

Instituto Tecnológico de Costa Rica

Escuela de Ingeniería Electrónica



Cuestionario Previo 2

Taller de diseño digital

Integrantes:

Arias Ortiz Kendy Raquel

Pérez Jiménez Jorge

Fabián M. Villegas Bonilla

Adrián Parajeles Alvarado

Profesor:

Dr.-Ing. Roberto Molina Robles

03 de abril de 2025

Lab. 1: Lógica Secuencial

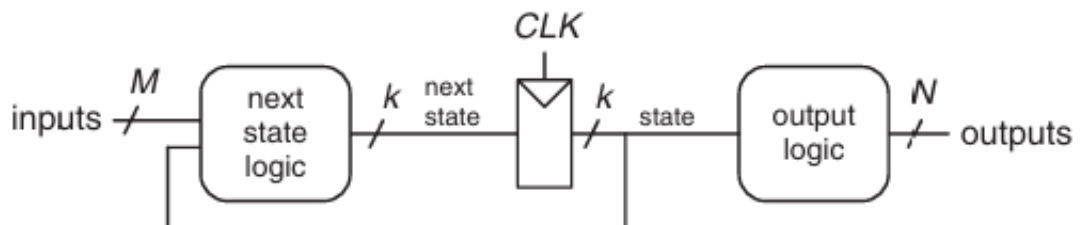
Preguntas

1. Investigue cuál es el funcionamiento de las máquinas de estado finitos. Explique la diferencia entre una máquina de Moore y una de Mealy, y muestre la diferencia por medio de diagramas de estados y señales.

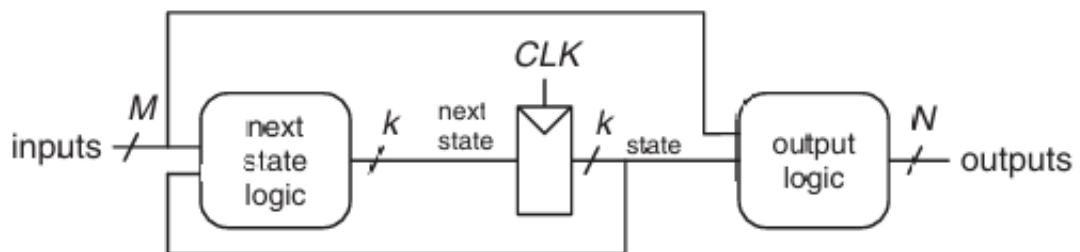
Las máquinas de estado finito (FSM) son circuitos secuenciales síncronos diseñados para realizar una función específica. En general, consisten en dos bloques de lógica combinacional: "next state logic" (lógica de siguiente estado) y "output logic" (lógica de salida), además de un registro que guarda la información de los estados actuales.

Una FSM tiene M entradas, N salidas y k estados (siendo 2^k el número total de combinaciones posibles de estados), un reloj y, opcionalmente, un botón de reset para reiniciar el sistema. Existen dos clases generales de FSM: Moore y Mealy.

- Máquina de Moore: En este tipo de máquina, las salidas dependen únicamente del estado actual. Es decir, las salidas están asociadas a los estados de la máquina, y no cambian directamente con las entradas.
- Máquina de Mealy: En las máquinas de Mealy, las salidas dependen tanto del estado actual como de las entradas del sistema. Esto significa que una entrada puede modificar las salidas incluso si el estado no cambia.



(a)



(b)

Figura 1. Diagramas de FSM: (a) FSM tipo Moore, (b) FSM tipo Mealy.

2. Explique los conceptos de setup time y hold time. ¿Qué importancia tienen en el diseño de sistemas digitales?

En una Máquina de Estado Finito (FSM), la lógica secuencial se implementa utilizando flip-flops, que son los encargados de almacenar los estados de la máquina. El cambio de estado depende de las entradas y el reloj, y es fundamental que los tiempos estén bien ajustados para asegurar que la FSM funcione correctamente. Dos conceptos clave en este contexto son el setup time y el hold time.

- Setup Time: Es el tiempo mínimo que la entrada de datos (que representa el próximo estado) debe permanecer estable antes del borde del reloj (flanco ascendente o descendente) para que el flip-flop registre adecuadamente el nuevo valor. Si la entrada cambia demasiado cerca del borde del reloj, el flip-flop podría no registrar el estado correctamente, lo que podría ocasionar transiciones erróneas en la FSM.
- Hold Time: Es el tiempo mínimo que el valor de la entrada debe mantenerse constante después del borde del reloj para asegurar que el flip-flop retenga correctamente el valor. Si la entrada cambia inmediatamente después del borde, podría generarse un glitch o error, lo que afectaría la estabilidad de la FSM.

3. Explique los conceptos de tiempos de propagación y tiempos de contaminación en circuitos combinacionales. Investigue sobre la ruta crítica y cómo esta afecta en el diseño de sistemas digitales complejos; por ejemplo, un procesador con pipeline. Investigue su relación con la frecuencia máxima de operación de un circuito.

Tiempo de propagación: Es el intervalo que transcurre desde que una señal cambia en la entrada de un circuito combinacional hasta que se refleja en la salida. Representa el retardo máximo que puede tener una señal al atravesar los distintos componentes lógicos.

Tiempo de contaminación: Se refiere al tiempo mínimo en el que una variación en la entrada comienza a influir en la salida, generando posibles valores intermedios o inestables antes de que la señal llegue a establecerse completamente.

Ruta crítica: Dentro de un circuito, la ruta crítica es el trayecto que toma más tiempo en completarse desde una entrada hasta una salida. Este camino define el retardo máximo del sistema, y por tanto, determina el límite superior de la velocidad a la que puede funcionar de manera segura el circuito.

En sistemas digitales complejos como los procesadores con arquitectura *pipeline*, el diseño se divide en múltiples etapas que trabajan de forma paralela. Cada etapa debe estar sincronizada con el reloj del sistema, y la etapa con mayor retardo (su ruta crítica) marca el tiempo mínimo que debe durar un ciclo de reloj. Si la ruta crítica no está bien optimizada, puede ralentizar todo el procesamiento, afectando negativamente el rendimiento general.

Por último, la frecuencia máxima de operación de un circuito está directamente relacionada con el tiempo de la ruta crítica: a mayor retardo, menor puede ser la frecuencia del reloj. Para lograr un rendimiento óptimo, es esencial minimizar los tiempos de propagación dentro de las rutas críticas del diseño.

4. **Investigue sobre las mejores prácticas para la asignación de relojes y división de frecuencia en FPGAs. En este apartado haga énfasis en el uso de las entradas habilitadoras de reloj (clock enables) presentes en las celdas de la FPGA, para lograr tener tiempos de ejecución diferentes a lo largo del sistema mientras se utiliza un solo reloj.**

Los relojes en las FPGAs son una herramienta indispensable para el control de los sistemas digitales. Sin embargo, ciertas características de los relojes pueden provocar errores como delays o bugs relacionados con la temporización. Uno de los principales desafíos ocurre cuando se utilizan múltiples relojes en un mismo diseño, lo que se conoce como Clock Domain Crossing (CDC). Este fenómeno puede generar complicaciones como metaestabilidad, pérdida de datos o errores de sincronización (Harvey, 2025).

Para evitar estos problemas y mejorar el rendimiento de los sistemas implementados en FPGA, es fundamental aplicar buenas prácticas en la planificación y distribución de los relojes. Esto incluye: Comprender los conceptos clave de temporización, sincronización y los riesgos asociados al uso de múltiples dominios de reloj y Planificar cuidadosamente el número de relojes utilizados, procurando reducir la cantidad de dominios para evitar traslapes y facilitar el análisis de tiempos. (Harvey, 2025).

Dos prácticas recomendadas para la asignación de relojes y la división de frecuencia son:

- **Uso de Clock Enable (CE):** Esta técnica permite controlar cuándo se actualiza una parte del sistema sin necesidad de generar un nuevo reloj. Se mantiene un único dominio de reloj y se reduce significativamente el riesgo de errores de temporización y consumo innecesario de energía.
- **Uso de bloques dedicados (como PLLs o MMCMs):** Estos permiten generar nuevas frecuencias o fases de reloj a partir de una fuente principal. Aunque generan nuevos dominios de reloj, son necesarios cuando se requiere una señal de reloj con características específicas .

5. **Investigue sobre el fenómeno de rebotes y ruido en pulsadores e interruptores. Defina qué técnicas digitales y analógicas (circuitos) se utilizan para cancelar este fenómeno. Además, investigue sobre los problemas de metaestabilidad cuando se tienen entradas asíncronas en circuitos digitales. Finalmente, presenta circuitos que permiten la sincronización de entradas como pulsadores e interruptores.**

Rebotes, Ruido y Metaestabilidad en Circuitos Digitales

Rebotes y ruido:

Cuando se acciona un pulsador o interruptor, pueden aparecer oscilaciones rápidas no deseadas llamadas *rebotes*, causadas por la mecánica interna del contacto. Además, el *ruido eléctrico* puede distorsionar la señal por interferencias externas.

Técnicas para eliminar rebotes y ruido:

- Debouncing digital: Usa lógica o software para ignorar cambios rápidos hasta que la señal se estabilice.
- Filtros RC: Combinación de resistencias y capacitores que suavizan la señal.
- Capacitores en paralelo: Ayudan a amortiguar los rebotes.
- Retardos temporales: Espera antes de procesar la señal para asegurar estabilidad.

Metaestabilidad:

Ocurre cuando una señal asincrónica cambia cerca del flanco del reloj, lo que puede dejar al flip-flop en un estado indefinido temporalmente.

Sincronización de señales:

Para evitar metaestabilidad, se usan dos flip-flops tipo D en serie conectados al mismo reloj, permitiendo estabilizar señales asincrónicas antes de usarlas.

6. Investigue sobre el concepto de IP-Core. Revise la documentación relativa al uso de las herramientas de IP-Core en Vivado, en particular sobre el Clocking-Wizard, y los IPs de verificación física: ILA (Integrated Logic Analyzer) y VIO (Virtual Input/Output). Sobre estos IPs, resuma para qué se utilizan, cómo configurarlos y cómo utilizarlos en su proyecto.

Un IP-Core es un bloque funcional reutilizable que implementa una tarea específica dentro de un diseño digital. Estos bloques pueden representar funciones simples como sumadores o contadores, o complejas como controladores de memoria, procesadores o interfaces de comunicación. Los IP-Cores están disponibles en los entornos de desarrollo como Vivado (de Xilinx), y pueden ser incorporados directamente al diseño sin necesidad de escribir todo el código RTL desde cero. Esto acelera el desarrollo y garantiza la reutilización de bloques probados previamente.

En Vivado, el uso de IP-Cores está integrado mediante la herramienta IP Catalog, donde se puede acceder a un amplio conjunto de IPs listos para configurar e insertar en un proyecto. Uno de los IPs más utilizados es el Clocking Wizard, que permite generar señales de reloj a diferentes frecuencias y con distintas características (por ejemplo, fases o relaciones de multiplicación/división). Este IP facilita la gestión de dominios de reloj dentro de un diseño y puede derivar múltiples señales de reloj a partir de una fuente principal, utilizando primitivas internas como MMCMs o PLLs.

En cuanto a los IPs de verificación física, Vivado incluye herramientas como ILA (Integrated Logic Analyzer) y VIO (Virtual Input/Output). El ILA es un analizador lógico integrado que permite observar señales internas del diseño en tiempo real a través del hardware. Se conecta directamente al diseño y captura datos basándose en condiciones de disparo configurables. Esto permite al diseñador verificar el comportamiento del sistema en la FPGA sin tener que sacar señales al exterior, algo especialmente útil cuando no hay suficientes pines o cuando se necesita analizar señales internas que no son accesibles físicamente.

Por otro lado, el VIO permite tanto monitorear cómo controlar señales internas del diseño. A través de esta herramienta, el usuario puede enviar valores desde el entorno de Vivado al hardware, modificando registros o líneas de control sin necesidad de volver a sintetizar el diseño. También permite observar el estado de señales internas en tiempo real. Es útil, por ejemplo, para probar diferentes combinaciones de entradas o activar manualmente ciertos módulos del diseño durante la depuración.

La configuración de estos IPs en Vivado es bastante intuitiva. Al agregar un IP desde el catálogo, se abre un asistente gráfico donde se seleccionan los parámetros necesarios (como número de probes en ILA, ancho de datos en VIO o frecuencia de salida en Clocking Wizard). Una vez configurado, se genera el bloque IP, que puede integrarse al diseño mediante diagramas de bloques (Block Design) o directamente en el RTL instanciando el módulo generado. Luego, al implementar el diseño en la FPGA, las herramientas de Vivado permiten interactuar con ILA y VIO a través de la interfaz de Hardware Manager.

Referencias

1. **Harris, D. M., & Harris, S. L.** *Digital Design and Computer Architecture: ARM Edition* (2nd Edition). Morgan Kaufmann, 2016.
2. Harvie, L. (2025, 21 febrero). *Best Practices for Managing Multiple Clock Domains in FPGA Designs - RunTime Recruitment.* RunTime Recruitment. <https://runtimerec.com/best-practices-for-managing-multiple-clock-domains-in-fpga-designs/>