



**Mi Universidad**

Ensayo

Nombre del alumno: Yahir Aguilar Sicalhua.

Nombre del tema: Unidad I y II.

Parcial: 1.

Nombre de la materia: Diseño Lógico.

Nombre del profesor: Juan José Ojeda Trujillo.

Nombre de la licenciatura: Ingeniería en Sistemas Computacionales.

Cuatrimestre: 6.

Existen dos tipos de circuitos secuenciales:

Síncronos: el circuito solo cambia al cambiar una señal específica de reloj.

Asíncronos: evolucionan a cada cambio de entrada, las realimentaciones son directas y el modelo matemático es complicado y restrictivo (solo cambia una entrada cada vez).

## INTRODUCCIÓN

### 1.1.- CIRCUITOS DIGITALES.

Un circuito integrado es una pieza o cápsula que generalmente es de silicio o de algún otro material semiconductor, que utilizando las propiedades de los semiconductores, es capaz de hacer las funciones realizadas por la unión en un circuito, de varios elementos electrónicos, como: resistencias, condensadores, transistores, etc.

### 1.2.- CIRCUITOS COMBINACIONALES VS. CIRCUITOS SECUENCIALES.

Circuito combinacional. Está formado por funciones lógicas elementales (AND, OR, NAND, NOR, etc.), que tiene un determinado número de entradas y salidas. Es un circuito cuya salida depende solamente de la "combinación" de sus entradas en el momento que se está realizando la medida en la salida.

Los circuitos de lógica combinacional son hechos a partir de las compuertas básicas compuerta AND, compuerta OR, compuerta NOT. También pueden ser construidos con compuertas NAND, compuertas NOR, compuerta XOR, que son una combinación de las tres compuertas básicas.

### 1.3.- SISTEMAS SECUENCIALES SÍNCRONOS.

Un circuito secuencial síncrono utiliza señales que afectan a los elementos de almacenamiento únicamente en instantes discretos. La sincronización se logra con un dispositivo de temporización llamado generador de reloj, el cual produce un tren periódico de pulsos de reloj. Los pulsos de reloj se distribuyen por todo el sistema de modo que los elementos de almacenamiento sólo se vean afectados al llegar al pulso.

### 1.4.- EL LENGUAJE DE DESCRIPCIÓN DE HARDWARE VHDL.

VHDL es un lenguaje de descripción de circuitos electrónicos digitales que utiliza distintos niveles de abstracción. El significado de las siglas VHDL es VHSIC (Very High Speed Integrated Circuits) Hardware Description Language. Esto significa que VHDL permite acelerar el proceso de diseño.

### **1.4.1.- ORIGEN Y UTILIDAD DEL LENGUAJE.**

Los estudios para la creación del lenguaje VHDL (VHSIC HDL) comenzaron en el año 1981, bajo la cobertura de un programa para el desarrollo de Circuitos Integrados de Muy Alta Velocidad (VHSIC), del Departamento de Defensa de los Estados Unidos. En 1983 las compañías Intermetrics, IBM y Texas Instruments obtuvieron la concesión de un proyecto para la realización del lenguaje y de un conjunto de herramientas auxiliares para su aplicación. Finalmente, en el año 1987, el lenguaje VHDL se convierte en la norma IEEE-1076 –como todas las normas IEEE, se somete a revisión periódica, por lo que en 1993 sufrió algunas leves modificaciones.

### **1.4.2.- ELEMENTOS BÁSICOS DEL LENGUAJE: TIPOS DE DATOS Y OPERADORES.**

COMENTARIOS: Los comentarios van precedidos de dos guiones. En una línea se ignorará todo aquello que vaya después de dos guiones seguidos. Ejemplo:

--Esto es un comentario

### **1.4.3.- CONCEPTO Y DEFINICIÓN DE ENTITY.**

El tipo de entidad es el bloque de creación fundamental para describir la estructura de los datos con el Entity Data Model. En un modelo conceptual, los tipos de entidad se construyen a partir de propiedades y describen la estructura de conceptos de nivel superior, como clientes y pedidos en una aplicación empresarial. Del mismo modo que una definición de clase en un programa es una plantilla para las instancias de la clase, un tipo de entidad es una plantilla para las entidades.

### **1.4.4.- CONCEPTO Y DEFINICIÓN DE ARCHITECTURE.**

La Arquitectura la podemos definir como la estructura que describe el funcionamiento de la Entidad, de manera tal que nos permita el desarrollo de los procedimientos que se deberá llevar a cabo con la finalidad de que la Entidad cumpla con las condiciones de funcionamiento que queramos.

### **1.4.5.- SENTENCIAS CONCURRENTES.**

Se ejecutan de forma asíncrona unas respecto de las otras en el mismo tiempo de simulación. El orden en que se escriban es indiferente ya que no siguen un orden de ejecución predefinido. – No obstante, conviene escribir el código en el orden que mejor se pueda entender y mejor documente el programa.

Sirven para especificar:

Interconexiones entre componentes.

Estructuras jerárquicas.

Estructuras regulares.

Transferencias entre registros.

#### **1.4.6.- SENTENCIAS SECUENCIALES: PROCESS.**

Los procesos solo se permiten dentro de una arquitectura. Las declaraciones dentro de los procesos se ejecutan secuencialmente, no simultáneamente.

Los procesos se pueden escribir de varias maneras. El enfoque más común cuando se usan procesos para describir diseños es usar el formulario que tiene una lista de sensibilidad.

## **UNIDAD II. SISTEMAS SECUENCIALES SÍNCRONOS.**

### **2.1.- AUTÓMATAS DE ESTADOS FINITOS: MELAY VS. MOORE.**

En la teoría de la computación, una Máquina de Mealy es un tipo de máquina de estados finitos que genera una salida basándose en su estado actual y una entrada. Esto significa que el Diagrama de estados incluirá ambas señales de entrada y salida para cada línea de transición. En contraste, la salida de una máquina de Moore de estados finitos (el otro tipo) depende solo del estado actual de la máquina, dado que las transiciones no tienen entrada asociada. Sin embargo, para cada Máquina de Mealy hay una máquina de Moore equivalente cuyos estados son la unión de los estados de la máquina de Mealy y el Producto cartesiano de los estados de la máquina de Mealy y el alfabeto de entrada.

### **2.2.- ESPECIFICACIÓN DEL SISTEMA MEDIANTE DIAGRAMAS Y TABLAS DE ESTADOS.**

Diagrama de Estado: Esta muestra la secuencia de estados por los que pasa bien un caso de uso, un objeto a lo largo de su vida, o bien todo el sistema. Es una forma de representación gráfica más intuitiva de los autómatas finitos basadas en dígrafos con arcos acotados llamados transiciones en los cuales se ponen los símbolos de tránsito entre un vértice (estado) y otro y se identifican los estados de partida y los de aceptación del resto.

### **2.3.- REPRESENTACIÓN COMPORTAMENTAL DEL SISTEMA MEDIANTE VHDL.**

El lenguaje de descripción en hardware VHDL se estructura en módulos o unidades funcionales, identificados mediante una palabra reservada y particular de este lenguaje. En tanto, a su vez, cada módulo tiene una secuencia de instrucciones o sentencias, las cuales, en conjunto con las declaraciones de las unidades involucradas en el programa, permiten la descripción, la comprensión, la evaluación y la solución de un sistema digital.

## **2.4.- IMPLEMENTACIÓN ESTRUCTURADA DE SISTEMAS SECUENCIALES SÍNCRONOS.**

El circuito secuencial debe ser capaz de mantener su estado durante algún tiempo, para ello se hace necesario el uso de dispositivos de memoria. Los dispositivos de memoria utilizados en circuitos secuenciales pueden ser tan sencillos como un simple retardador (inclusive, se puede usar el retardo natural asociado a las compuertas lógicas) o tan complejos como un circuito completo de memoria denominado multivibrador biestable o Flip Flop.

### **2.4.1.- CODIFICACIÓN DE ESTADOS: RANDOM, ONE- HOT Y SALIDAS IGUAL A VARIABLES DE ESTADO.**

One Hot: codificación de estados donde hay solamente un uno por código y se utilizan tantos flip-flops como estados existan.

Mínimo cambio de Bit (código Grey): codificación de estados donde solamente hay un cambio de bit entre códigos adyacentes.

Conteo enumerado (binario): codificación de estados que utiliza la numeración binaria como asignación de códigos. Codificación de estados realizada directamente por el diseñador:

Aleatorio (Random): codificación de estados sin orden aparente.

Two Hot: codificación de estados donde hay dos unos por código. También utiliza tantos Flip-Flops como estados existan.

Igual a las salidas: codificación de estados donde el código del estado es igual a la salida de la máquina de estados.

### **2.4.3.- IMPLEMENTACIÓN DE SISTEMAS SECUENCIALES SÍNCRONOS MEDIANTE BIESTABLES Y PUERTAS LÓGICAS.**

Las puertas lógicas y los circuitos creaos con ellas constituyen la lógica combinacional, llamada así porque la salida únicamente depende de la combinación de las variables de entrada que haya.

Existe un segundo grupo de circuitos lógicos denominados secuenciales llamados así porque la salida depende del valor de salida anterior, además de las variables de entrada. Ésto significa que estos circuitos están dotados de memoria. Además, una gran parte de los circuitos secuenciales sólo se activan con una señal cíclica o de reloj, y se denominan circuitos secuenciales síncronos.

## 2.4.4.- RETARDO DE PROPAGACIÓN.

Cuando una señal digital pasa a través de un circuito lógico, siempre experimenta un retardo temporal llamado tiempo de retardo de propagación. Este tiempo es muy importante porque limita la frecuencia máxima a la que es posible trabajar.

## 2.4.5.- FRECUENCIA MÁXIMA.

Periodo Mínimo

En primer lugar, recordemos que el periodo de reloj de un sistema es el que 'marca el paso' en el sistema. Si el periodo es largo, el sistema da pasos (entrega resultados/valores) a pasos grandes (sistema lento), si el periodo del reloj es pequeño, el sistema entregará resultados a pasos más cortos, es decir más rápidamente.

Recordemos también que la frecuencia de funcionamiento de un sistema es la inversa del periodo de ese mismo sistema:  $F = 1/T$  (periodo). Por ello para encontrar la máxima frecuencia debemos encontrar el periodo mínimo . . .

¿Cómo calcular u obtener el periodo mínimo?

Bien, en un sistema digital el periodo mínimo está compuesto por los retardos MÁXIMOS de los componentes sincrónicos, de los componentes combinacionales, y de las interconexiones entre ellos.

## 2.4.6.- INICIALIZACIÓN DEL SISTEMA.

Un aspecto que un administrador de un sistema Linux debe considerar en ocasiones, es la secuencia de inicialización del sistema. Esto puede servir para iniciar junto con el sistema servicios importantes (especialmente si el computador se emplea como servidor), para eliminar servicios que no se usen y así aprovechar mejor los recursos o para afinar algunos detalles que afectarán a todos los usuarios del sistema. Por otra parte, algunas fallas que pueden ocurrir en un sistema Linux pueden deberse a una secuencia de inicialización errada y para solucionarlas puede ser necesario seguirla de cerca.

## 2.4.7.- REPRESENTACIÓN ESTRUCTURAL DEL SISTEMA MEDIANTE VHDL.

VHDL fue diseñado con base a los principios de la programación estructurada. La idea es definir la interfaz de un módulo de hardware mientras deja invisible sus detalles internos. La entidad (ENTITY) en VHDL es simplemente la declaración de las entradas y salidas de un módulo mientras que la arquitectura (ARCHITECTURE) es la descripción detallada de la estructura interna del módulo o de su comportamiento.

El motor síncrono es aquel en el que el rotor y el campo magnético rotante giran a la misma velocidad y de manera sincronizada. El motor asíncrono es aquel en el que el campo magnético siempre va unos grados por delante del rotor, por lo que nunca llegan a girar a la misma velocidad y por lo tanto no están sincronizados.

Fuentes de información o bibliografía:

<http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro16/45-circuitos-de-tiempo.html>

<https://plataformaeducativauds.com.mx/libro.php?idLibro=168452965316>