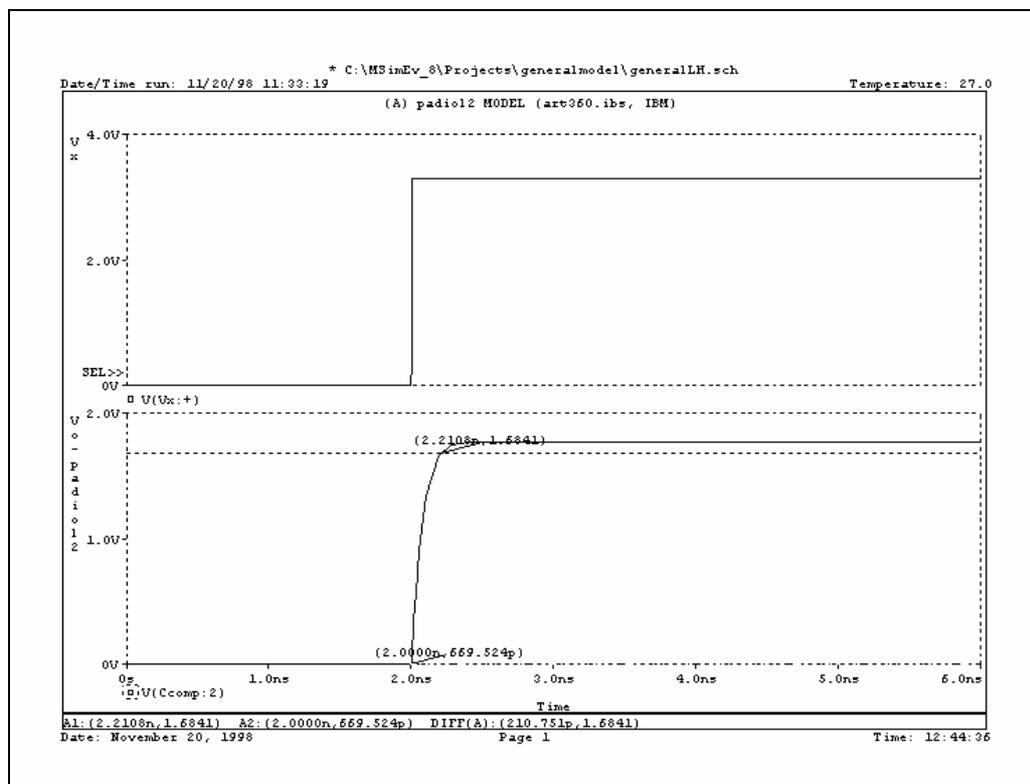


Figura 3.8: Curva de subida para el modelo *padiol2*, de IBM.

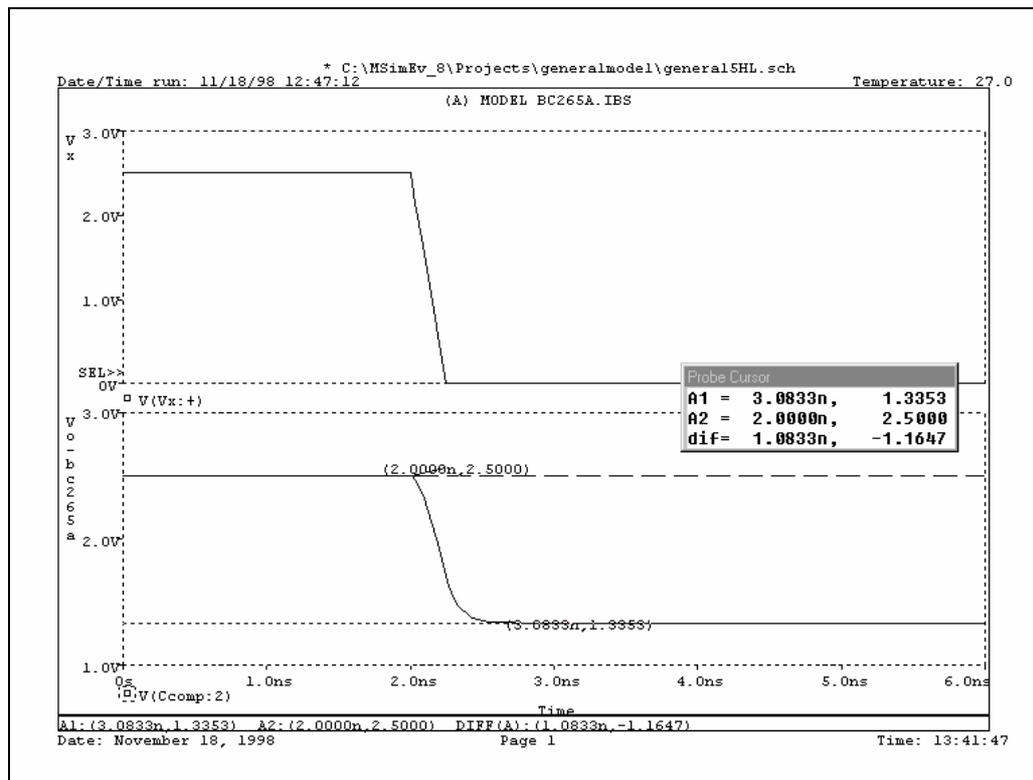


Figura 3.9: Curva de bajada para el modelo *DR-1a*, de SIEMENS.

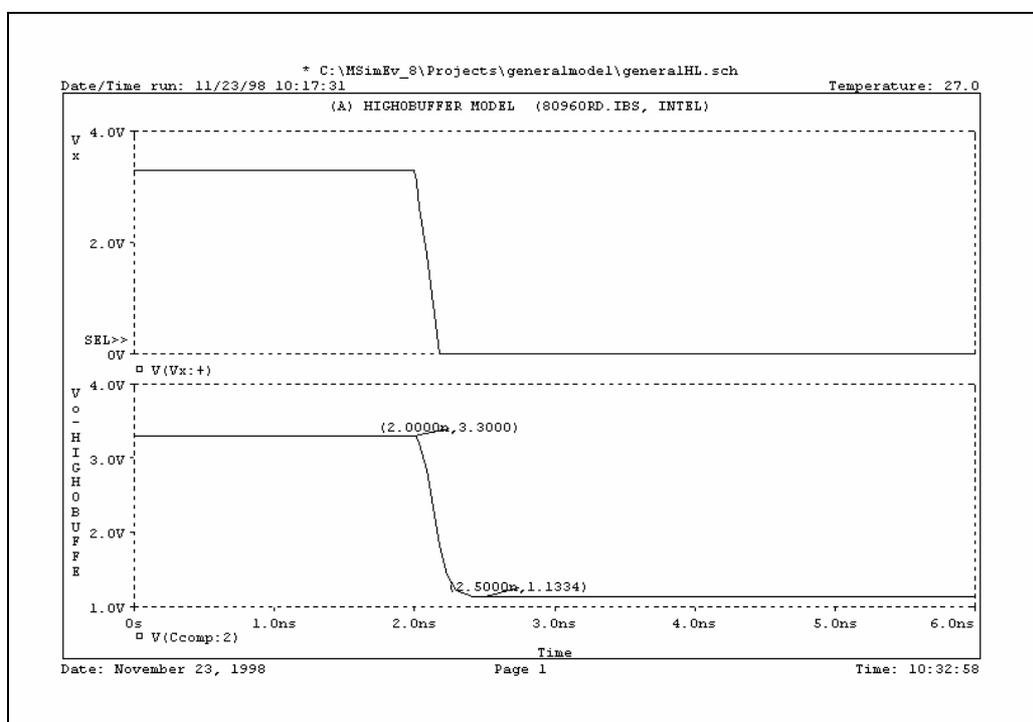


Figura 3.10: Curva de bajada para el modelo *highbuffer*, de INTEL.