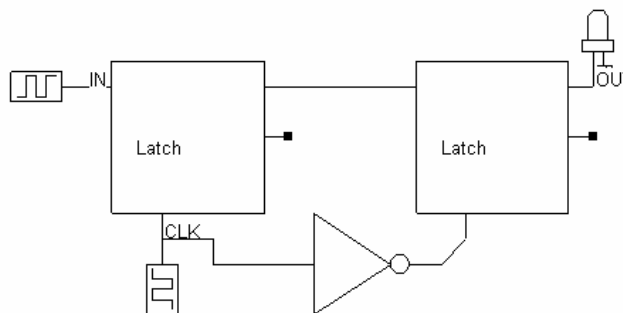


CIRCUITOS SECUENCIALES

EL objetivo de este trabajo es conseguir un contador hexadecimal, es decir, que cuenta de 0 a 15. Para tal fin disponemos de distintas técnicas, que iremos desarrollando, pero primero, explicaremos la forma general de hacerlo partiendo de un biestable latch.

Un biestable latch sigue a la entrada cuando el reloj está a un determinado nivel (es el nivel que da el tipo de latch), mientras que en el otro nivel del reloj, la salida no cambia. Para entender este funcionamiento podemos ejecutar el archivo *“latch.sch”*.

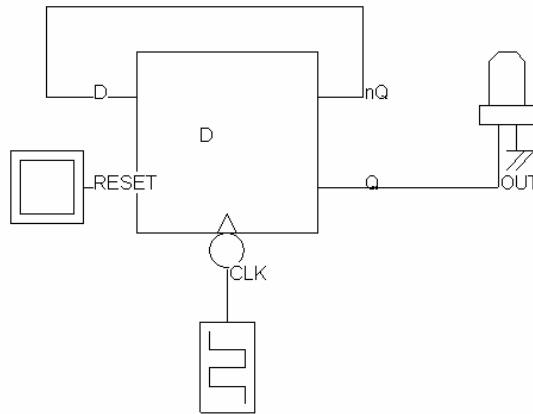
A partir del biestable latch, debemos conseguir un biestable de tipo D. Para ello lo que debemos hacer es unir en cascada dos latch activos por nivel distinto. Por ejemplo si primero colocamos un latch de nivel bajo y a continuación uno activo por nivel alto, obtendremos un biestable D activo por nivel alto.



Ejecutando el archivo *“D.sch”* podemos comparar el propio biestable D del programa con uno implementado por la unión de dos latch tal y como acabamos de comentar.

El funcionamiento del biestable D es similar al latch pero por flanco en vez de por nivel, es decir, que en el momento del flanco, la salida se actualiza con la entrada.

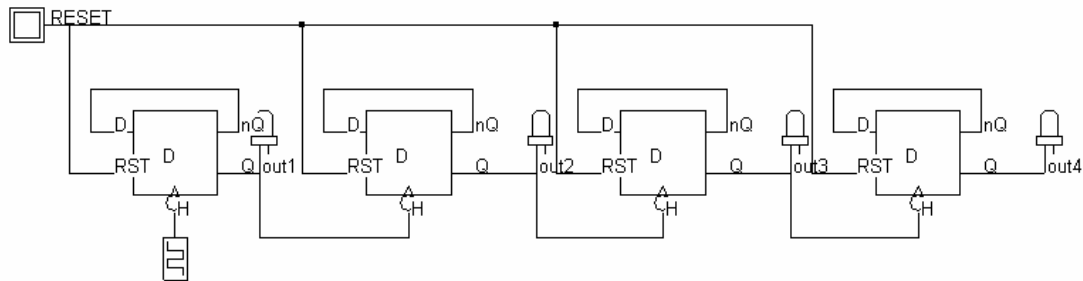
El siguiente paso es conseguir un biestable T, para lo cual debemos partir del biestable D que acabamos de explicar. Para conseguir el biestable T, debemos realimentar la salida negada a la entrada.



El funcionamiento, que se puede comprobar con el archivo “*T.sch*”, es básicamente el de un divisor de tensión, con lo cual divide la frecuencia del reloj entre dos.

Cabe mencionar que en el archivo “*T.sch*”, se ha colocado un pulsador de reset para que se pueda iniciar el funcionamiento normal del biestable T.

Ya por último, habiendo implementado un biestable de tipo T, podemos realizar el contador encadenando cuatro biestables de este tipo. La salida de uno será el reloj del siguiente.



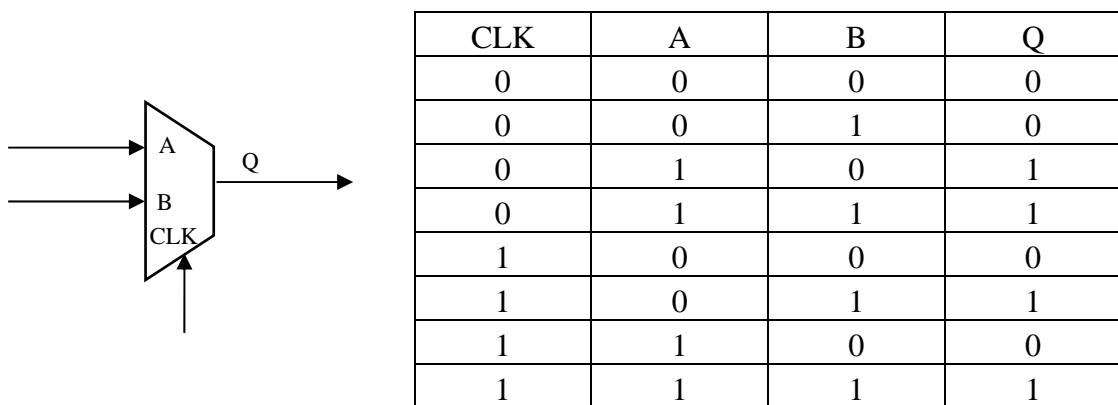
El archivo en el que podemos comprobar el funcionamiento del contador se llama “*contador.sch*”, en el cuál también hay un pulsador de reset, para que se inicie el conteo.

Una vez comentado brevemente el proceso general para conseguir el contador hexadecimal, iremos comentando cada una de las diferentes técnicas que hay para tal fin.

TECNOLOGÍA CMOS

Para esta técnica debemos partir de un paso más atrás aún, con respecto al biestable latch. Este paso es un multiplexor, a partir del cuál podremos ya realizar el biestable latch.

Para realizar un multiplexor con tecnología CMOS, lo primero que hay que realizar, es la tabla de verdad, para ello simplemente se analizan las salidas en función de las entradas.



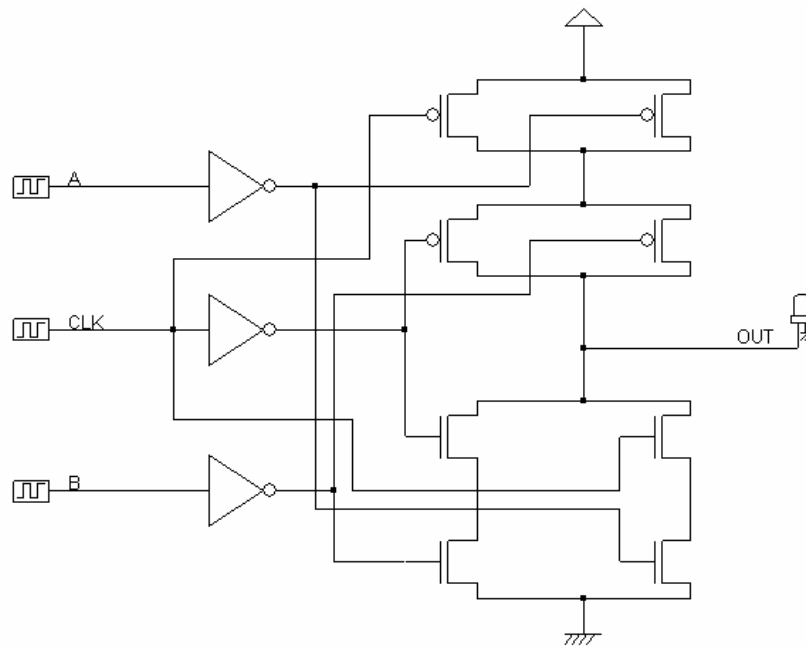
Para reducir la función obtenida nos valdremos del mapa de Karnaugh, colocando en él los ceros para así estimar la red N de la implementación

CLK\AB	00	01	11	10
0	0	0		
1	0			0

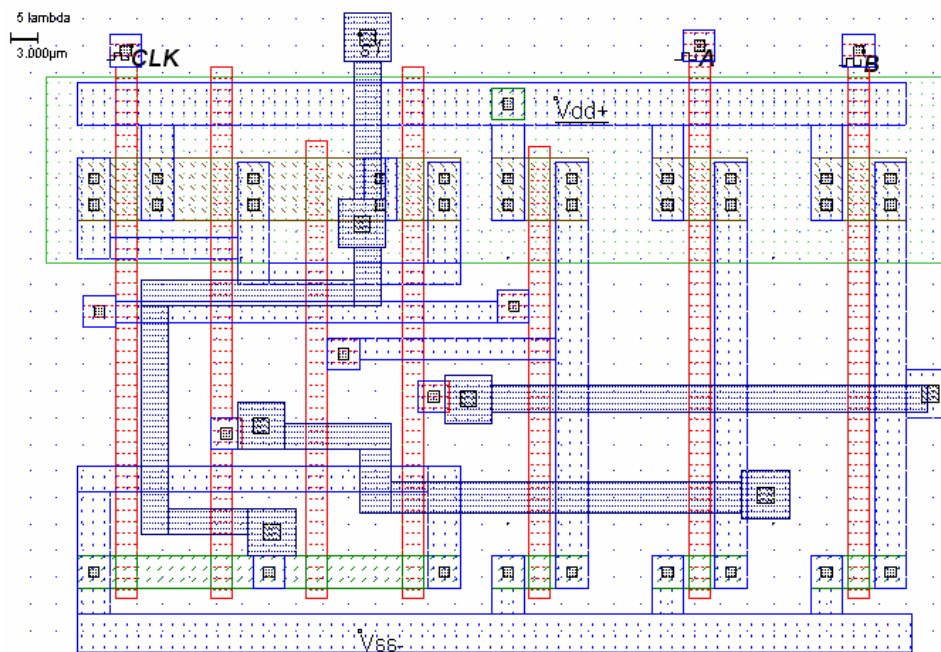
Se obtiene la siguiente función:

$$OUT = CLK \times \bar{A} + \overline{CLK} \times \bar{B}$$

Una vez obtenida la red N, para implementar la red P simplemente habrá que cambiar los paralelos a series y los series a paralelos. De esta manera las puertas utilizadas deben ser las que se muestran a continuación.

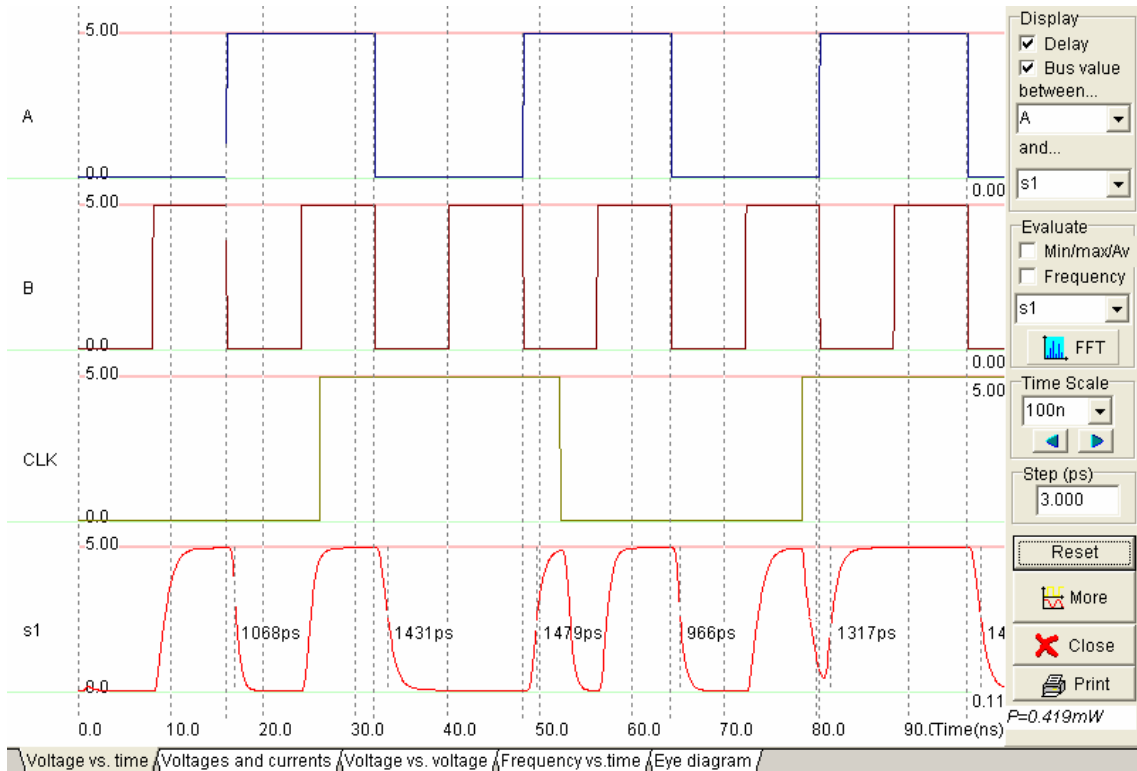


El layout que hemos obtenido es el que se muestra en la siguiente figura, en el cuál hemos mantenido una distancia entre masa y alimentación de aproximadamente 105 μm , tal y como hicimos en la primera parte de la asignatura con las funciones lógicas.

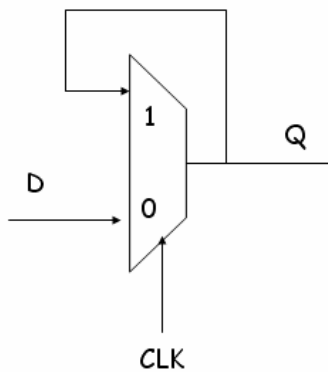


Los tamaños elegidos para los transistores de tipo P son el doble que para los transistores de tipo N. De esta manera obtenemos resistencias en las redes P y N similares.

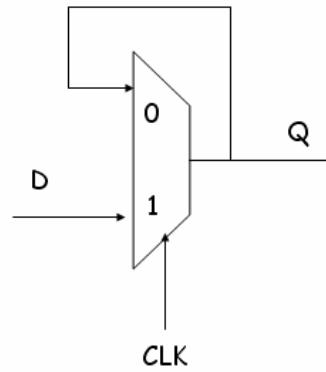
Comprobamos el funcionamiento del multiplexor con la simulación:



Una vez implementado el multiplexor estamos en disposición de realizar biestables latch (de nivel bajo o de nivel alto), para ello realizaremos el siguiente conexionado:

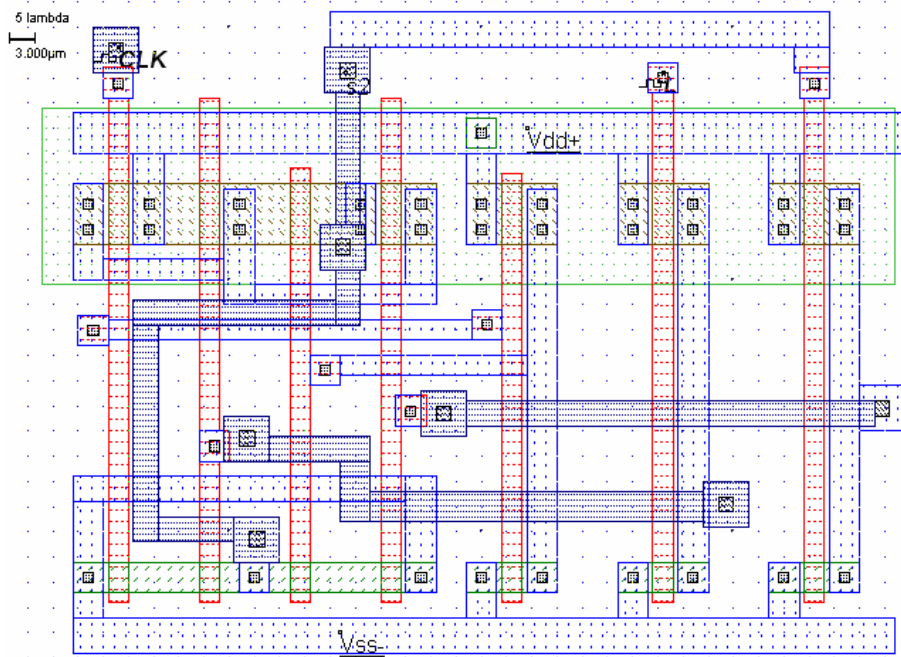


Biestable latch de nivel bajo

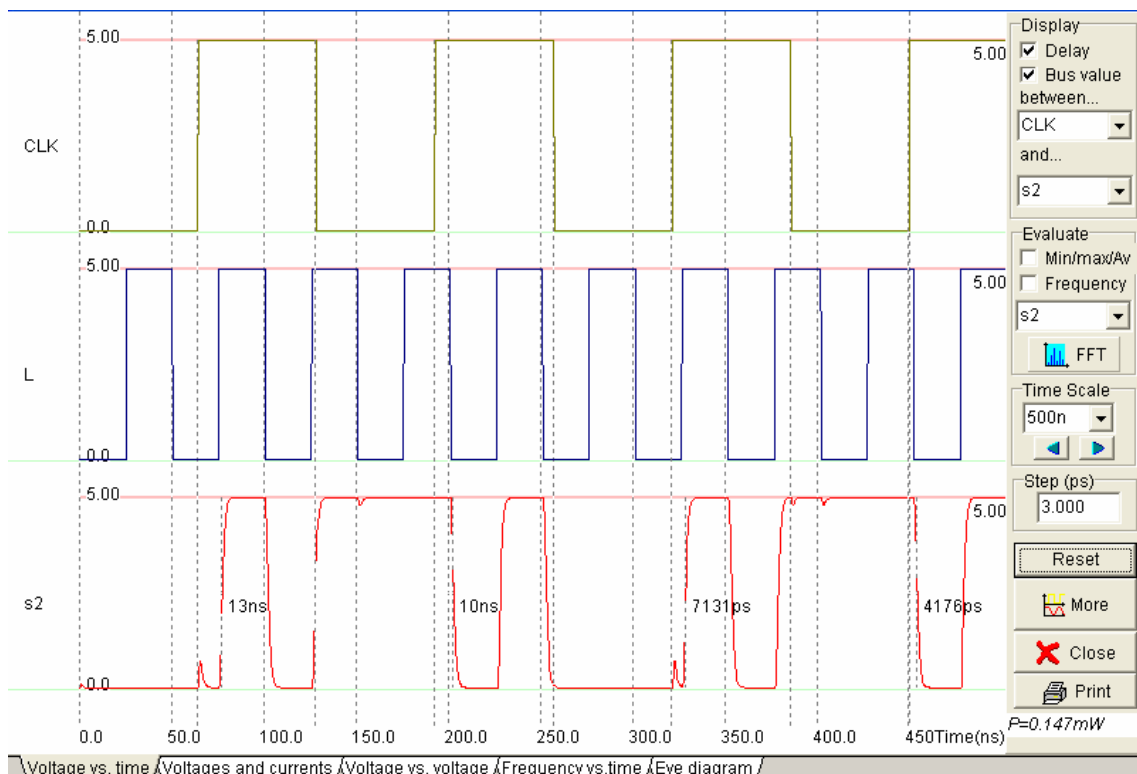


Biestable latch de nivel alto

Realimentada la salida a una de las entradas conseguimos el latch, que en nuestro caso es de nivel alto

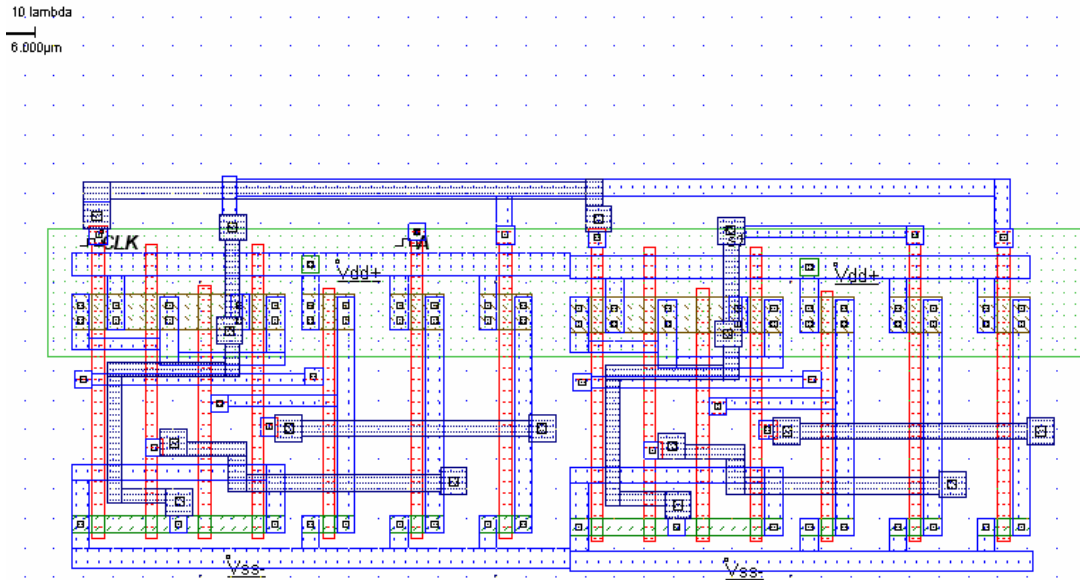


A continuación vemos cómo la salida sigue a la entrada cuando el reloj está a nivel alto, mientras que cuando el reloj está a nivel bajo, la salida no varía.

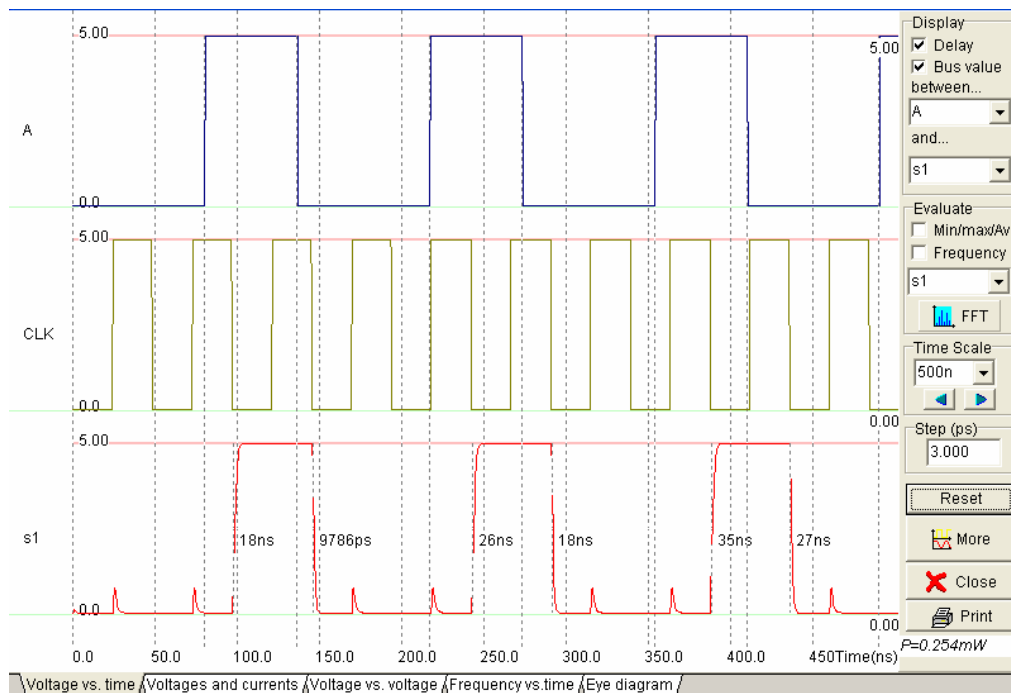


El siguiente paso es realizar el biestable D, para lo cuál simplemente hay que unir en cascada dos latch activos a niveles distintos. En función de cuál pongamos primero, el biestable D será activo por flanco ascendente (latch de nivel bajo + latch de nivel alto) o por flanco descendente (latch de nivel alto + latch de nivel bajo).

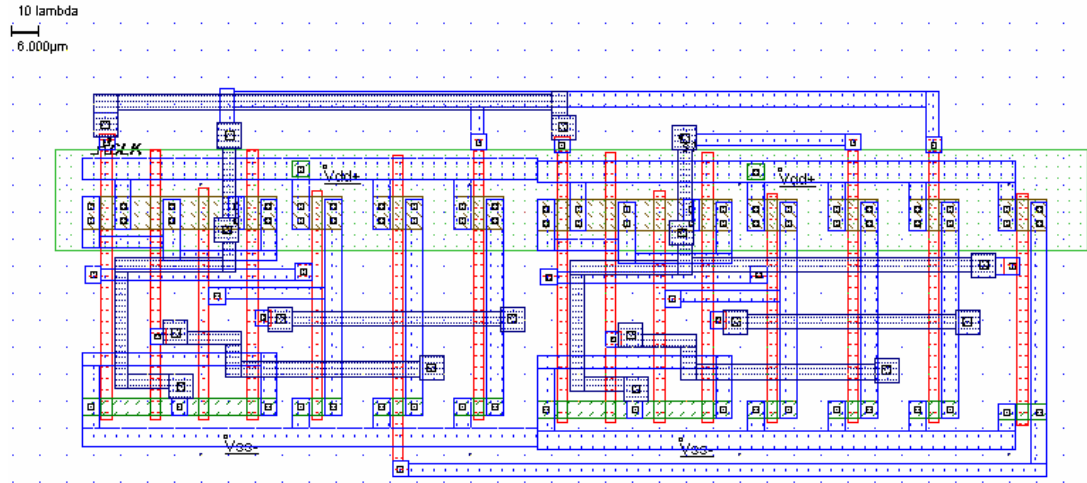
En el layout hemos realizado la conexión de tal manera que el biestable D es de flanco de bajada.



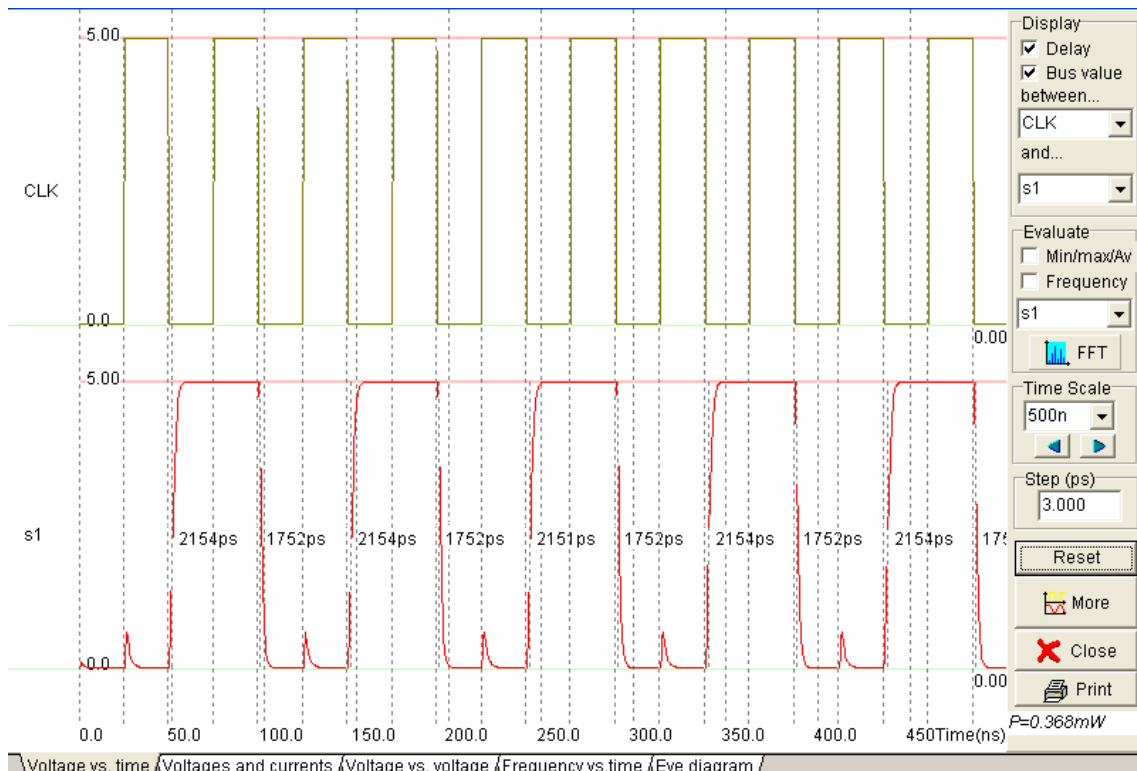
Comprobamos que la gráfica resultante se corresponde con un biestable D de flanco descendente.



El siguiente paso a dar es realizar un biestable T a partir de un biestable D. Para ello hay que unir la salida negada con la entrada. De esta manera y como se puede apreciar en layout, se ha de introducir un inversor más para obtener la salida negada.

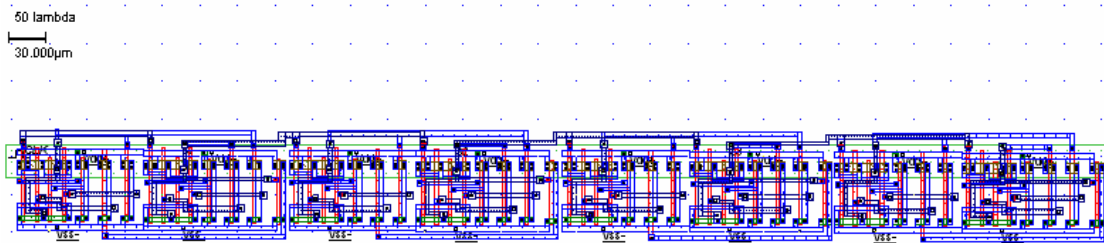


Para comprobar que este layout se comporta como un biestable T, comprobamos las señales obtenidas en la simulación.

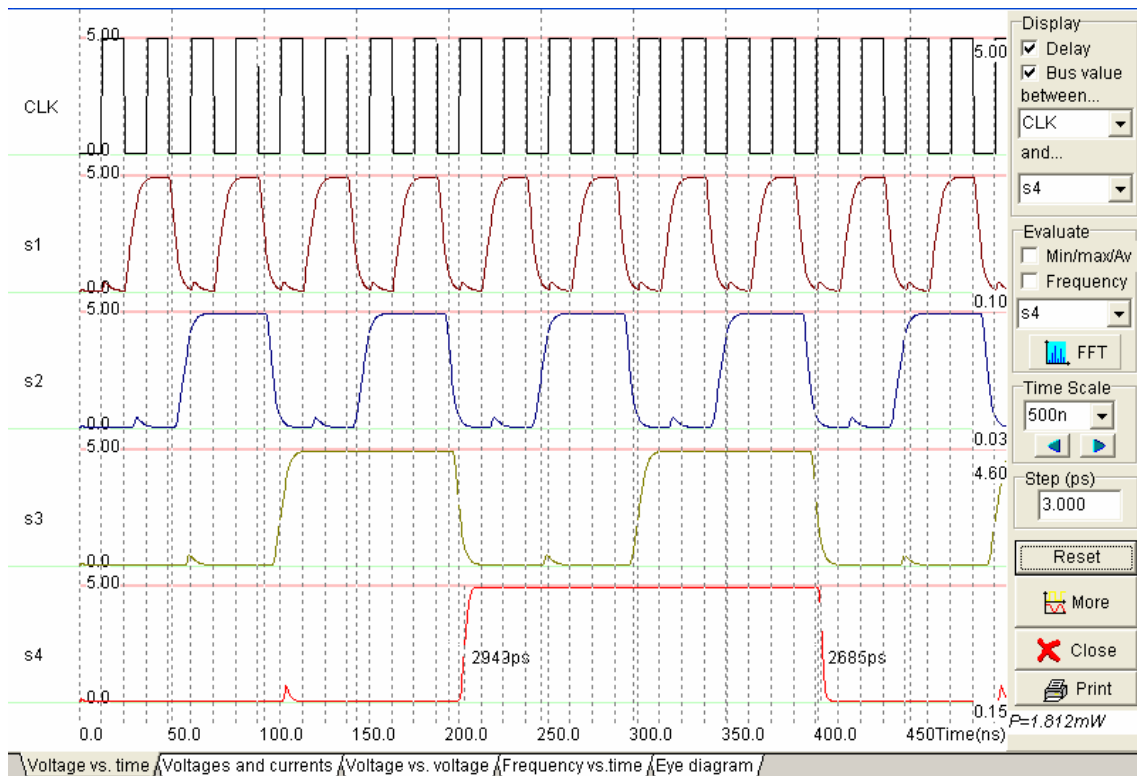


Vemos que la frecuencia del reloj se divide entre dos.

Ahora para conseguir un contador hexadecimal, solamente nos queda colocar en cascada cuatro biestables T, de manera que la salida de uno será el reloj del siguiente. Esto se muestra en el siguiente layout.

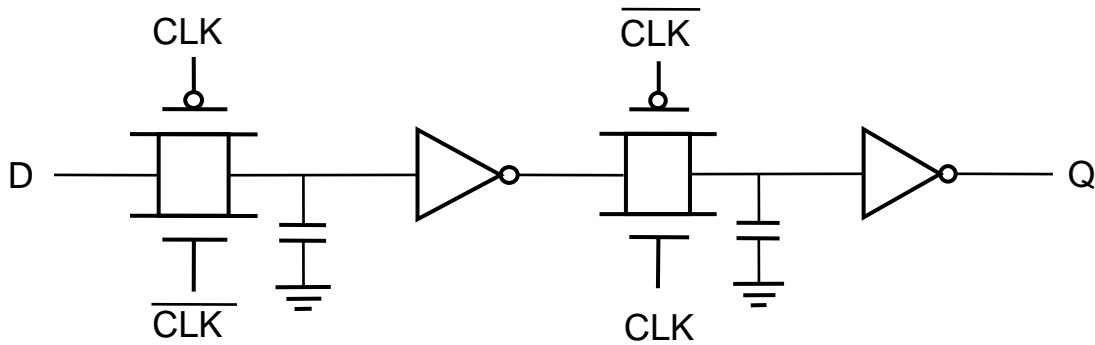


Vemos que hemos obtenido un contador ascendente ya que hemos introducido como reloj la salida del anterior biestable T (sin negar) que es de flanco descendente.



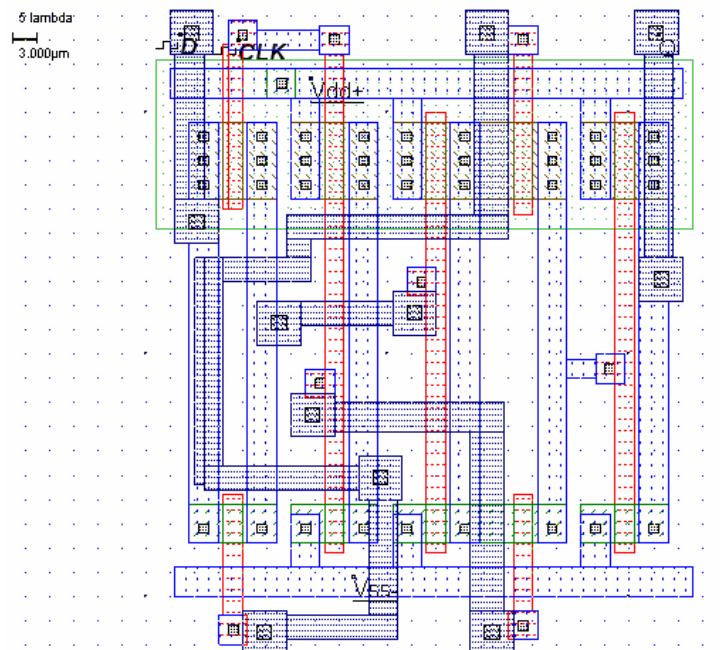
REGISTRO DINÁMICO MAESTRO-ESCLAVO

Con esta técnica para realizar el contador hexadecimal, obviamos la implementación tanto del multiplexor como del biestable latch, ya que partimos de un esquema que funciona como biestable D.

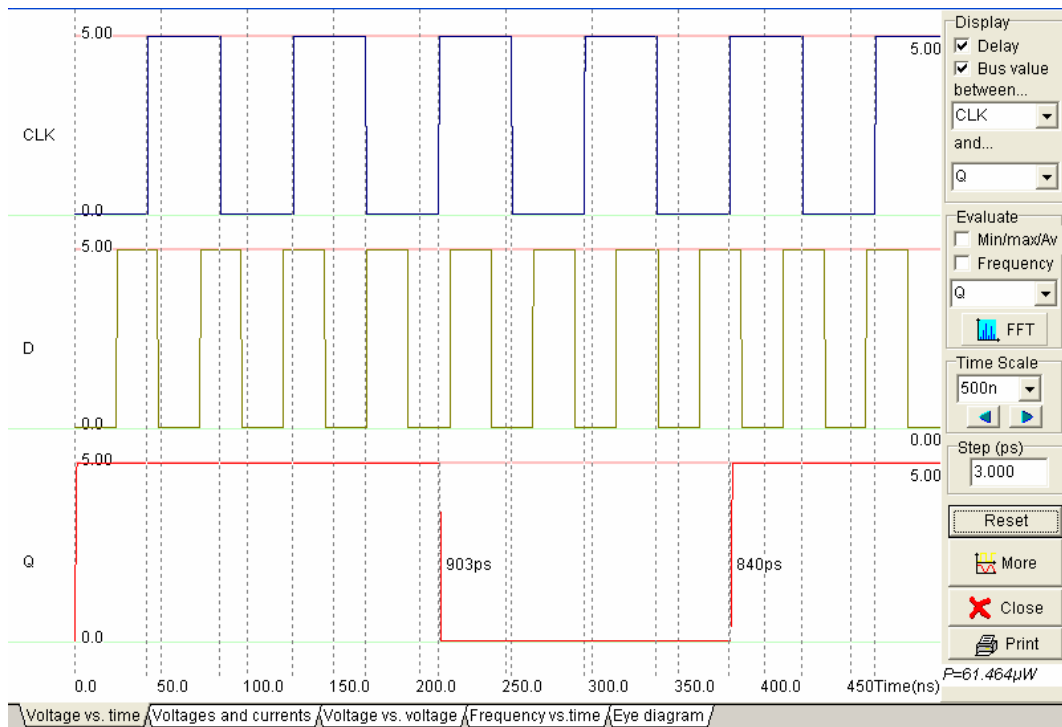


Para realizar este layout, seguimos las normas comunes para los inversores, es decir, que los transistores de tipo P son el doble de grandes que los transistores de tipo N. Para los demás transistores seguiremos unas pautas similares, con la misma relación entre transistores tipo P y N.

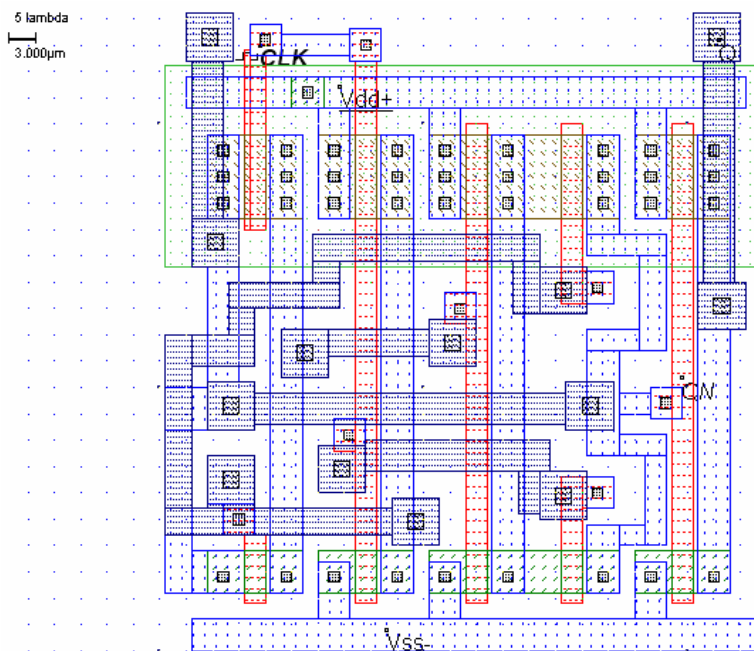
El layout obtenido vemos que es bastante más reducido en cuanto a tamaño (entorno a 105 μm de ancho) que cuando utilizábamos técnicas CMOS a la hora de realizar un biestable D.



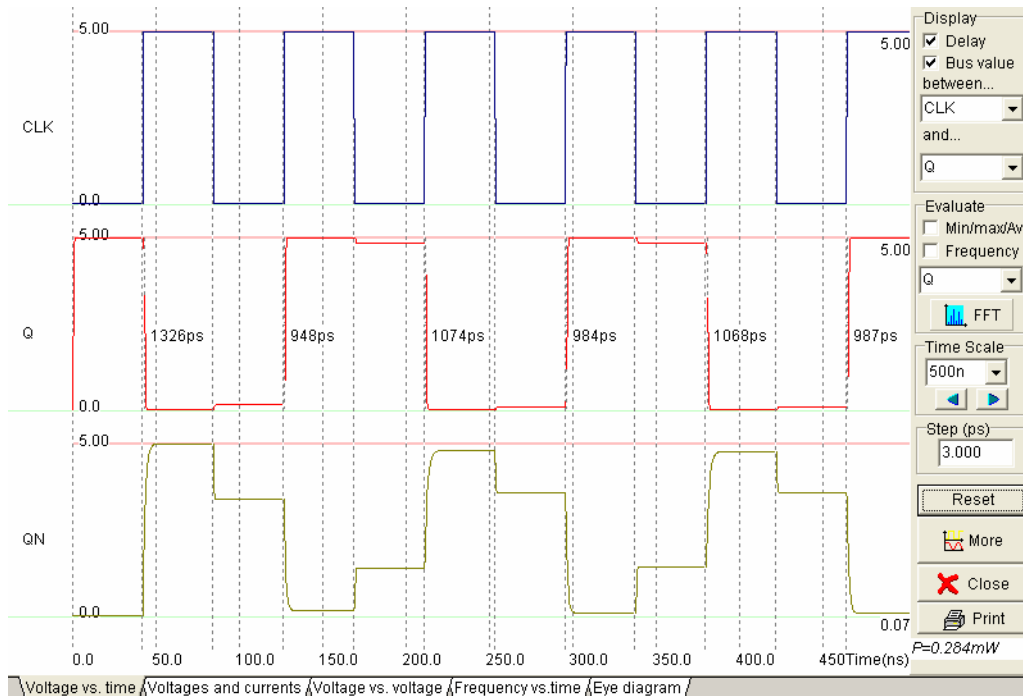
Para comprobar el funcionamiento del circuito ejecutamos la simulación (apreciamos que es un biestable de tipo D, activo por el flanco de subida).



Como es habitual, para implementar el biestable T, simplemente unimos la salida negada del biestable D, con la entrada del mismo. En este caso como en el circuito disponemos de la señal negada (justo antes del inversor final), no necesitamos incrementar el ancho del circuito. El layout es el que se muestra:



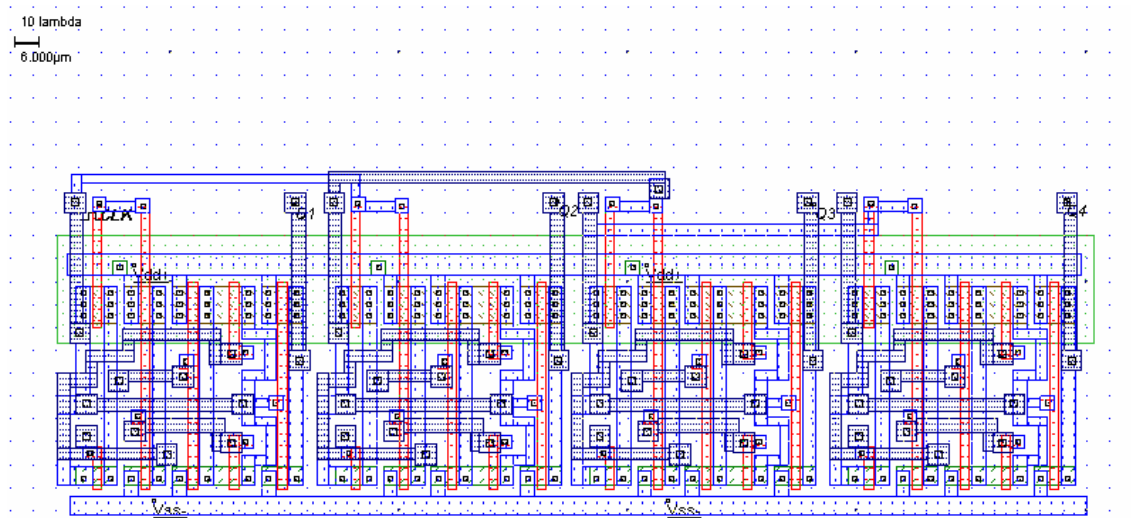
Al ejecutar la simulación, también visualizamos la salida negada (justo antes del último inversor, ya que podremos apreciar un efecto que comentaremos a continuación.



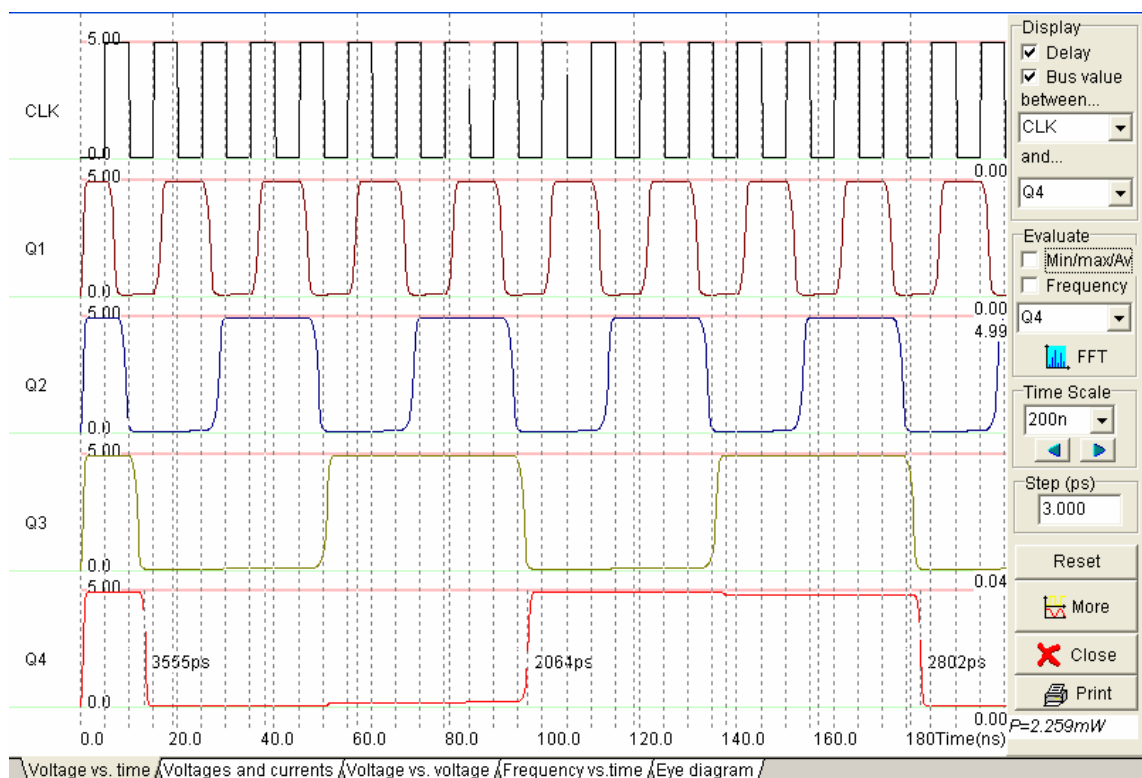
Aunque la salida que nos interesa (Q), más o menos es aceptable, vemos que en la salida negada observada, se producen diferentes niveles. Esto es debido a que hay instantes en que se producen caminos que permiten cargarse o descargarse al condensador que almacena la carga de salida. En el momento en que estos caminos se cortan, el condensador mantiene su carga, de ahí los diversos niveles.

Encadenando cuatro biestables T conseguimos un contador hexadecimal. Si uniésemos la salida de los biestables T (de flanco ascendente) al reloj, obtendríamos un contador descendente. Como lo que pretendemos es obtener un contador ascendente, conectamos la salida negada al reloj del siguiente biestable. Aunque parezca que esto puede inducir a algún error, debido a los diferentes niveles que había en la señal negada de salida de los biestables T, veremos que el contador funciona correctamente.

Veamos el layout:

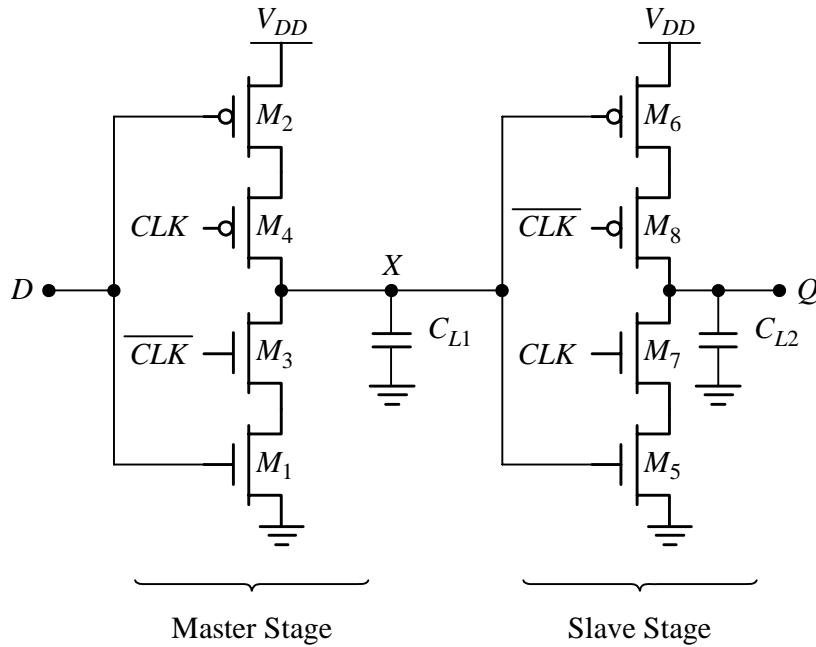


El ancho de este layout es simplemente de unos 450 um.
 Comprobemos que el contador realiza su función:



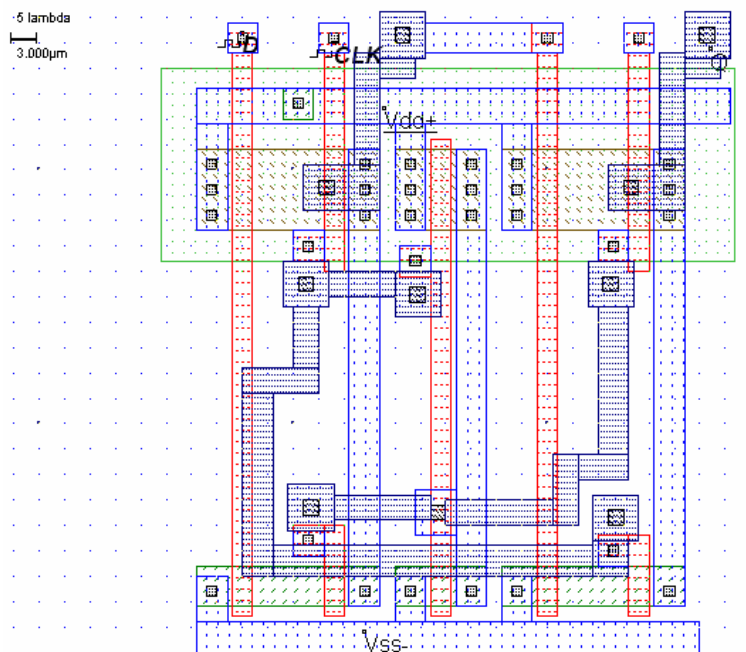
TÉCNICA C2MOS

Como en el caso anterior, partimos de una estructura maestro-esclavo que realiza las funciones de biestable D. El esquema es el siguiente:



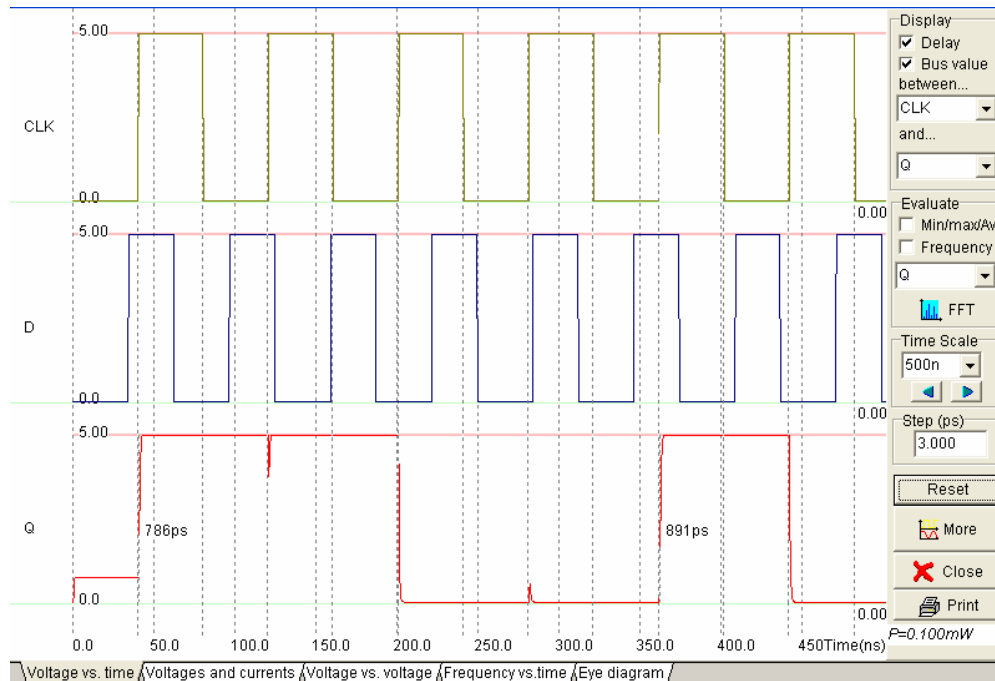
Debido a que tanto la estructura maestra como la esclava van a funcionar como inversores cuando el reloj esté en el instante adecuado, los tamaños de los transistores de tipo P van a ser el doble que los de tipo N.

De esta manera el layout obtenido es el que se muestra a continuación:

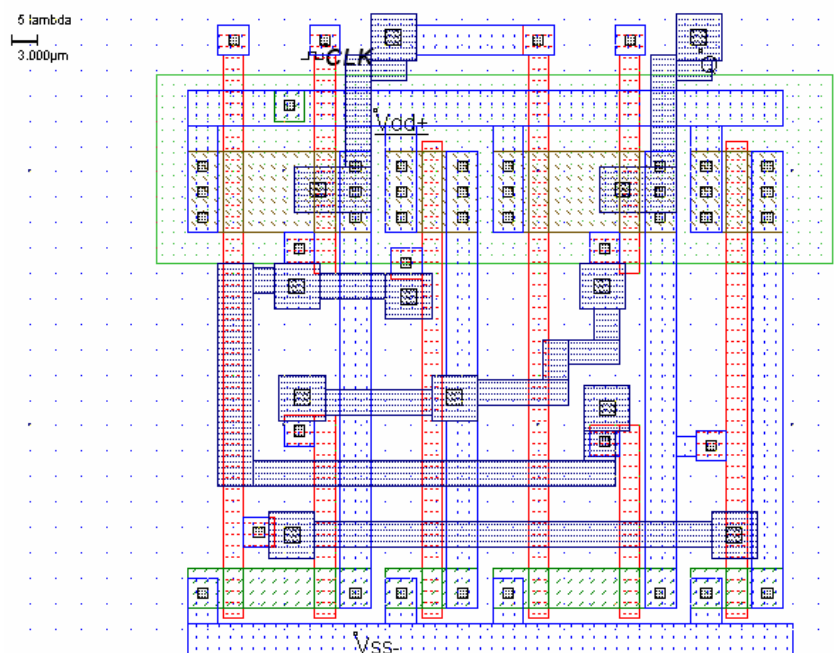


El ancho obtenido es similar al de la técnica del registro dinámico maestro-esclavo, es decir, 105 um.

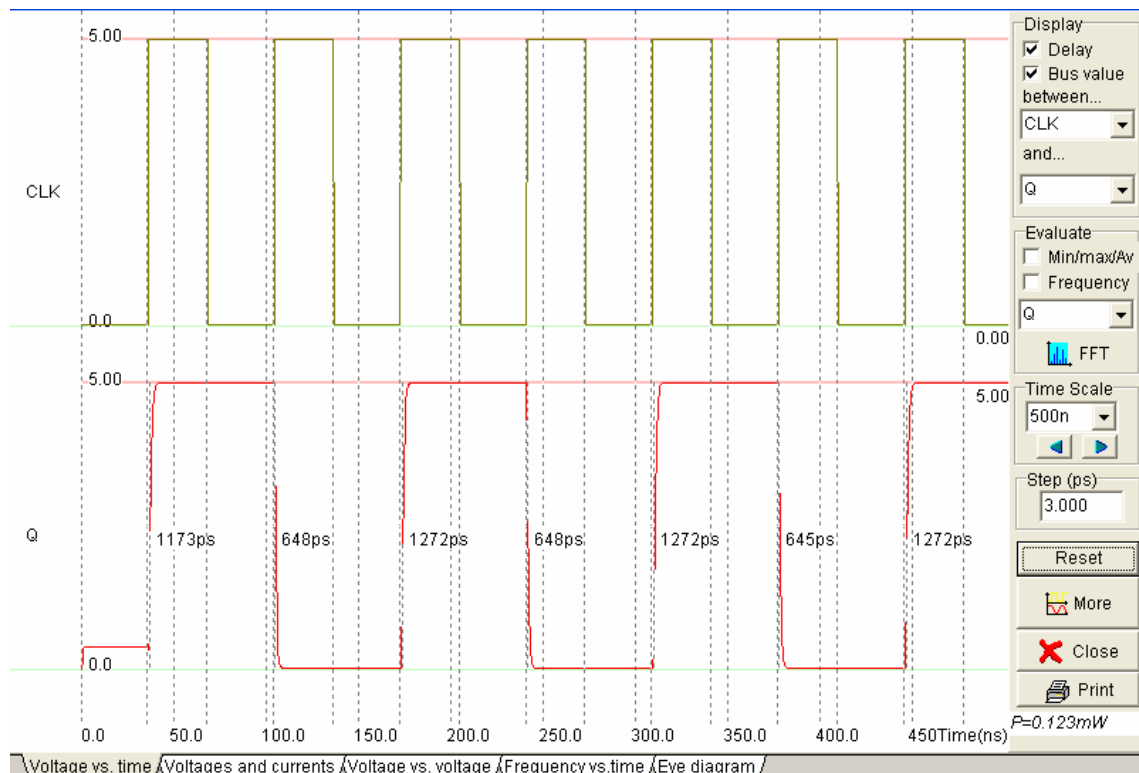
Vemos el funcionamiento del biestable D y deducimos que es activo por flanco ascendente.



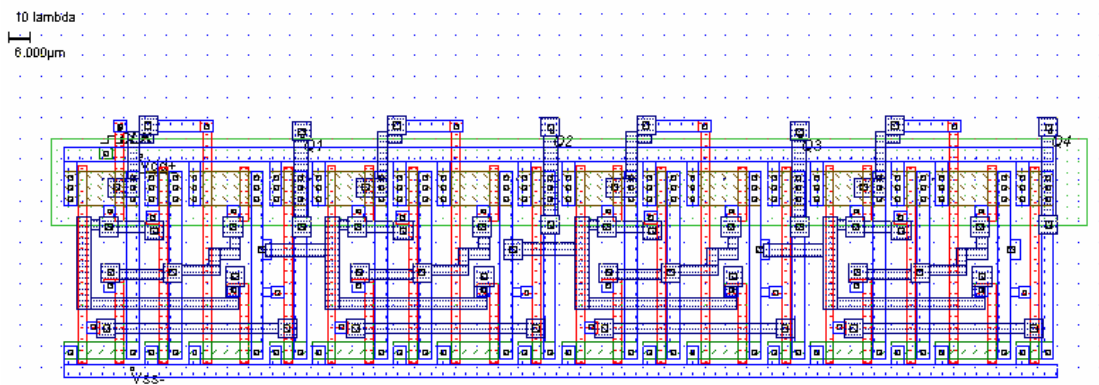
Para realizar el biestable T, añadimos un inversor a la salida para realimentarla a la entrada. No es válido realimentar desde la parte intermedia del circuito.



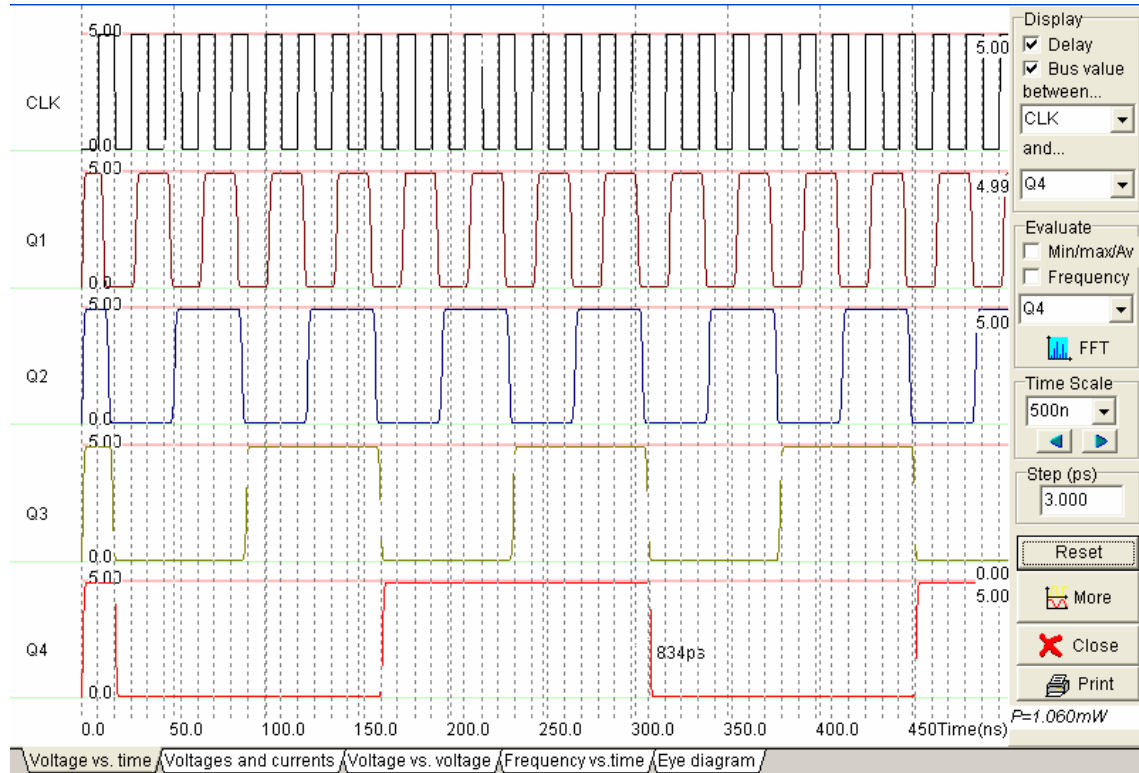
La simulación se representa a continuación.



Para realizar el contador hexadecimal ascendente realizamos el proceso comentado a lo largo del trabajo.

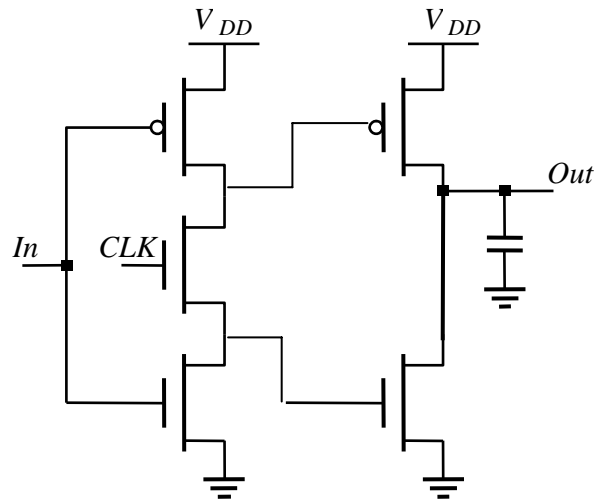


Para conseguir que el contador sea ascendente, visualizamos las salidas negadas, y llevamos al reloj del siguiente biestable, la salida normal.

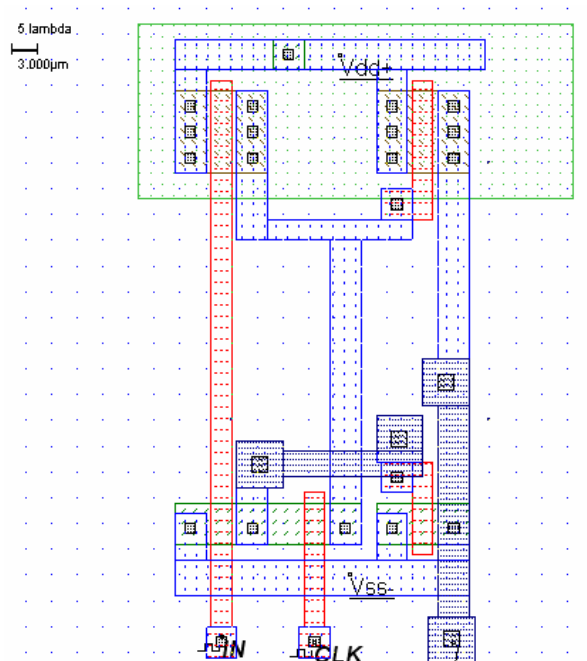


TÉCNICA TSPCR (True Single Phase Clock Register) SIMPLIFICADA

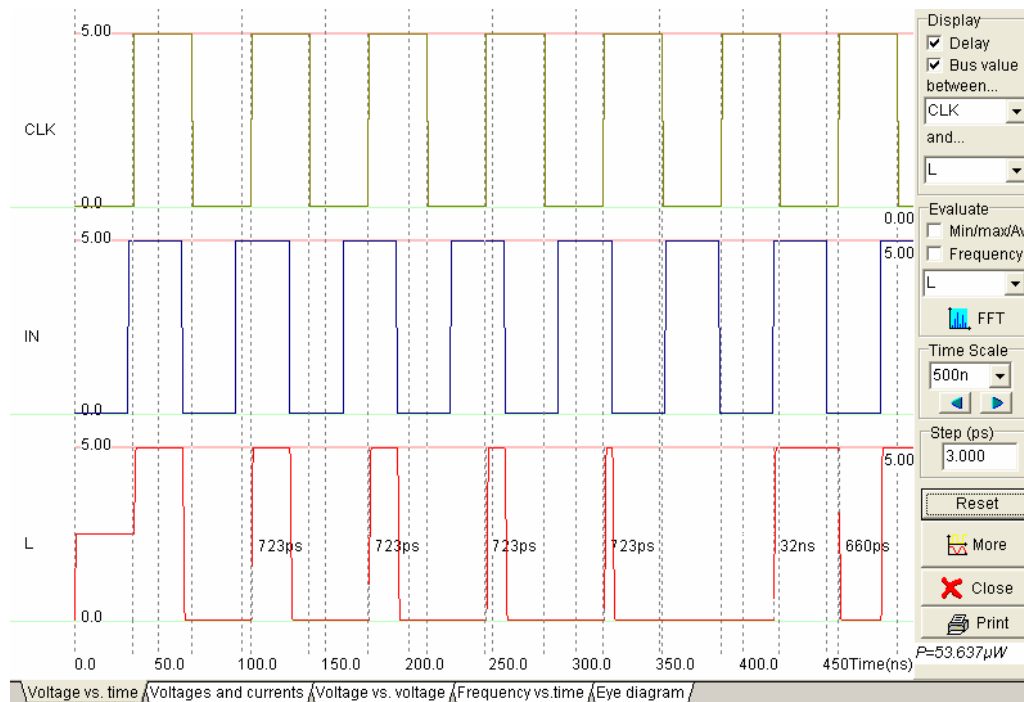
En esta técnica partimos de un biestable latch, ya sea de nivel alto o de nivel bajo (en nuestro caso será de nivel alto). Este esquema, que está simplificado con respecto a la técnica TSPCR, se presenta a continuación.



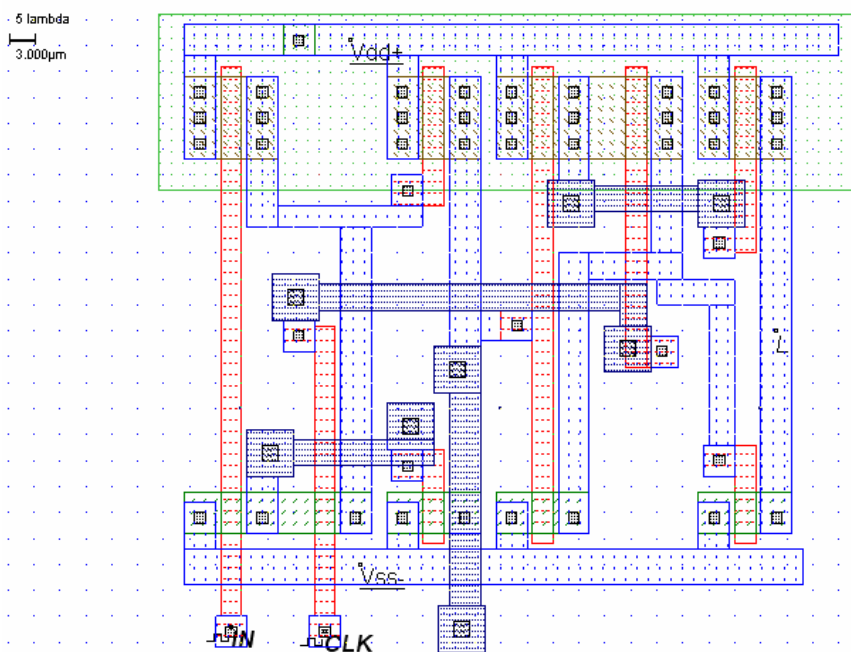
Este esquema se puede implementar en layout simplemente utilizando un ancho de aproximadamente 60 μm . Cabe mencionar que hemos realizado los transistores de la red P el doble de grandes que los de la red N. Esto se representa en el siguiente esquema:

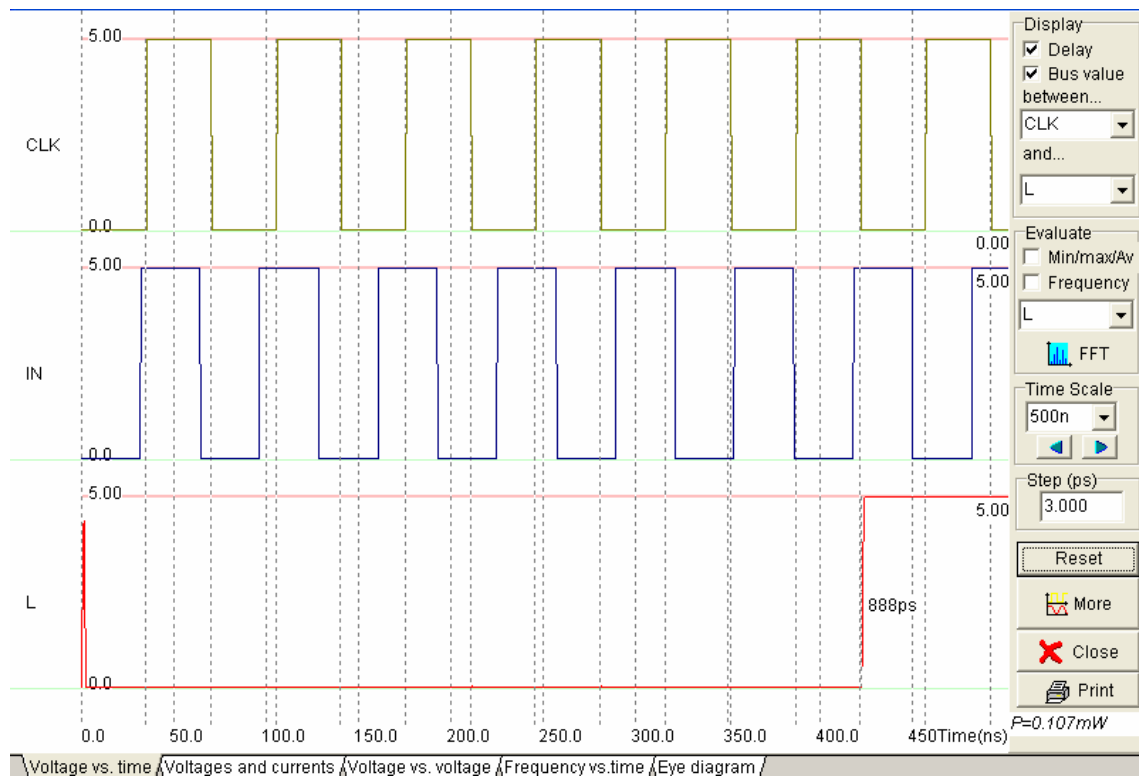


Podemos apreciar que el comportamiento es el correcto, ejecutando la simulación.

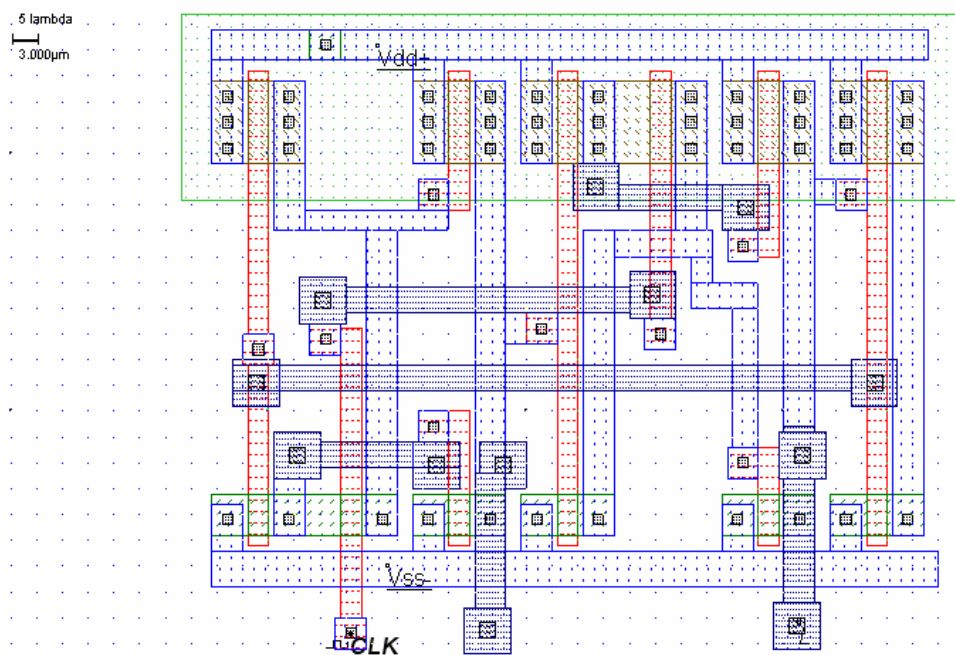


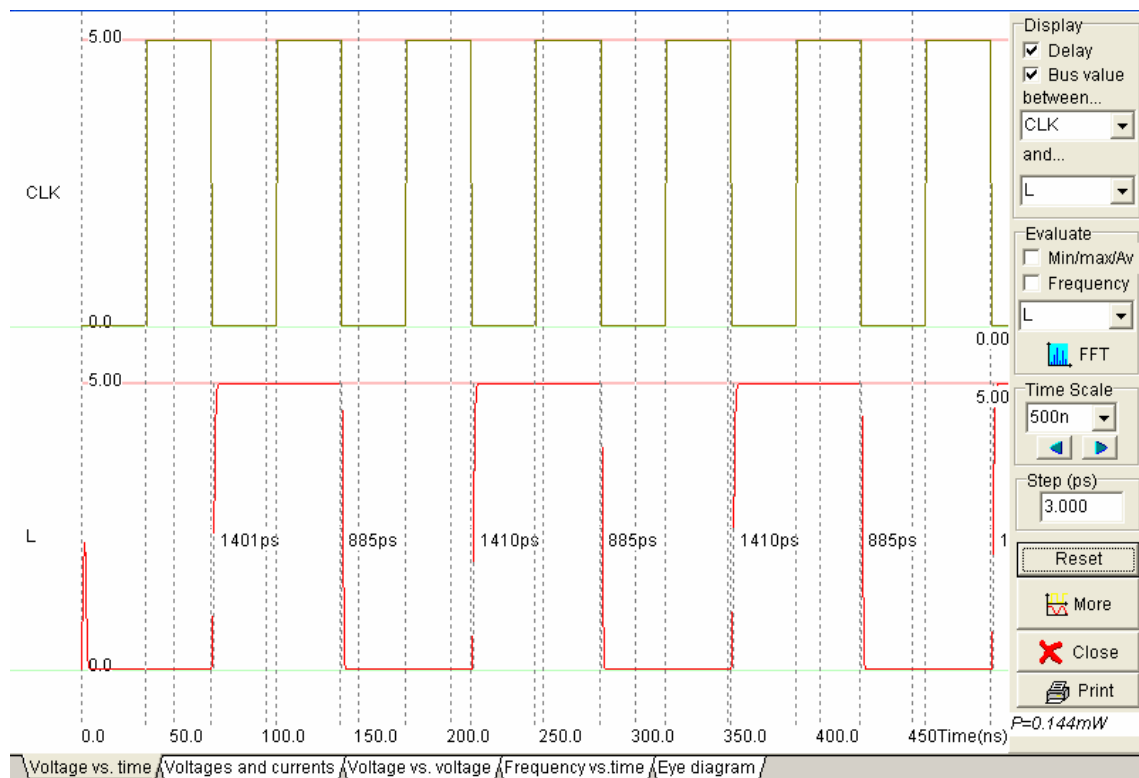
Para realizar el biestable de tipo D, realizaremos la conexión que hemos venido realizando a lo largo de todo el trabajo, es decir, conectar dos latch en cascada. En nuestro caso hemos conectado primero el de nivel bajo y a continuación el de nivel alto, de forma que obtendremos un biestable D activo por flanco descendente.



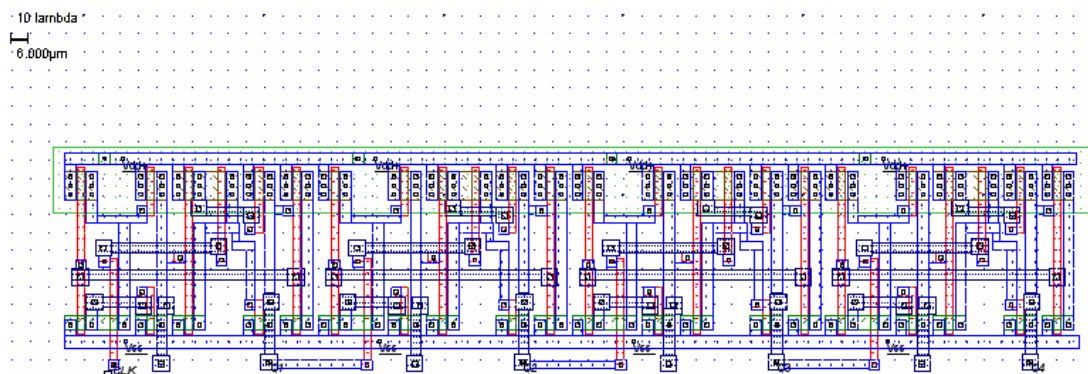


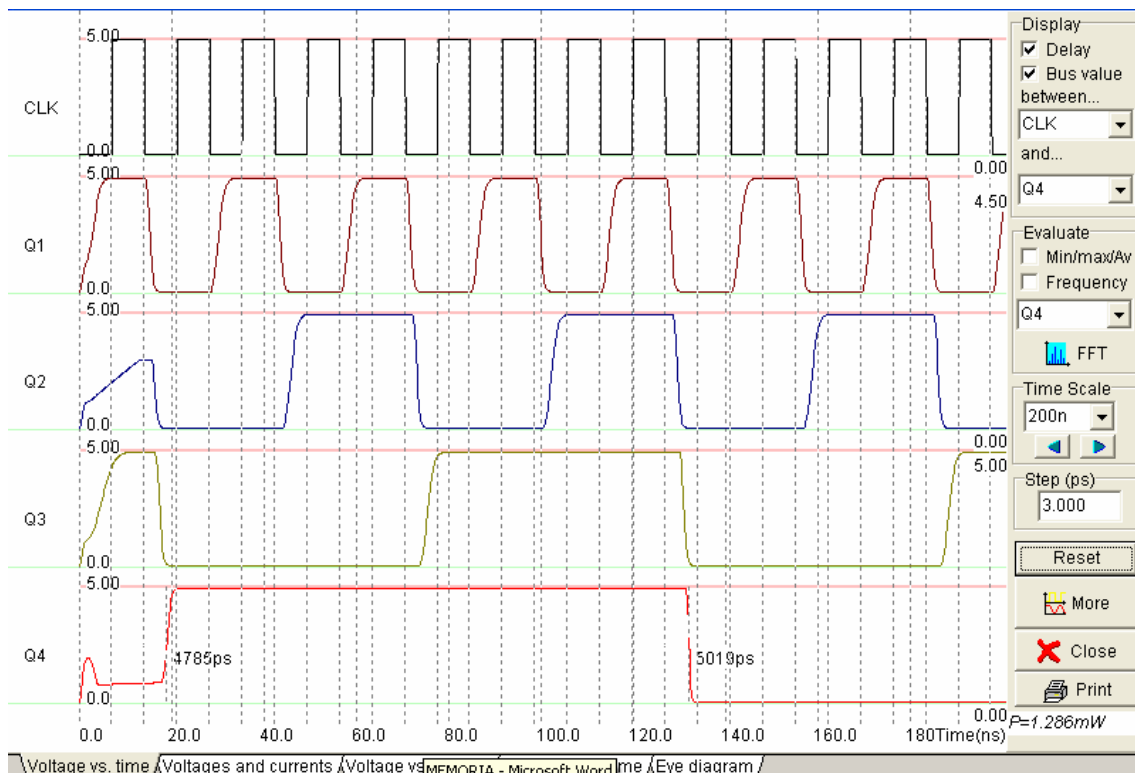
A partir del biestable D, obtenemos el biestable T, realimentando la entrada con la salida negada. Para esta técnica necesitamos incluir un inversor para obtener la salida negada.





Concatenando cuatro biestables T, obtenemos el contador hexadecimal. Para conseguir que sea ascendente simplemente hay que unir la salida de un biestable con el reloj del siguiente.





Apreciamos que el contador es ascendente, pero que comienza en el estado '1000'. Esto es debido a la precarga involuntaria que se produce por los niveles iniciales.

Referencias: "Circuitos Integrados Digitales" 2ª Edición
 Jan M. Rabaey, Anantha ChandraKasen, Borivoje Mikólic
 Pearson Prentice-Hall

"Introduction to nMOS and CMOS VLSI Systems Design"
 Amar MuKherje
 Prentice-Hall International Editions