

MEMORIAS

ALARCON FAJARDO ROCIO BELEN

INGENIERIA ELECTRONCIA

2025

RESUMEN:

En esta práctica se estudió y se implementó el tema de memorias semiconductoras en lenguaje de VHDL, comprendiendo tanto su funcionamiento teórico como su aplicación práctica en un entorno de simulación en la FPGA. El objetivo principal fue comprender como se organizan, controlan y verifican los distintos tipos de memoria, en este caso no centramos en la memoria ROM, RAM y los conjuntos.

Los módulos implementados fueron: una memoria ROM 512x32, una memoria RAM 512x32 con un puerto de lectura y un puerto de escritura con mascara por byte, capaz de actualizar selectivamente cada uno de los cuatro bytes de una palabra. También se realizó la implementación práctica en la placa FPGA de una memoria RAM de 16×4 bits, utilizando interruptores como entradas de control y un display de 7 segmentos como salida visual para mostrar los datos almacenados.

Mediante la realización los bandos de prueba se pudieron verificar los funcionamientos de cada módulo, las operaciones de lectura, escritura, enmascarado y preservación de datos se realizaron correctamente.

INTRUDUCCION:

La Memoria **ROM de Máscara** (*Mask Read-Only Memory*) es un tipo de memoria de solo lectura no volátil cuyo contenido de datos binarios se establece de forma permanente durante el proceso de fabricación del circuito integrado. Este proceso difiere de otras tecnologías de memoria en que la programación no ocurre después de la manufactura, sino que está integrada en la estructura física del chip (Harris & Harris, 2022).

El nombre hace alusión directa a la fotomáscara utilizada en la fotolitografía. El diseño de esta máscara define si una celda de memoria particular (que típicamente consiste en un transistor o diodo) está conectada o no, estableciendo de manera inalterable si almacena un valor binario “0” o un “1”.

Debido a que los datos están grabados en la configuración física del silicio, la información no puede ser modificada, borrada ni reescrita una vez que el proceso de fabricación ha finalizado.

La memoria **PROM** (*Programmable Read Only Memory*) es un tipo de memoria semiconductor no volátil que permite al usuario programar su contenido una sola vez después de la fabricación. A diferencia de la ROM de máscara, cuyo contenido se graba de manera permanente en fábrica, la PROM se entrega vacía y puede ser programada por el diseñador mediante un dispositivo especial denominado programador de memorias o PROM programmer (Floyd, 2015).

El principio de funcionamiento de la PROM se basa en una matriz de fusibles o enlaces programables. Cada celda de memoria está compuesta por un transistor o diodo conectado a través de un fusible. Durante la programación, se aplica un pulso de corriente o tensión elevado que “quema” o abre los fusibles correspondientes a los bits que deben almacenar un “0”, mientras que los fusibles intactos representan un “1”. Este proceso modifica de manera irreversible el estado físico de las celdas, por lo que el contenido no puede borrarse ni reescribirse.

El acceso a la información, al igual que en otras memorias ROM, es aleatorio y rápido, permitiendo leer cualquier dirección de memoria sin necesidad de recorrer otras posiciones. Este tipo de memoria se emplea cuando el contenido debe fijarse definitivamente después de la etapa de desarrollo, como en sistemas embebidos, controladores industriales o equipos electrónicos de producción masiva, donde el firmware no requiere modificaciones posteriores (Dally, Harting & Aamodt, 2016).

La PROM representa un avance sobre la ROM tradicional al brindar flexibilidad en la programación inicial, pero su principal limitación radica en la imposibilidad de modificar los datos una vez grabados. Por ello, fue progresivamente reemplazada por tecnologías reprogramables como las EPROM y EEPROM, que permiten borrar y actualizar la información eléctricamente. (Floyd, 2015; Dally, Harting & Aamodt, 2016)

La memoria **EPROM** (*Erasable Programmable Read Only Memory*) es un tipo de memoria semiconductor no volátil y reprogramable que permite borrar y volver a grabar su contenido mediante un procedimiento físico basado en la exposición a luz ultravioleta. Este tipo de memoria fue desarrollada como una evolución de la PROM, superando su principal limitación la imposibilidad de modificar los datos una vez programados (Floyd, 2015).

Su estructura interna está compuesta por una matriz de transistores MOS con puerta flotante, la cual puede almacenar carga eléctrica durante largos períodos de tiempo. Para programar la EPROM, se aplican altos niveles de tensión a los terminales de control, provocando que los electrones queden atrapados en la puerta flotante. Estos electrones modifican el umbral de conducción del transistor, estableciendo valores.

El borrado del contenido se realiza exponiendo el chip a una radiación ultravioleta intensa a través de una pequeña ventana de cuarzo en su encapsulado. La energía de la luz libera las cargas atrapadas, restaurando los transistores a su estado original. Este proceso suele requerir entre 15 y 30 minutos, tras lo cual la memoria queda lista para ser reprogramada con nueva información (Dally, Harting & Aamodt, 2016).

La memoria **EEPROM** (*Electrically Erasable Programmable Read Only Memory*) es un tipo de memoria semiconductor no volátil y eléctricamente reprogramable, que permite borrar y escribir datos de forma selectiva mediante la aplicación de impulsos eléctricos. Representa una mejora significativa respecto a la EPROM, ya que elimina la necesidad de utilizar radiación ultravioleta y permite modificar el contenido directamente dentro del circuito, sin extraer el dispositivo del sistema (Floyd, 2015).

Su principio de funcionamiento se basa en celdas de transistores MOS con puerta flotante, similares a las de la EPROM, pero con una estructura que posibilita la inyección y eliminación de carga eléctrica por efecto de túnel de Fowler Nordheim. Esta técnica permite borrar y escribir bits individuales o bloques pequeños aplicando tensiones controladas sobre los terminales de compuerta y drenaje, lo que brinda una gran flexibilidad y precisión en la actualización de datos (Dally, Harting & Aamodt, 2016).

Una de las principales ventajas de la EEPROM es su capacidad para reprogramarse en el propio circuito, característica que la hace ideal para sistemas embebidos, microcontroladores y dispositivos que requieren ajustes o calibraciones posteriores a la fabricación, como la configuración de parámetros, tablas de control o almacenamiento de datos de usuario. Además, mantiene su contenido sin necesidad de alimentación eléctrica, ofreciendo alta fiabilidad y durabilidad en el tiempo (Harris & Harris, 2022).

Sin embargo, la EEPROM presenta limitaciones en la velocidad de escritura y el número de ciclos de reprogramación, generalmente del orden de 10^4 a 10^6 ciclos, lo cual restringe su uso en aplicaciones que demandan escritura intensiva.

(Floyd, 2015; Dally, Harting & Aamodt, 2016; Harris & Harris, 2022)

La **memoria Flash** es una evolución de la tecnología EEPROM, diseñada para permitir el borrado y escritura de datos en bloques completos en lugar de hacerlo bit a bit, lo que reduce significativamente los tiempos de operación y aumenta la densidad de almacenamiento. Al igual que las demás memorias no volátiles, conserva su contenido aun cuando se interrumpe la alimentación eléctrica, siendo actualmente la más utilizada en dispositivos portátiles, sistemas embebidos y unidades de almacenamiento masivo (Harris & Harris, 2022).

Para borrar la memoria, se aplica un voltaje inverso que libera los electrones almacenados. A diferencia de la EEPROM tradicional, el proceso de borrado no se realiza sobre cada celda individual, sino sobre bloques o páginas completas, lo que optimiza el tiempo de acceso y la eficiencia energética (Dally, Harting & Aamodt, 2016).

La memoria **SRAM** (*Static Random Access Memory*) es un tipo de memoria semiconductor volátil que permite el acceso aleatorio tanto para lectura como para escritura, y cuyo contenido se conserva mientras el dispositivo permanece alimentado. A diferencia de la DRAM, no requiere ciclos periódicos de refresco, lo que le otorga mayor velocidad de acceso y una estructura de control más simple (Harris & Harris, 2022).

Cada celda de una memoria SRAM está formada típicamente por seis transistores MOS (6T), organizados en pares que actúan como un biestable tipo flip-flop encargado de almacenar un bit de información. Dos transistores adicionales se emplean como interruptores de acceso, conectando o desconectando la celda de las líneas de datos (bit lines) durante las operaciones de lectura o escritura. Cuando se aplica una señal a la línea de selección de palabra (word line), las celdas de la fila correspondiente quedan habilitadas, permitiendo que la información almacenada se lea o se reemplace (Dally, Harting & Aamodt, 2016).

El funcionamiento de la SRAM se caracteriza por su bajo tiempo de acceso (del orden de pocos nanosegundos), lo que la convierte en la opción ideal para memorias caché, registros internos de procesadores y buffers de alta velocidad. Sin embargo, su estructura compleja ocupa un área considerable dentro del chip, por lo que su densidad es menor y su costo por bit mayor que en las memorias DRAM (Floyd, 2015).

La memoria **DRAM** (*Dynamic Random Access Memory*) es una memoria semiconductor volátil de lectura y escritura aleatoria, utilizada como memoria principal en la mayoría de los sistemas computacionales. Su principal característica es que almacena cada bit de información en un capacitor asociado a un transistor MOS, lo que permite una alta densidad de integración y un bajo costo por bit en comparación con la SRAM (Harris & Harris, 2022).

Cada celda de DRAM consiste en un transistor de acceso y un condensador (1T-1C). Cuando el transistor se activa mediante la señal de selección de palabra, el condensador puede cargarse o descargarse a través de las líneas de bit, representando así los estados lógicos “1” (condensador cargado) y “0” (condensador descargado). Sin embargo, debido a que los condensadores pierden gradualmente su carga eléctrica con el tiempo, el contenido de la memoria debe refrescarse periódicamente mediante ciclos de regeneración automáticos, de allí su denominación “dinámica” (Dally, Harting & Aamodt, 2016).

El proceso de refresco se realiza cientos o miles de veces por segundo, dependiendo de la tecnología y la temperatura de operación, lo que introduce cierta complejidad en los circuitos de control. Pese a esto, la DRAM ofrece una densidad muy superior a la de la SRAM, permitiendo construir memorias de gran capacidad con un costo reducido por bit (Floyd, 2015).

Los **conjuntos de registros** (*o register files*) constituyen el nivel más alto de velocidad dentro de la jerarquía de memoria de un sistema digital. Se componen de un conjunto finito de registros de almacenamiento rápido, implementados mediante flip-flops o memorias SRAM de baja capacidad, que permiten guardar temporalmente datos o direcciones utilizados directamente por la unidad central de procesamiento (CPU) o por bloques lógicos dentro de un sistema digital (Harris & Harris, 2022).

Cada registro es capaz de almacenar una palabra binaria completa (por ejemplo, 8, 16, 32 o 64 bits) y cuenta con circuitos de control de lectura y escritura que permiten acceder a su contenido en uno o dos ciclos de reloj. Los conjuntos de registros están organizados como una memoria de acceso aleatorio estática, pero con un número de posiciones limitado, generalmente entre 8 y 64 registros en microcontroladores o entre 32 y 256 en procesadores de propósito general (Dally, Harting & Aamodt, 2016).

La **memoria caché** es un tipo de memoria semiconductor de alta velocidad, ubicada entre los conjuntos de registros del procesador y la memoria principal (DRAM), cuya función es reducir el tiempo promedio de acceso a los datos y las instrucciones más utilizados. Actúa como un intermediario inteligente que almacena temporalmente copias de bloques de memoria recientemente accedidos, aprovechando los principios de localidad temporal y espacial en los programas (Harris & Harris, 2022).

La **memoria principal** es una memoria semiconductor volátil que almacena temporalmente los datos y programas que el procesador necesita mientras se ejecutan las instrucciones. Su función es actuar como un espacio de trabajo inmediato entre la memoria caché y los dispositivos de almacenamiento masivo, permitiendo que la CPU acceda rápidamente a la información requerida en cada ciclo de procesamiento (Harris & Harris, 2022).

La **memoria de almacenamiento**, también denominada memoria secundaria o masiva, es el nivel de la jerarquía de memoria encargado de conservar de forma permanente la información. A diferencia de la memoria principal, es no volátil, es decir, mantiene los datos incluso cuando el equipo se apaga (Harris & Harris, 2022).

Está implementada con tecnologías magnéticas, ópticas o de estado sólido, entre las que se destacan los discos duros (HDD), las unidades de estado sólido (SSD) y las memorias Flash. Estos dispositivos ofrecen gran capacidad de almacenamiento, medida en gigabytes o terabytes, a un costo por bit muy bajo, aunque con tiempos de acceso considerablemente mayores que los de las memorias RAM (Floyd, 2015).

MATERIALES Y METODOS:

Definición de entradas (inputs):

- Clk: señal de reloj
- Addr_x: direcciones de acceso.
- We_x (write enable): habilita la escritura en el flanco ascendente de reloj.
- Mask_w (write mask): permite seleccionar que byte de la palabra se actualiza.
- Din_w: datos a escribir, representados por vectores de 32 bits.

Definición de las salidas (outputs):

- Dout_x: salida de datos, representados por vectores de 32 bits.

Declaración del tipo de arreglo:

- Un arreglo de 512 posiciones x 32 bits para la memoria ROM y RAM.
- Un arreglo de 32 posiciones x 32 bits para el conjunto de registros.

Inicialización del arreglo:

- Mediante una función impura para cargar los datos iniciales.

Descripción del proceso de lectura.

Descripción de proceso de escritura.

Verificación mediante los bancos de prueba.

RESULTADOS:

- Memoria ROM 512 x 32:

Este archivo representa una memoria de 512 posiciones por 32 bits, su función es solo de lectura. Su contenido se carga al inicio mediante un archivo de texto y permanece inalterable durante toda la simulación. En las pruebas, la salida entregó siempre los valores correspondientes a las direcciones seleccionadas, sin cambios ni errores, lo cual confirmó la correcta inicialización y estabilidad del módulo.

- Memoria RAM 512 x 32:

Este archivo describe una memoria de 512 posiciones por 32 bits que permite tanto la lectura como la escritura. Su estructura interna se basa en un arreglo de palabras y utiliza una función impura para inicializar el contenido desde un archivo externo en formato hexadecimal. Durante las pruebas se comprobó que la lectura era asíncrona, ya que el valor de salida cambiaba de forma inmediata al modificar la dirección, mientras que la escritura se efectuaba solo en el flanco ascendente del reloj cuando la señal de habilitación estaba activa. También se verificó el correcto funcionamiento de la máscara de escritura, que permitió modificar solo los bytes seleccionados dentro de cada palabra. En la simulación se pudo comprobar que los datos cargados desde el archivo de inicialización se mostraban correctamente en las direcciones indicadas, demostrando un comportamiento estable del módulo.

- Conjunto de registros 32 x 32:

Este archivo implementa un conjunto de 32 registros de 32 bits cada uno. Este módulo también se inicializa mediante archivo externo y cuenta con dos puertos de lectura y uno de escritura. En la simulación se observó que las lecturas eran combinacionales, es decir, que las salidas respondían instantáneamente al cambio de dirección, mientras que la escritura se realizaba de forma sincronizada con el reloj. Además, se confirmó que el registro número cero permanecía siempre en valor nulo, aun cuando se intentaba escribir sobre él. Este detalle reproduce el comportamiento de los bancos de registros usados en procesadores, donde uno de los registros suele estar fijado a cero para operaciones lógicas o de referencia.

- Implementación de memoria en la placa EDU-CIAA-FPGA

Para esta etapa se procedió a describir, sintetizar e implementar en la placa FPGA un sistema basado en una memoria RAM de 16×4 bits, con el agregado de interruptores como entradas de control y un display de 7 segmentos como salida visual. El objetivo principal fue comprobar el funcionamiento real de la memoria ram_16x4 y su respuesta frente a señales externas de entrada y salida. Los interruptores (x) de la placa se configuraron de la siguiente manera: los primeros cuatro, x(3 downto 0), se utilizaron como entradas de dirección y, simultáneamente, como entradas de datos a cargar; el interruptor x(5) se asignó como habilitación de escritura, permitiendo almacenar el valor presente en la entrada; el x(6) se utilizó como habilitación de dirección, controlando cuándo la dirección seleccionada debía ser interpretada como válida; y el x(7) se destinó a la visualización, mostrando alternativamente la dirección o el dato cargado en esa posición de memoria.

El resultado leído de la memoria, correspondiente a los 4 bits almacenados, se conectó al display de 7 segmentos, de modo que cada combinación binaria se representara como un número hexadecimal (del 0 al F). Durante las pruebas en laboratorio, el comportamiento del sistema fue el esperado: al accionar los interruptores se pudo seleccionar una dirección, cargar un valor en la memoria y visualizarlo correctamente en el display. Al modificar el interruptor x(7), se observó el cambio inmediato entre la visualización de la dirección (x(7) = '1') y del dato

almacenado ($x(7) = '0'$), confirmando así el funcionamiento correcto de la memoria y de la lógica de control implementada.

CONCLUSIONES:

La descripción de hardware es la forma de representar el funcionamiento de un circuito digital mediante un lenguaje que permite modelar su estructura y comportamiento, como en este caso se hizo con VHDL. A diferencia de la programación tradicional, donde se escriben instrucciones que se ejecutan de forma secuencial, la descripción de hardware define cómo se interconectan y operan los componentes físicos (puertas lógicas, registros, memorias, etc.) dentro de un dispositivo.

Describir hardware implica pensar en paralelismo, temporización y señales eléctricas, más que en algoritmos. A través de este enfoque es posible sintetizar el diseño y llevarlo a un circuito real dentro de una FPGA. En este trabajo, por ejemplo, la descripción en VHDL permitió implementar distintas memorias (ROM, RAM y registros) y comprobar su funcionamiento tanto en simulación como directamente en la placa, entendiendo cómo las líneas de código se traducen en comportamiento físico y lógico del hardware.

REFERENCIAS:

Sarah L. Harris y Daavid Harris (2022). *Digital Design and Computer Architecture, RISC-V Edition*. Morgan Kaufmann.

Thomas F. Floyd (2015). *Digital Fundamentals* (11th ed.). Pearson.

Dally, W. J., Harting, R., & Aamodt, J. (2016). *Digital Design Using VHDL: A Systems Approach*. Cambridge University Press.