

# *Generación de cronogramas a medida en Sistemas Digitales*

Javier Roca Piera

Departamento de Informática  
Universidad de Almería

Carretera de Sacramento S/N 04120 Almería, España  
jroca@ual.es

Toni Schaarschmidt

Departamento de Informática  
Universidad de Almería

Carretera de Sacramento S/N 04120 Almería, España  
Toni.Schaarschmidt@hotmail.de

**Resumen**—Los diagramas de tiempos son una herramienta habitual utilizada en el aprendizaje de sistemas digitales. La relación temporal existente entre varias señales, expresada de forma gráfica, nos facilita la comprensión de la forma de actuar de aquellos dispositivos que son constitutivos del cualquier sistema digital y por extensión de cualquier sistema computacional. Ejemplificar el comportamiento de un circuito digital necesita cronogramas funcionales adecuados que pongan de manifiesto todas las posibilidades que pueden darse y que necesitan ser mostradas para una correcta comprensión del funcionamiento de los distintos dispositivos. Se presenta en este trabajo una herramienta que posibilita la generación de cronogramas de forma sencilla con fines didácticos para el aprendizaje del comportamiento de los circuitos digitales.

**Keywords**—tecnología de computadores; simulación digital; cronogramas; docencia en grado

## I. INTRODUCCIÓN

El aprendizaje de conceptos asociados a asignaturas del campo de los sistemas digitales, donde el análisis del comportamiento de dispositivos incluidos en esta área, requiere constantemente la utilización de señales lógicas, las cuales deben modificar su valor de forma cronológica sin la obligación de mantener un patrón determinado. La generación de estas señales de esta forma nos asegura plantear un conjunto flexible de supuestos que abarquen las distintas posibilidades que se pueden presentar a la entrada de un circuito digital.

La enseñanza de tales materias obliga a la generación continua de cronogramas que muestren la casuística a la que tendrán que responder los distintos circuitos digitales. Siendo esta generación simple desde el punto de vista conceptual, se ocasionan no pocos inconvenientes cuando se quiere implementar su realización de forma digital, dado que no existen aplicaciones que faciliten tal tarea.

Por otra parte, la elaboración de relaciones de ejercicios en los cuales se propongan múltiples casos que pueden darse a la entrada de un circuito digital, se hace una tarea tediosa únicamente por el inconveniente de no poder generar de forma fácil diagramas de tiempos que sean de nuestro interés y que sabemos pueden ser de utilidad para presentarlos en la resolución de ejercicios por parte de los estudiantes.

Los simuladores del comportamiento de circuitos digitales, utilizados de forma general para el aprendizaje de estas materias como PSpice [1] o Logic Works [2] son indiscutibles para un correcto aprendizaje. Su utilidad se pone de manifiesto por la facilidad con la que podemos incorporar dispositivos a nuestro panel gráfico. Así mismo, estos programas nos permiten, además de otras múltiples posibilidades, establecer las líneas de conexión necesarias y analizar el comportamiento del circuito de forma precisa. Estos programas nos muestran las señales que se obtienen en la salida del circuito, tras incorporar a su entrada un determinado panel de señales representativas de los valores digitales que toman las distintas variables que forman parte de la función o funciones. Así mismo nos ofrecen valoraciones temporales de la respuesta del circuito desarrollado.

Estos programas nos permiten abordar proyectos académicos con distintos niveles de complejidad, desde los simples con un pequeño número de dispositivos, a otros que se acercan significativamente a los proyectos profesionales. Los valores de estas variables se ejemplifican en los simuladores por medio de interruptores binarios, si bien se suele disponer para los casos que requieren sincronismo, de una señal de reloj en la que se puede determinar su periodo y ciclo eficaz, sin embargo no tienen como objetivo prioritario facilitar la disposición de cronogramas flexibles y a medida en las entradas de los distintos dispositivos seleccionados, ni ofrecerlos como salida de una forma versátil para su presentación gráfica.

Niveles iniciales de aprendizaje en este campo requieren una comprensión detallada del comportamiento digital de los elementos fundamentales como puertas y biestables de una forma particularizada elemento por elemento. Esto nos permitirá posteriormente analizar agrupaciones cada vez más complejas de los mismos. Asignaturas como Estructura y Tecnología de Computadores que se imparte en el primer curso de Ingeniería Informática, en donde se adquieren desde los conocimientos básicos de sistemas digitales hasta los conceptos de organización de computadores y que son base para construir un conocimiento avanzado en Arquitectura de Computadores, necesitan estrategias de aprendizaje que faciliten y afiancen el conocimiento que deben tener los estudiantes de cara a poder presentarles escenarios más complejos.

La responsabilidad con que el alumno debe abordar el aprendizaje de estas asignaturas está basada fundamentalmente en su motivación y ésta, como pieza fundamental, en la adecuada estrategia que el profesor debe llevar en todo el proceso. Es necesario planificar distintos tipos de actividades que fomenten su participación activa con objeto de mantener el interés del alumno en el aprendizaje de determinados conceptos, que no siempre considera de utilidad para una formación práctica en un determinado campo profesional.

El esfuerzo que vamos a demandar del estudiante solo es posible obtenerlo si éste confía en que el camino que hemos marcado en la asignatura es no solo rentable, sino realmente formativo y es ahí donde la generación de herramientas adecuadas en conjunción con una buena labor de tutoría adquiere un papel relevante [3] [4]. Otros proyectos metodológicos fomentan la utilización de tecnologías interactivas en el aula, con el ánimo de facilitar al profesor el conocimiento adquirido por los estudiantes en la propia sesión de clase presencial [5].

Es en estos niveles iniciales de aprendizaje donde las exposiciones realizadas en clase y la presentación de ejercicios, planteados como actividad para realizar de forma autónoma o grupal por los estudiantes, echan en falta una forma cómoda de generar señales digitales a medida, las cuales puedan servir como variables booleanas que evolucionan en el tiempo o como señales de reloj para aquellos elementos digitales que requieren sincronismo. Tenemos presente que en los inicios de un primer curso se debe prestar más atención a la comprensión de conceptos que al manejo de herramientas. Desde este punto de vista la utilización de aplicaciones que visualicen de forma gráfica lo diseñado y sean cómodas de usar facilitan en mayor medida la consecución de este objetivo [6] [7]. Su utilización favorece mucho más el aprendizaje de conceptos, que si tuviera el alumno que focalizar su atención en manejar herramientas más complejas, las cuales se incorporarán una vez avanzado el curso; todo ello sin perder de vista el grado en que debe ser guiado [8], cuestión ésta clave así mismo, para una correcta evolución del aprendizaje.

La experiencia nos enseña que una tarea simple, que de forma manual en pizarra o papel no plantea ninguna dificultad, como la de generar cronogramas, se torna tediosa cuando se quiere realizar sobre un documento o aplicación digital, constatando algo no deseable como docentes, que las tareas fáciles a veces nos llevan mucho tiempo. Por otra parte, el disponer de una señal binaria cronológicamente variable en formato digital, nos ofrece la posibilidad de utilizarla como señal adaptada para ser manejada por otras etapas en las que se pueda abordar la comprensión de los elementos fundamentales mencionados anteriormente.

Ejemplificar el comportamiento de un circuito digital necesita cronogramas funcionales adecuados que pongan de manifiesto todas las posibilidades que pueden darse y que necesitan ser mostradas para una correcta comprensión del funcionamiento de los distintos dispositivos.

Es una necesidad continua generar cronogramas para proponer casos y supuestos que sirvan para analizar el comportamiento de sistemas digitales.

La ausencia de una herramienta específica que sirva de apoyo tanto a profesores de éste área que en su clase quieran mostrar de forma clara el funcionamiento de determinados circuitos digitales, como a alumnos que en su tareas autónoma de aprendizaje tengan a disposición una forma de comprobar lo explicado en clase, es lo que nos motivó a construir esta aplicación.

## II. ENTORNO DE DISEÑO

Uno de los factores fundamentales que se plantearon inicialmente cuando se constató la necesidad de una herramienta que solventara las dificultades que se tenían, fue asegurar un marco de diseño claramente sencillo de utilizar. Además se tenía conciencia de que lo que buscábamos solventar no podría tener un largo recorrido de ampliación, dado que el incremento de potencialidad está claramente limitado por las aplicaciones utilizadas masivamente como simuladores en el campo de la enseñanza de los sistemas digitales, mencionadas anteriormente. Nuestro objetivo no requería grandes inversiones, sino conceptos claros que nos ofrecieran como salida cronogramas en los que la flexibilidad de su definición fuera el factor clave.

La aplicación desarrollada corre sobre un entorno Excel en el que se han parametrizado el número máximo de señales, así como el número de pulsos máximo a representar, los cuales se definen por filas y columnas respectivamente. Se consideró que ofrecer la posibilidad de representar 16 señales como máximo era más que suficiente para nuestros intereses. De igual forma la representación de hasta 40 pulsos nos posibilita analizar cualquier situación que queramos plantear en nuestro esfuerzo docente. La asignación de valores a las variables de entrada se realiza de forma natural colocando un 0 o un 1 en el periodo de tiempo que queremos que la señal esté a un valor bajo o alto, respectivamente.

De igual forma el periodo de tiempo asignado a una columna puede ser subdividido con el ánimo de flexibilizar la forma de onda que queremos construir. Esto no da una gran versatilidad a la hora de configurar la señal de entrada. Cada espacio temporal puede ser subdividido en 6 subespacios, los cuales pueden tomar los valores que asignemos 0 o 1, de forma similar a como lo hacemos con todo el espacio correspondiente a la representación del periodo de tiempo asignado a una columna.

En la figura 1 se muestra la generación de dos señales digitales. Una de ellas A, se puede interpretar como la forma clásica de una señal de reloj, como explicaremos más adelante. La otra B, ejemplifica el rango de posibilidades que nos ofrece la aplicación, toda vez que podemos establecer un conjunto amplio de relaciones casuísticas entre ella y la posible señal de reloj.

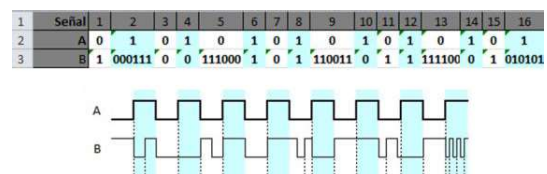


Fig. 1. Asignación de valores a dos variables lógicas

Con el límite de señales y pulsos indicado anteriormente, 16 señales y 40 pulsos, se dispone de un entorno sobre el cual podemos definir un conjunto de señales a nuestra voluntad.

El entorno de diseño se muestra en la figura 2, en la cual podemos visualizar los distintos elementos que nos permiten configurar nuestro cronograma. Por una parte el espacio de filas y columnas Excel donde asignaremos nombre a las distintas señales, así como la evolución de sus valores digitales y por otra un conjunto de “botones” que nos permitirán visualizar la propuesta o propuestas planteadas numéricamente.

Una vez asignados los valores digitales en cada espacio temporal, podemos proceder a representar gráficamente estas señales simplemente pulsando el icono correspondiente “Recalcular y Dibujar”. Esto provoca la representación gráfica de los conjuntos de valores asignados previamente, en el orden en que se han definido, fila tras fila.

Tal como se muestra en la figura 2, la posibilidad de resaltar en negrita o con un determinado color las señales que nos interese, ofrece una vistosidad muchas veces necesaria cuando se quiere presentar el comportamiento de los circuitos digitales.

Los botones “señales” y “pulsos” nos permiten incrementar o decrementar el número de señales o el número de pulsos a visualizar. Un “-“delante del nombre asignado a una señal, nos representará la forma complementaria de la misma.

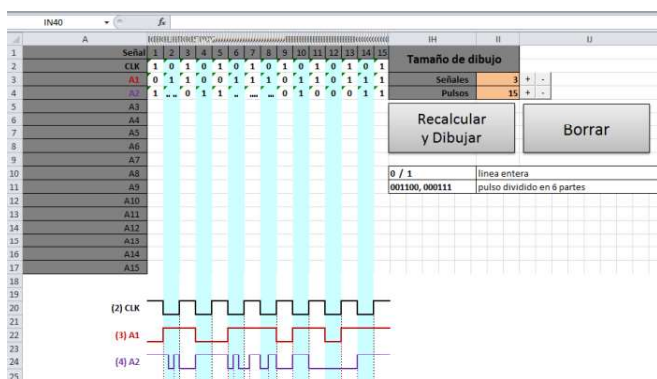


Fig. 2. Entorno de diseño de cronogramas

#### A. Lógica de representación gráfica

Durante la ejecución de “Recalcular y Dibujar”, el código convierte cada señal en 6 números. (1 -> 111111, 0->000000 o toma las entradas explícitas de 6 valores, tal como se hayan definido). A continuación la macro coge cada trozo de un pulso y formatea el borde de la celda de este trozo con una línea continua. Si es un 1, formatea el borde de arriba, si es un 0 el borde de abajo.

De igual forma, la macro controla si el valor anterior es distinto del actual, lo cual significa que hay un cambio de señal de 0 a 1 o viceversa. En este caso la macro formatea el borde de la izquierda del trozo con una línea continua o el de la derecha. El resultado es la visualización de una gráfica que con las características numéricas, intensidad y color, definidas

previamente, va mostrando su forma gráfica por periodos, con la forma correspondiente al valor o valores asignados en esa columna temporal. De esta forma es muy fácil visualizar la correlación entre valor y forma de onda, para su posible modificación en caso de estar interesados en ello.

Cabe significar que tal como se ha mencionado antes, la representación permite 6 subespacios dentro del periodo asignado a una columna, esto no ofrece multitud de posibilidades a la hora de definir una forma de señal que puede tomar hasta 12 valores en el periodo de una hipotética señal de reloj definida. Esto nos permitiría que la señal cambiara de valor hasta 12 veces dentro de un periodo de reloj.

#### B. Entorno de datos y programación

Este programa usa las funcionalidades del programa “Microsoft Excel 2010” para mostrar la lógica de señales de entrada y salida. En el entorno de Excel tenemos una plataforma familiar para insertar datos fácilmente y representar soluciones gráficamente. Vía el entorno VBA que es implementado en cada producto de Microsoft Office, disponemos de una herramienta potente para calcular expresiones difíciles y manejar trabajos y lógicas complejas mediante la programación de macros.

### III. SIMULACIÓN DEL COMPORTAMIENTO DE CIRCUITOS COMBINACIONALES

Una vez que se tenía la posibilidad de generar cronogramas de forma fácil a nuestra voluntad, se pensó en potenciar la aplicación con la posibilidad de ofrecer la respuesta de salida de circuitos digitales simples. Desde nuestro punto de vista esto ofrecía también una forma fácil de que los alumnos pudieran comprobar la respuesta de circuitos digitales, que en las primeras fases de su aprendizaje requiere experimentar con distintos casos a fin de afianzar su comprensión. De igual forma, la exposición en clase puede ser más comprensible si acompañamos la exposición del comportamiento de un circuito con la visualización de la salida, tras la definición de señales de entrada.

Con este objetivo, se ha aportado la funcionalidad de establecer relación entre las señales definidas en distintas filas del entorno de diseño. Esto supone incrementar la opción didáctica de la herramienta por medio de ofrecer de forma visual la señal que obtendríamos a la salida de un circuito combinacional o secuencial. Basta con insertar funciones típicas constitutivas de circuitos SSI como puertas lógicas o biestables.

El análisis de circuitos electrónicos, tanto digitales como analógicos lo abordan distintas herramientas de uso generalizado, las cuales están preparadas también para diseñar circuitos a nuestra medida, utilizando módulos existentes en el mercado. Como hemos mencionado anteriormente, PSpice o Logic Works son claros ejemplos de herramientas de gran utilidad para la enseñanza de estas materias y con una potencialidad que permite ejemplificar lo explicado en distintas asignaturas del grado de Informática. Sin embargo, estos programas nos facilitan a nuestro entender, una salida

gráfica con el carácter didáctico deseable para unos primeros niveles de aprendizaje.

Otras herramientas como VerilUOC [6] permiten además chequear la bondad del diseño de un circuito, realizado como tarea por los alumnos, facilitando en gran medida la autoevaluación y trabajo autónomo del estudiante.

En nuestra herramienta, dependiendo del dispositivo digital en el que se quiera visualizar su comportamiento ante señales presentadas a su entrada, el código analiza las señales que hemos definido y escribe los valores de salida en el espacio correspondiente a cada pulso en el espacio de visualización, cumpliendo las reglas lógicas de los elementos Flip-Flop o puerta lógica con el que estemos trabajando. Esta lógica está programada mediante operaciones elementales (suma, multiplicación, *IF then else*, bucles,...) y los comandos internos de Microsoft Excel para acceder a las celdas, sus contenido (valores, texto), sus propiedades (color, fuente), etc.

En cada una de las 6 partes de un pulso, el código analiza las señales actual, anterior y siguiente, calculando a continuación el valor de salida respecto de lógica de Flip-Flop o Puerta Lógica. La macro guarda el valor temporalmente para generar las 6 partes de la salida, por ejemplo 001010. Al final la macro escribe este número en la correspondiente señal de salida y pulso actual del elemento digital que estemos utilizando.

Además del cálculo de valores de salida resultado del comportamiento de Flip-Flops (JK, SR, D) y puertas lógicas (AND, OR, etc.), la herramienta de diseño ofrece como complemento un simulador gráfico que posibilita la visualización de los cronogramas diseñados a medida de forma totalmente libre.

Para iniciar un cálculo se tiene que definir el elemento que se quiere representar. Introducir un texto que empieza con dos “##” abre el formulario para puertas lógicas. De igual forma, si ponemos dos “\*\*” abre el formulario para Flip-Flops.

El siguiente cuadro muestra los distintos parámetros que intervienen en la definición del comportamiento de Puertas Lógicas. En la definición únicamente hay que indicar el tipo de puerta que queremos analizar y las filas del entorno Excel en el que se encuentran las distintas señales que actuarán como entrada de la puerta. Esto a su vez configura el número de entradas que tendrá la misma, sin necesidad de añadir ningún parámetro más.

TABLA1. PARÁMETROS QUE INTERVIENEN EN EL COMPORTAMIENTO DE PUERTAS LÓGICAS

Puertas Lógicas: ## X [t,u,v,w,...]	
##Elemento puerta Lógica	
X Tipo de puerta (AND, OR, NOR, ...)	t, u, v,.... Filas de entradas de la puerta
Las entradas son en forma “+i”	
+ (¿entrada negada?, “+”=NO, “-”=SI)	
i (fila de señal de entrada en Excel [i = fila i en Excel])	

Ejemplo:

##OR[+2;+3;+4;+5;+6] Puerta OR con 5 entradas: la entrada A es la fila 2, la entrada B es la fila 3, la entrada C es la fila 4, la entrada D es la fila 5 y la entrada E es la fila 6. De igual forma aparecen definidas las otras funciones combinacionales de la figura 3.

Como podemos comprobar en esta figura, para obtener la señal de salida correspondiente a un conjunto de valores digitales a la entrada, es necesario indicar la función que queremos (AND, NOR, etc.) precedida de ##. La aplicación interpretará estos caracteres como una función a representar y abrirá una ventana de diálogo (figura 4) para determinar que parámetros intervienen en la función. En este caso los parámetros serán el tipo de función a representar y las variables de entrada, identificadas por las filas en las que se encuentren cada una de las señales que nos interese utilizar. Una selección de las mismas y la pulsación del botón “Crear”, dejará la aplicación a expensas del momento en que nosotros queramos visualizar gráficamente el resultado pulsando el botón “Recalcular y Dibujar”.

Un botón “Añadir negada” nos posibilita introducir la señal seleccionada de forma complementada. Podremos observar así mismo que tras la pulsación del botón “Crear”, la fila en la que iniciamos nuestra propuesta de implementar la función, aparece con los parámetros asignados en la ventana de diálogo.

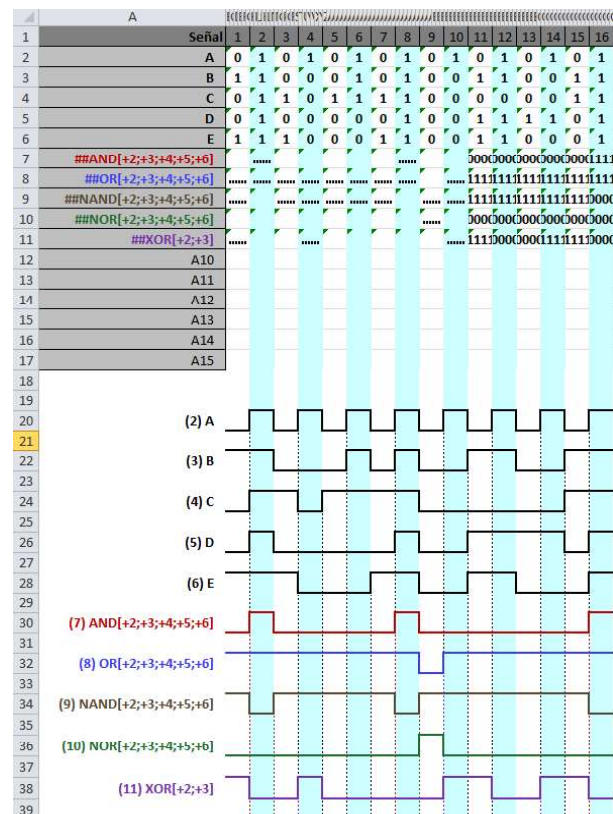


Fig. 3. Representación de funciones combinacionales

Por ejemplo

##AND[+2;+3;+4;+5;+6] indica que se visualizará la función AND de 5 señales, las cuales se encuentran en las filas 2, 3, 4, 5 y 6, no estando ninguna de ellas complementada, ya que en ese caso el número de fila vendría precedido por un signo (-).

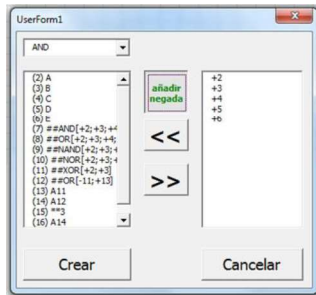


Fig. 4. Ventana de diálogo de una Puerta Lógica

#### IV. SIMULACIÓN DEL COMPORTAMIENTO DE CIRCUITOS SECUENCIALES

De igual manera la aplicación está preparada para ofrecer la salida de circuitos biestables utilizados en sistemas digitales. En este caso en la fila en la que se quiera representar la respuesta de un determinado biestable, debe iniciarse con un doble \*. Inmediatamente se nos abre un formulario en una ventana de diálogo, tal como se ve en la figura 5, la cual nos permite definir los parámetros que intervienen. En este formulario tenemos que definir los elementos relevantes que determinan el funcionamiento del dispositivo. En el caso de Flip-Flops son por ejemplo el tipo (pulso, flanco), entradas A y B, posible negación de las entradas, valor predeterminado del estado del Flip-Flop, valores predeterminados de los señales (antes del primer pulso), si tiene pin *Enable*, *Preset* o *Clear*,....

Dependiendo de la configuración, la macro crea una llave de texto y escribe en la celda con el nombre de la señal para que el simulador gráfico pueda representar los valores correctos.

La ventana de diálogo que se abre para definir los parámetros del biestable en el que estemos interesados ofrece un conjunto de posibilidades representativas de las características que nos podemos encontrar al trabajar con este tipo de dispositivos.

En primer lugar debemos seleccionar el tipo de Flip-Flop, a continuación si queremos que actúe por pulsos o por flanco. Así mismo deberemos seleccionar que señales actúan como cada una de las entradas del biestable, si estas tienen su entrada negada o no y su valor inicial antes de que llegue el primer pulso activo.

En la TABLA 2 se muestran los distintos parámetros que intervienen en la definición de Flip-Flops.

Ejemplo:

\*\*JK[0;0+3,0+4,0+2,0,0,0] FF JK, valor antes 0, entrada filas 3 (J) y 4 (K) no negadas, CLK es fila 2, no tiene activado ni *Enable*, ni *Preset*, ni *Clear*.

\*\*D[1;0+3,0+2,0,0,0] FF D, valor antes 1, entrada A es fila 3, entrada B no existe, CLK es línea 2, no tiene activado ni *Enable*, ni *Preset*, ni *Clear*.

TABLA2. PARÁMETROS QUE INTERVIENEN EN LA DEFINICIÓN DE FLIP-FLOPS

Flip-Flops: **X Y   t ; u ; v ; w ; z ; o ; p	
<b>** Elemento Flip-Flop</b>	v = fila entrada B (K,R,-)
X FlipFlop (JK,SR,D)	w = fila entrada CLK
Y Activación (P=pulso, F=flanco)	z = fila entrada ENABLE
t Valor inicial del Flip-Flop	o = fila entrada PRESET
u = fila entrada A (J, S, D),	p = fila entrada CLEAR
	si u, v, w, z, o, p = 0 -> no tiene este PIN
<b>Las entradas son en forma "0+i"</b>	
0 (valor inicial, puedes ser 0 o 1)	
+ (¿entrada negada?, "+"=NO, "-"=SI)	
i (fila de señal de entrada en Excel)	

Al igual que con la ventana de diálogo de puertas lógicas, una vez seleccionados los parámetros necesarios y pulsar "Crear", la aplicación está en disposición para visualizar la respuesta de salida del Flip-Flop seleccionado.

Se muestra en la figura 5 un ejemplo práctico en la ventana de diálogo. Las opciones nos permiten seleccionar si lo que queremos utilizar es un Flip-Flop (biestable síncrono) o un Latch (asíncrono). En el caso de que hayamos seleccionado activación por flanco, la opción de "activación por nivel" pulsando alto o bajo debe ser interpretada como ascendente o descendente respectivamente.

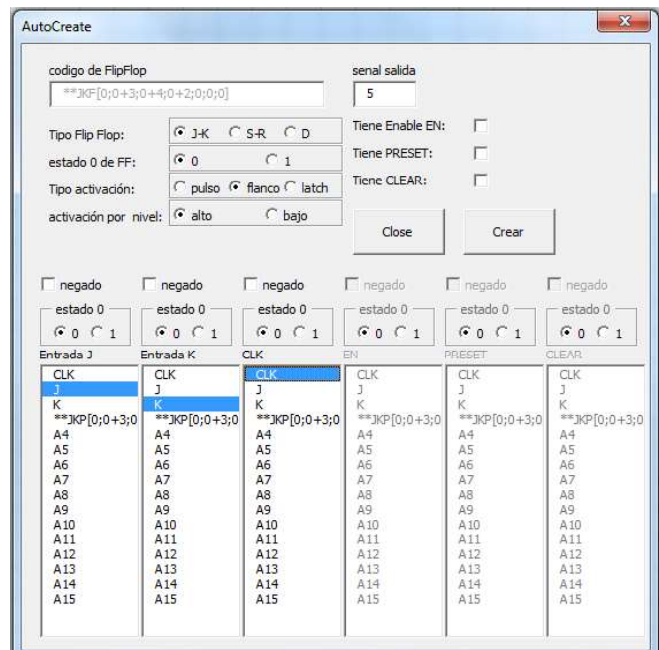


Fig. 5. Ventana de diálogo para la definición de biestables

Es conveniente recordar que el botón con la denominación "negado" nos permitirá definir la respectiva señal activa en bajo simplemente marcándolo. Esto es de especial importancia para la definición de las señales *Preset* y *Clear*, normalmente activas en bajo en los circuitos comerciales (*Preset* y *Clear*).

La figura 6(a) visualiza la representación del comportamiento de este Flip-Flop ante la configuración de valores digitales indicada. En este caso la señal de la fila 2 actúa como señal de reloj, siendo las señales de las filas 3 y 4, las entradas J y K del Flip-Flop respectivamente.

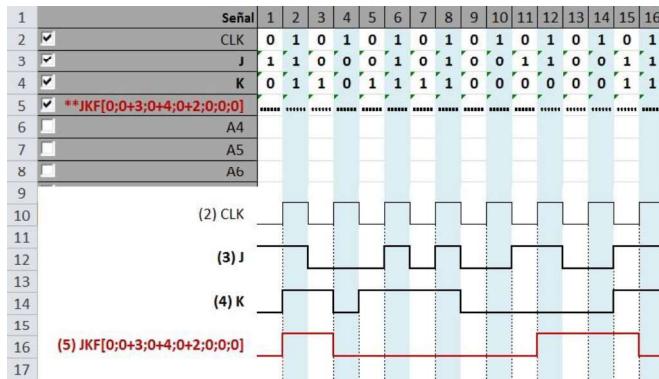


Fig. 6. (a) Representación de Flip-Flop J-K activado por flanco ascendente

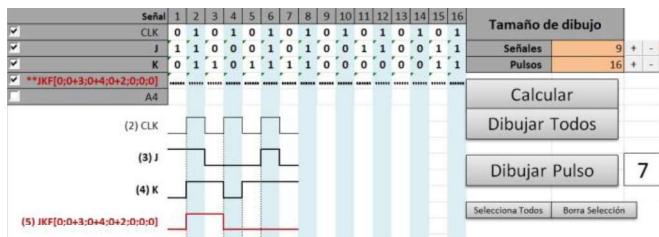


Fig. 6. (b) Visualización por pulsos

Se puede comprobar como en la fila correspondiente aparecen los parámetros de la señal visualizada: JKF[0;0+3;0+4;0+2;0;0;0], para indicar Flip-Flop J-K activo por Flanco, con entradas las filas indicadas anteriormente (3 y 4) y señal de reloj la señal de la fila 2. Los tres últimos parámetros que toman el valor 0 nos indican que en esta representación no existe una señal de Enable, ni Preset, ni Clear.

Las ventajas didácticas de la aplicación se ponen de manifiesto cuando en la exposición del funcionamiento de un determinado dispositivo hacemos que la visualización se realice por tramos en vez de la representación completa de una sola vez. Estos tramos pueden ser configuraciones de valores de entrada, si estamos ante circuitos combinatoriales o tramos temporales (pulso tras pulso) si estamos ante circuitos secuenciales, como se puede observar en la figura 6(b).

Como complemento a la aplicación también se le ha dotado con la posibilidad de generar señales de reloj de forma automática, para lo cual escribiendo la palabra clave CLK en una de las filas de la ventana de diseño, no aparecerá una nueva ventana de diálogo en la cual podremos determinar el número de ciclos que queremos representar, así como la duración de los periodos de bajo y alto de la señal de reloj.

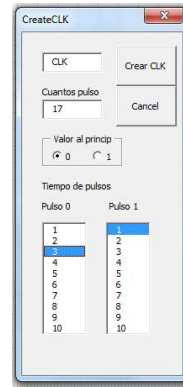


Fig. 7. (a) Ventana de creación de una señal de reloj



Fig. 7. (b) Señal de reloj generada

De esta forma, podremos generar señales de reloj totalmente regulares, en las que los tiempos en alta y baja de la señal de reloj son iguales en el espacio de un periodo, o desiguales dando un valor distinto al seleccionar la duración del pulso 0 y 1 en la ventana de creación del reloj. Esta asimetría se repetirá a continuación de forma periódica.

Por ejemplo, la selección de valores marcada en la ventana de creación de reloj de la figura 7(a), nos da como resultado la visualización de la señal que podemos ver en la figura 7(b).

A. Funciones

Es importante resaltar que toda señal generada como resultado de la intervención de una puerta o Flip-Flop, puede ser utilizada en una nueva fila del visualizador como entrada de otra función combinatorial o secuencial. Con ello tenemos la posibilidad de demostrar el comportamiento de la interconexión de varios elementos, con la única limitación de las 16 señales mencionadas como límite anteriormente, si bien esto es fácilmente ampliable si fuera necesario.

En la figura 8 se muestra un ejemplo de generación de la función combinatorial  $f = \bar{a}b + c$

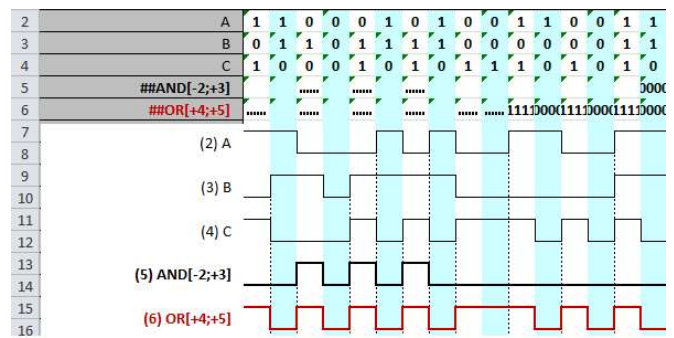


Fig. 8. Ejemplo de salidas de la función combinatorial  $f = \bar{a}b + c$  (en rojo)

De igual forma la demostración del comportamiento de agrupaciones de Flip-Flops es totalmente factible, con lo cual completamos el conjunto de elementos básicos que pueden ser manejados en electrónica digital para un aprendizaje práctico del funcionamiento de estos circuitos.

La comprobación de la forma de actuar tanto de contadores síncronos y asíncronos, como de registros de desplazamiento es totalmente factible simplemente seleccionando como entradas las salidas de una función generada anteriormente, lo cual es válido para entradas de datos y para entradas de reloj. Una selección de una señal Q de salida de un Flip-Flop como entrada de reloj de otro Flip-Flop, permitirá diseñar circuitos asíncronos.

Por ejemplo en la figura 9 se muestra la representación de las formas de onda en las salidas Q de cada uno de los Flip-Flops que constituyen un registro de desplazamiento de 4 bits con entrada serie D.

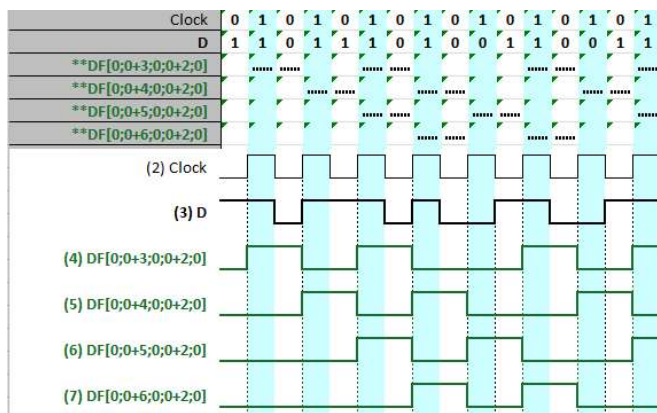


Fig. 9. Ejemplo de salidas de un registro de desplazamiento de 4 bits

## V. CONCLUSIONES

El manejo de cronogramas funcionales es de gran utilidad en el aprendizaje de electrónica digital, siendo necesario un procedimiento sencillo de generación de los mismos.

Se ha desarrollado una herramienta de gran versatilidad para definir patrones de señales de entrada y analizar el comportamiento de circuitos digitales.

La herramienta permite la definición libre de cronogramas y su utilización como variables de entrada para observar la señal de salida generada.

La definición libre de patrones de señales me permite ejemplificar de detalladamente la forma en que actúan los circuitos digitales.

La herramienta muestra su poder didáctico en la presentación de supuestos simulados de forma gráfica en clase.

Así mismo, la herramienta facilita la elaboración de supuestos de trabajo escritos, destinados a la resolución autónoma o colaborativa por parte de los estudiantes.

Su utilización ha sido aplicada de forma experimental durante un curso académico por parte de los profesores de la asignatura Estructura y Tecnología de Computadores en el Grado de Informática, UAL, estando su puesta a disposición de los alumnos en el presente curso.

## ACKNOWLEDGMENT

Este Trabajo ha sido desarrollado en el marco “Convocatoria de Grupos Docentes para la Creación de Materiales Didácticos en la Universidad De Almería. Bienio 2017-2018”.

## REFERENCES

- [1] <http://www.pspice.com/about>
- [2] <http://designworkssolutions.com/logicworks-5-windows/>
- [3] J.D. Baneres, “Aprendiendo jugando fundamentos de sistemas digitales,” Actas de las XXII Jenui. Almería, 6-8 de julio 2016. ISBN: 978-84-16642-30-4, pp. 269-276
- [4] M. Ruiz-Ferrándiz, G. Ortega, J. Roca-Piera. “Learning Analytics and Evaluative Mentoring to increase the students' performance in Computer Science”. EDUCON2018-IEEE Global Engineering Education Conferenc. Santa Cruz de Tenerife, april, 2018. En prensa.
- [5] M. Prim, J. Oliver, y V. Soler, “Aprendizaje de Sistemas Digitales utilizando tecnologías interactivas,” IEEE-RITA Vol. 4, Núm. 1, Feb. 2009.
- [6] D. Baneres, R. Clarisó, J. Jorba, and M. Serra, “Experiences in digital circuit design courses: A self-study platform for learning support,” in IEEE Transactions on Learning Technologies, vol. 7, no. 4, pp. 360–374, Oct 2014.
- [7] G.A. Contreras, R. García y M.S. Ramírez, “Uso de simuladores como recurso digital para la transferencia de conocimiento,” Apertura Revista de Innovación Educativa, ISSN Impreso: 1665-6180 Número 1, Volumen 2, Universidad de Guadalajara, México, Abril 2010
- [8] C. Quintáns, J. Fariña, J.J. Rodríguez-Andina, “Aprendizaje Práctico de Sistemas Electrónicos Digitales a través de Proyectos Semiguizados,” IEEE Revista Iberoamericana de Tecnologías del Aprendizaje. Electrónico ISSN: 1932-8540 Issue: 99. 31 July 2017.