

Transcripciones de las presentaciones de clases de teoría 2016

IMPORTANTE: Estas notas de clases sirven como complemento de los apuntes ya editados por esta cátedra y no deben ser considerados como el material didáctico final a estudiar. Se aprovecha en las mismas refrescar ciertos conceptos vertidos en los mismos, complementarlos y actualizarlos.

TEMA 10: Análisis y Síntesis

Análisis:

Filmina 2: El análisis de un circuito, dispositivo ó sistema busca entender el funcionamiento de los mismos ya sea para saber cuál es su función o como medio de realizar un test para comprobar si el diseño realizado cumple con nuestras expectativas.

Los circuitos combinatorios son los más fáciles de analizar siempre y cuando no sea elevado el número de entradas y/o salidas del mismo.

Generalmente se emplea la tabla de verdad, pero también se puede realizar una simulación por computadora con software como Orcad, Pspice, u otras herramientas de fabricantes de circuitos digitales programables (Altera, Xilinx, etc.).

El test de hardware es el más realista ya que implica probar el circuito en condiciones reales, pero a veces requiere un conocimiento preciso del entorno alrededor del circuito para poder realizar las pruebas y en algunos casos, por ejemplo de falla, esta opción se torna imposible.

Un recurso utilizado es el del relevamiento del mismo buscando en las hojas de datos la identificación de los componentes junto con la inspección de cómo se conectan las entradas y salidas y a donde. Asociado a todo método de análisis existe una componente heurística, es decir, basados en experiencias previas y aplicando el sentido común es posible deducir cuál es la función de un dado circuito o encontrar la causa probable de falla por mediciones secuenciales en puntos de prueba estratégicos.

Filminas 3 a 5: El circuito mostrado se puede analizar, planteando las ecuaciones de entrada a los FFs.

En este caso hay dos.

El análisis del de arriba, es fácil ya que funciona como "T" con entrada en "1" por lo que cambia su salida en cada flanco de reloj.

El segundo, es más difícil ya que si bien actúa como "T", su entrada depende de la señal X y de las salidas Q y /Q del FF superior.

Para conocer el comportamiento de ambos FFs se pueden plantear la ecuación general de un FF "JK" que vincula la salida con las entradas:

$$y(n+1) = J(\overline{y(n)}) + K(y(n))$$

Donde $y(n+1)$ es la salida Q futura que aparecerá con el nuevo flanco de reloj e $y(n)$ es la salida actual que junto con los valores actuales de J y K darán lugar al cambio de $y(n+1)$

Planteando esta ecuación y sustituyendo llegamos a la conclusión que el FF superior genera una onda cuadrada de la mitad de frecuencia que la del reloj sin importar el valor de la entrada X.

Por otro lado, el FF inferior tendrá un comportamiento diferente si $X=0$ ó si $X=1$.

Realizando un diagrama de tiempos para ambos casos se puede notar que las salidas Q1 y Q2 tienen una evolución como las de un contador binario de 2 bits progresivo o regresivo dependiendo de X.

Se puede afirmar entonces que el circuito es un contador binario de 2 bits con control de sentido de conteo.

Filminas 6 y 7: Una estructura muy utilizada para encarar diseños lógicos que involucran señales de reloj y en particular realizar trabajos en forma secuencial se denomina "Máquina de Estados".

Existen dos modelos ampliamente utilizados. En ambos casos cada modelo está compuesto por una parte de lógica combinatoria y otra de memoria (Flip-Flops)

Modelo de Mealy: Donde las salidas del circuito responden a las entradas y al estado actual.

Se entiende como estado a cada una de las combinaciones posibles que se puedan formar con los elementos de memoria. Por ejemplo: Si un circuito está formado con 3 FFs, es decir, donde existen 8 combinaciones posibles de sus salidas Q, se dice que como máximo, ese circuito tendrá 8 estados posibles (N° de estados = 2^N , donde N es el número de FFs del circuito).

Modelo de Moore: En este modelo las salidas dependen sólo del estado en el que se encuentren. A diferencia de Mealy. Dado un estado, las salidas asumirán un dado valor y no habrá cambio alguno hasta que el circuito cambie de estado. Se dice que las salidas responden en forma sincrónica con el reloj. En Mealy las salidas aún estando en un dado estado, pueden cambiar si las entradas así lo indiquen. Se dice en ese caso que las salidas responden en forma asincrónica.

Filminas 8 a 9: En este caso, como el circuito es secuencial y sincrónico (todos los FFs conectados a una misma señal de reloj) se analizará el circuito por el método de diagrama ó tabla de estados.

Se llama “estado” a una determinada combinación de las salidas de los FFs.

Un solo FF tiene 2 estados posibles 0 y 1, que los podemos llamar S0 y S1.

Dos FFs tienen 4 estados posibles, 00, 01, 10 y 11, llamándolos S0, S1, S2 y S3 respectivamente.

Tres FFs tendrán 8 estados posibles, desde 000 hasta 111, y así siguiendo.

En general, un circuito síncrono con N FFs tendrá 2^N estados posibles como máximo, esto significa que no todos deben estar presente. Eso depende del diseño.

Para analizar este circuito, como antes, se plantean las ecuaciones de salida de los 3 FFs.

La idea es armar una tabla ó diagrama, con los estados presentes en el circuito y ver como se pasa de uno a otro en cada flanco de reloj .

Como de las ecuaciones tenemos para cada FF como responderá la salida según su valor actual de dicha salida y sus entradas, podemos ir armando el rompecabezas.

Por ejemplo para el FF0 de las ecuaciones sale que la salida Q0 cambia de valor en cada flanco de reloj.

La salida del FF1 cambiará cuando venga el siguiente flanco de reloj si el valor actual de Q1 es “1”.

El FF2, de manera parecida, cambiará sólo cuando Q0 y Q1 valgan ambos “1”.

De esta manera, se ponen en una columna los estados posibles desde S0 hasta S7 ya que hay 3 FFs. En otra columna, los valores actuales de las salidas Q2Q1Q0 y en la última los correspondientes a los valores futuros de dichas salidas.

En forma gráfica, un círculo representa un estado y las flechas indican cómo cambia el estado del circuito cada vez que viene un flanco de reloj nuevo.

Si no hay flanco, el circuito no se mueve del último estado al cual llegó, manteniendo los mismos valores de los Q.

En este ejemplo, dentro del círculo se anota el valor de las salidas del circuito que “coincide” con el de las salidas de los FFs. Esto no siempre es así, ya que las salidas pueden ser una combinación lógica de las salidas de los FFs (caso del Modelo de Moore) ó incluyendo a las entradas de datos (caso del modelo de Mealy).

De la inspección se puede concluir que el circuito es un contador binario progresivo de 3 bits.

Filminas 10 a 12: Este ejemplo trata de un circuito síncrono que tiene una entrada de datos “Y” y una salida “Z”. Posee dos FFs tipo “D” por lo que como máximo podrán haber 4 estados posibles (el número exacto dependerá del diseño).

Para analizarlo por tabla de estados, partimos de plantear las ecuaciones como antes: una es la de la salida Z y las otras son las de las salidas de los FFs Q0 y Q1.

La ecuación de un FF "D" es $Q(n+1) = D(n)$, por lo que se simplifica el análisis respecto del ejemplo anterior con los FFs "JK".

Tanto las ecuaciones de Z como las de Q0 y Q1, dependen de Y por lo que conviene hacer una división para $Y=0$ e $Y=1$.

De esta manera se arma la denominada tabla de estados donde por en la 1ra columna se ponen los 4 estados posibles (S0, S1, S2 y S3), en la segunda los valores de las entradas de los FFs actuales que según el valor de Y y de Q1Q0 generarán un par de valores nuevos de dichos Q.

La salida Z queda definida por lo que valen Q0, Q1 e Y.

El nuevo estado saldrá de dicha tabla: por ejemplo si estoy en el estado S0 donde $Q1Q0=00$ y la entrada $Y=1$, entonces la salida valdrá $Z=0$. Los FFs pasarán a valer $Q1Q0=01$ recién cuando venga el próximo flanco de reloj.

Del diagrama obtenido se puede pensar que si Y es sincrónica con el reloj, sólo cuando se presente la combinación $Y=1 \rightarrow 0 \rightarrow 0$, la salida pasará de valer $Z=0$ a $Z=1$.

Puede tratarse, entonces, de un detector de secuencia serie del dato Y.

Se ha hecho una simulación utilizando el MaxPlus-II, donde se puede verificar con diferentes valores de Y como cambia la salida Z.

Síntesis:

Filmina 13: La síntesis es la operación de generación de hardware y conlleva la misión de simplificar el circuito a la mínima expresión posible.

Generalmente hoy en día, al menos hay 3 parámetros que son considerados de importancia a la hora de sintetizar un circuito digital: velocidad de respuesta, consumo y recursos utilizados ya sea en compuertas (circuitos lógicos programables o con lógica estándar) ó área de Silicio en caso de diseño tipo "custom" (a medida) de chips.

Existen diferentes métodos para realizar la síntesis donde podrá variar el nivel de participación del diseñador en el desarrollo.

En el diseño de circuitos combinatorios para lógica **estándar**, podemos emplear fácilmente métodos heurísticos basados en tablas de verdad, ecuaciones ó diagramas de Karnaugh ó algoritmos que permitan simplificar las funciones lógicas en forma tabular como el veterano método Quine-McKluskey (ver libro de Hill-Peterson) u otros más nuevos, basados en él.

En todos estos casos, el diseñador debe plantear el problema y realizar la síntesis basado en su conocimiento de los diversos tipos de circuitos que existan en el mercado, tratando de optimizar su diseño (en general) con la menor cantidad de chips posibles, dado que esto

implicará menor consumo, menor área de impreso, menores tiempos de retardo, entre otros factores.

En la síntesis de circuitos secuenciales sincrónicos puede ser necesario emplear además otro tipo de método como el de diagrama ó tabla de estados.

En el caso de utilizar circuitos lógicos programables, dado el elevado nivel de complejidad en cuanto a la programación de los chips, el fabricante ofrece al diseñador herramientas de software para el desarrollo de los mismos, generalmente con las opciones de entrada de datos por esquemático ó por lenguaje de descripción del hardware (HDL) (éste último ya altamente empleado desde hace varios años, por su versatilidad en diseños complejos).

Filmina 14: En el diseño de circuitos secuenciales sincrónicos por diagrama de estados ó tabla de estados existen dos modelos a emplear como se comentó anteriormente:

a) Modelo de Máquina de Mealy donde las salidas del circuitos dependen de las entradas y del estado en que se encuentra el circuito

b) Modelo de Moore, donde las salidas del circuito sólo dependen del estado actual del circuito. Lo que sí depende de las entradas y del estado actual del circuito es el estado siguiente al aparecer el nuevo flanco de reloj.

Filmina 14: Aquí se ejemplifica un diagrama de estados que utiliza 3 estados y un tercero que si bien no forma parte del diseño se debe definir ya que hay que utilizar dos FFs y sobra un estado de los 3 necesarios, el cual no puede quedar indefinido (salvo excepciones como veremos más adelante, que implica un reset inicial para empezar siempre de un estado conocido).

Si se inspecciona con detenimiento, hay un estado que es redundante ya que se puede realizar el mismo trabajo pero anulando uno de ellos.

Aquí el estado S1 es superfluo ya que podemos hacer lo mismo de arriba, si vamos de S0 a S2 cuando la entrada vale 1; nos quedamos en S2 cuando siga valiendo 1, y pasamos a S0 nuevamente cuando la entrada pase a 0. Y la salida vale 1 sólo al pasar de S0 a S2.

Lo que hemos logrado en este caso es eliminar un FF ya que pasamos de 3 a 2 estados.

Independientemente de si hay o no redundancia, el modelo que usamos aquí para describir lo que queremos hacer, es el de Mealy.

Cada círculo es un estado diferente y nos movemos de uno a otro, indicándolo con una flecha con la combinación "entradas"/"salidas".

Filmina 15 a 19: Aquí se diseña un contador binario síncrono de 2 bits con programación del sentido de conteo a través del estado de la variable de entrada "X".

Primero se especifican las características deseadas:

Estructura a diseñar: Contador.

Tipo: síncrono

Formato: binario

Modalidad: Progresivo ó regresivo

Longitud de palabra: 2 bits

Controles: Entrada de selección para sentido del conteo, tal que si:

"X"=0 cuenta en modo regresivo.

"X"=1 cuenta en modo progresivo.

Reset: Ninguno.

Modelo de la máquina de estados a utilizar: Mealy.

Tipo de FF a utilizar: "D"

Realizamos el diagrama de estados. Como el contador es de 2 bits, deberemos tener 4 valores de conteo diferentes y asociaremos cada uno de esos valores a un estado diferente de los FFs que constituyen la máquina de estados.

Por lo tanto necesitamos 2 FFs.

Designamos entonces 4 estados desde S0 hasta S3, usándolos todos.

Si X=1, iremos de S0 a S1 en un flanco de reloj. Luego de S1 a S2, en otro y así siguiendo hasta S3, donde todo vuelve a empezar.

Lo mismo sucede si X=0 pero al revés.

Se realiza la tabla de excitación donde se debe generar la secuencia del próximo estado y el de las salidas para cada uno de los 4 estados que tenemos.

Estando en S0, si X=0 debemos ir a S3 y las salidas deberán valer 11 y para ello como se trata de FFs copiadores debemos cargarlos con D1D0=11.

Estando en S0, si X=1 debemos ir a S1 y las salidas deberán valer 01 y para ello, del mismo modo debemos cargar a los FFs con D1D0=01.

Y así siguiendo en cada uno de los otros 3 estados.

Construimos los digramas de Karnaugh para hallar los valores futuros de D1y D0 necesarios para pasar de un estado a otro según la entrada X.

NOTA: En este ejemplo hemos a propósito hecho coincidir las salidas del contador con las salidas de los FFs a fin de simplificar la lógica. En general, en Mealy las salidas son una combinación lógica entre los Q de los FFs y las entradas de datos.

Para terminar basta con sintetizar la lógica de la manera anteriormente explicada, es decir, elegir el tipo FF, realizar la tabla de excitación y por último sintetizar las salidas y hallar cada una de las ecuaciones que vinculan las entradas de los FFs con las demás variables (la entrada de datos y las salidas de los FFs).

Filmina 20 a 21: El diseño por Mealy de un monoestable que se dispara con flanco de subida de la entrada X, empleando FFs JK. Cada flanco ascendente detectado tiene que hacer que la salida se ponga a "1" durante un ciclo del reloj del circuito de la máquina de estado.

Aquí se plantea el problema con el diagrama de estados donde se requieren dos estados.

Hacemos la tabla de transición del FF JK que es el planteo al revés de la tabla de verdad del mismo. Esto es, saber que hay que poner en las entradas JK para que el FF vaya de un estado a otro.

Por ejemplo, si estoy con $Q_n=0$ y quiero ir a $Q_{n+1}=0$ en el siguiente flanco de reloj, tengo dos opciones: o fuerzo la salida a 0 ó mantengo el estado anterior, lo que significa $JK=0X$ (indiferente el valor de K para este caso).

Así se completan las otras 3 opciones de ir desde Q_n a Q_{n+1} .

NOTA: La tabla de transición del FF tipo D es obvia ya que dado un estado futuro al que se quiere llegar sólo hay que poner ese valor actual en D, pues se trata de un FF copiador.

Una vez hecho esto se hace la tabla de excitación, definiendo en cada estado actual a que estado voy a ir y cuál es la salida actual dado un valor de la entrada X.

Aquí Z sólo está en "1" cuando estando en S0 la entrada vale "1".

El circuito queda con un solo FF y dos compuertas.

Notar que como la entrada puede ser asincrónica con el reloj del circuito ya que generalmente se quiere detectar eventos externos al detector, puede ocurrir que estando en S0 como se indica en rojo en la filmina 19, X pase a "0" luego del "1" antes del próximo flanco de l reloj y la salida vuelva a ponerse en "0" lo cual produciría un efecto indeseado.

Filmina 22 a 24: El mismo diseño hecho con Moore, donde la salida depende sólo de las salidas de los FFs. Sólo el próximo estado depende en principio de la entrada X en este caso (además del estado actual en que se encuentre).

Aquí tenemos que usar un estado mas ya que si se detecta una transición de 0 a 1 (flanco ascendente) la salida debe permanecer en "1" sólo un ciclo de reloj y después bajar.

Como nos quedó el diseño con 3 estados, debemos usar 2 FFs y con éstos podemos definir hasta 4 estados. Entonces el cuarto aquí lo definiremos como que si aparece no importa que valga la entrada, vaya al estado S0 del inicio y nos aseguraremos que el circuito funcione siempre correctamente (por ejemplo, el S3 puede aparecer durante el encendido del circuito: si no lo definimos así el resultado puede ser impredecible).

Usando FFs tipo "D" armamos la tabla de excitación donde debemos generar los estados futuros del circuito para cada combinación de las salidas actuales de los mismos y la entrada X.

La salida Z, como es Moore, estará definida según el estado en el que nos encontremos. En este caso $Z=1$ sólo si estamos en el estado S1. En los restantes será $Z=0$. Lo que implica que Z es la AND entre Q0 y $\overline{Q1}$.

Filmina 25 a 26: Dado el diagrama de estados indicado se pide sintetizarlo con Moore usando un FF tipo JK.

El resultado es que se necesita sólo un FF JK sin lógica combinatoria adicional. La conclusión es que se trata del diagrama de estados de un FF JK donde J es A y K es B.

Filmina 27: Lo mismo para un FF tipo D.

Filmina 28: Diseño con máquina de estados Moore de un FF JK basado en uno tipo D.

Notar que la lógica combinatoria corresponde al circuito interno de un MUX 2:1.

Filminas 29 a 31: Diseño de un contador binario progresivo-regresivo de 2 bits por Moore con FFs tipo D y tipo JK.

Para este caso con tipo D obtenemos un circuito mas simple además de ser mas fácil la realización de la tabla de excitación, donde con los JK es posible de cometer errores en la tabla o en la síntesis del Karnaugh.

Filminas 32 a 39: Diseño de un detector de secuencia de una entrada sincrónica con el reloj tal que cuando se detecte la secuencia 1011 se deberá avisar, poniendo una salida Z en 1 durante un ciclo del reloj.

En la filmina 33 se diseña el diagrama de estados:

Estando en el inicio en S0 quedamos a la espera con la salida en "0" y nos quedamos en ese estado hasta que no se detecte detectar el primer "1" de la entrada.

Si es así vamos a un estado S1. Como debemos detectar todavía los valores 011 para seguir armando el diagrama nos conviene primero suponer que todo se dá en forma correcta, es decir, que aparece lo que estamos buscando. Por lo tanto de S1 pasamos a S2 al detectar un

“0”. Luego de S2 a S3, al detectar un “1” y por último a S4 al detectar el “1” final donde debemos poner a Z en “1” avisando que se detectó la secuencia completa (1011).

Luego hay que seguir con las otras posibilidades. Por ejemplo si en S1 no viene un “0” como se esperaba, sino un “1”, podemos quedarnos en S1 ya que ese nuevo “1” puede ser el primero de la secuencia correcta.

Estando en S2 (habiendo detectado ya 10), si viene un 0, lamentablemente eso rompe la secuencia y debemos volver a S0 a la espera de un primer “1” nuevamente.

Estando en S3 (hasta allí llegamos con 101) si viene un “0” implica que hasta ahí han ingresado los bits 1010 y esos dos últimos pueden ser los dos primeros de la secuencia buscada por lo que debemos volver a S2.

Si estamos en S4 dado que recibimos lo correcto (1011) y viene un 0, empezamos nuevamente. Si recibimos un “1” como éste puede ser un primer “1” de una nueva secuencia, vamos a S1.

Teniendo ya el diagrama, el circuito se diseña como se ha visto anteriormente.

Filmina 40: Se debe diseñar con máquina de estados del tipo Moore, un detector de paridad par de un número de dos bits que entra en forma serie al detector, donde se indican con 2 bits el resultado según se indica en la filmina 40. Aquí se considera que el análisis se hace en forma continua, es decir, que se recibe en forma permanente números de 2bits uno atrás del otro.

Para ello, dado que son 2 bits se tardan dos ciclos de reloj hacer el análisis.

Partiendo la primera vez desde S0, se va a S1 ó S2 según sea el valor del primer bit recibido.

Luego, dependiendo del segundo, se debe concluir si las salidas deben indicar, si hay o no paridad par (si hubo un número par de “unos” encontrados: aquí se considera que si no hay “unos” es también par).

Al necesitar 5 estados, ya hay que usar 3 FFs. Por lo que tenemos 3 estados indefinidos dado que no los usamos.

Una posibilidad es la de mandarlos a S0 si es que se presentan, sin importar que valga la entrada.

Otra forma es la de utilizar FFs con entradas de reset asincrónico y forzar siempre a los mismos desde el inicio que vayan al estado S0 por lo que nunca se deberían presentar S5, S6 ó S7, salvo algún ruido que cambie algo en las entradas de los FFs.

Filmina 41: Un diseño de comparador de magnitud de números en formato serie sin signo, suponiendo que se chequea primero el bit mas significativo.

Aquí se supone que luego de obtener el resultado el circuito se queda “congelado”.

Para realizar de nuevo la operación se debe resetearlo para ir de nuevo al estado S0.

Filmina 42 a 43: Diseño de un detector de la secuencia "101" por Moore pero sin solapamiento, es decir, al detectar 101, se debe volver al comienzo y no suponer que el último "1" pueda ser el principio de una nueva secuencia.

Aquí se usan 4 estados, por lo que tenemos 2 FFs.

Filmina 44 a 45: Diseño por Moore de un comando de un motor con dos pulsadores, uno de arranque "A" y otro de parada "B", cuyas acciones se activan cuando están en "1" uno u otro, pero no simultáneamente.

Filmina 46: Detector de número impar en formato serie de 3 bits en forma cíclica, suponiendo que el bit MSB se presenta primero en el análisis.

Este caso es sencillo ya que sólo nos interesa ver el valor del bit LSB con lo que en 3 ciclos de reloj se tiene el resultado. Se usan aquí, 3 FFs.

Filmina 47 a 50: Se diseña un control de motor paso a paso para alimentar al circuito driver del mismo.

En este caso se debe aplicar a ese driver 2 señales por cada uno de los campos del motor: A y B por un lado, C y D, por el otro.

La secuencia debe ser como se muestra en la filmina 47 para lograr que el motor gire en ambos sentidos de giro.

Tenemos entonces 4 salidas, con una entrada de selección de sentido de giro. La frecuencia del reloj define la velocidad de giro del motor.

Si bien se necesitan sólo 2 FFs para cubrir 4 estados, las salidas son 4 y en este ejemplo hemos decidido emplear tantos FFs como salidas se tengan a fin de que cada salida corresponda a un Q de un FF.

Filmina 51: Diseño de un generador de señales programable por dos líneas de selección A y B, donde se seleccionan 3 ciclos de trabajo diferentes del generador (el cambio en su frecuencia se maneja cambiando el período del reloj) según se muestra en la tabla.

Aquí las entradas de selección son evaluadas en el primer ciclo de reloj de cada nuevo período de la señal (en S0 la primera vez y en S4 para los subsiguientes períodos).

Filmina 52: Basado en el diseño anterior podemos sintetizar un generador PWM cuyo ciclo de trabajo lo determine la salida de un conversor ADC de 2 bits.

Este circuito podría utilizarse como un transmisor de datos donde una señal de algún transductor de presión, temperatura, etc.. es digitalizada en 4 niveles por el conversor analógico-digital, y luego el número convertido es transmitido en forma codificada por modulación de pulsos PWM a otra locación.

NOTA: Dado que el diagrama de estados se vá complicando a medida que se suman mas bits, existen otras formas de síntesis con por ejemplo el diseño ya mostrado en la teoría de "Contadores" filmina nº 27 "Generador de pulsos con ciclo de trabajo programable, donde basta conectar la salida de un conversor ADC a las entradas de datos D3...D0 del contador 74163 y realizar algunas modificaciones para que el conversor esté funcionando en forma sincrónica con el circuito, es decir, al inicio de un nuevo período, tener un nuevo dato ya adquirido.

Filmina 53: Se diseña un circuito comparador de magnitud de dos números sin signo de 3 bits en formato serie con máquina de Moore. Se manda primero los bits MSB. El circuito tiene dos entradas de datos X-Y y dos salidas Z1-Z0.

Dependiendo de si el resultado es $X < Y$ ó $X > Y$ ó $X = Y$, se reflejará en el código de Z1Z0.

Mientras se esté procesando el dato, el circuito lo informa poniendo Z1Z0 en 00.

Para este caso, se utilizan 4 FFs debido a que necesitamos 10 estados.

Filmina 54 a 55: Diseño de un circuito monoestable disparado por flanco ascendente de una entrada Y con máquina de Moore.

El circuito obtenido se ingresa en la modalidad de captura por esquemático (se dibuja el circuito) y se lo simula con el MaxPlus-II de Altera.

NOTA IMPORTANTE: Se hace ex profeso cambiar a Y en distintos momentos de la señal de CLK donde se puede observar que la máquina de estados sólo "analiza" lo que pasa en la entrada un instante antes del flanco sensible de reloj de los FFs. Entre flanco y flanco de disparo el circuito es "ciego", es decir, no es capaz de analizar a la entrada. Por esa razón, para este u otros casos donde la señal o señales de entradas sean asincrónicas con el circuito sincrónico, se debe tener conocimiento de la velocidad de cambio que puedan tener esas entradas a fin de garantizar que el período del CLK sea lo suficientemente pequeño como para poder detectar cualquier cambio en las entradas y así "no perderse" de ningún cambio en los eventos que se pretenden detectar.

Por ejemplo: si la señal que aquí se está detectando proviene de la salida de un relay electro-mecánico, donde las frecuencias esperadas son de no mas de varias decenas de Hz, con tener en la máquina de estados un CLK de 10 veces mas frecuencia ya es suficiente para no perder información.

Filminas 56 a 58: Con lógica programable puede diseñarse máquinas de estado. En este ejemplo empleando el modelo de Mealy para construir un contador sincrónico de 2 bits progresivo-regresivo descrito en VHDL.

Generalmente se trabaja con un diagrama de estados en papel y se describe luego la tabla de estados en VHDL.

Con dos procesos se describe el circuito.

El primero define como evolucionarán los estados en función de la entrada de control "dir".

El segundo define que valores adoptarán las salidas en cada estado dependiendo del valor de "dir", ya que se trata de un modelo de Mealy.

Filminas 59 a 61: El mismo proyecto pero basado en descripción de máquina de estados de Moore.

La diferencia radica en que en el segundo proceso, las salidas adoptan un determinado valor para cada estado independientemente del valor de "dir".

La entrada "dir" sólo tiene injerencia en el estado futuro al cual irá el circuito cuando venga un flanco de reloj.

Filmina 62: desde hace algunos años, con el avance tecnológico en el diseño de circuitos digitales programables por hardware (no microprocesadores ó micorcontroladores) denominados PLD (Programmable Logic Device) se ha diseñado una filosofía para la especificación, diseño y test de tales dispositivos, empleando el denominado Lenguaje de Descripción de Hardware (HDL: Hardware Description Language) en cual permite describir un proceso discreto en forma de sentencias similares a los lenguajes de programación de alto nivel como el "C", donde el resultado para nuestro caso de diseño se traduce en la generación de hardware, es decir, la creación de un circuito dentro de uno de estos dispositivos PLD, a través de la interconexión adecuada de sus estructuras internas a fin de sintetizar el circuito que responda a las condiciones de diseño que le hemos especificado.