

DISEÑO DE UNA MÁQUINA DE ESTADOS PARA EL CONTROL DE UNA  
INTERFAZ DE LA MEMORIA LPDDR3

JULIÁN DAVID GAMBOA GARCÍA

UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS  
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE  
TELECOMUNICACIONES  
BUCARAMANGA  
2017

DISEÑO DE UNA MÁQUINA DE ESTADOS PARA EL CONTROL DE UNA  
INTERFAZ DE LA MEMORIA LPDDR3

Autor:  
JULIÁN DAVID GAMBOA GARCÍA

Trabajo de grado para optar al título de Ingeniero Electrónico

Director:  
Ckristian Ricardo Esteban Durán Blanco  
Ingeniero Electrónico con Maestría en Ingeniería de Telecomunicaciones

Co-Director:  
Elkim Felipe Roa Fuentes  
Doctorado en Ingeniería Eléctrica y Ciencias de la Computación

UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS  
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE  
TELECOMUNICACIONES  
BUCARAMANGA  
2017

# Contenido

	<b>Pag.</b>
INTRODUCCIÓN	11
1 MEMORIA LPDDR3	12
2 CONTROLADOR DE LA MEMORIA LPDDR3	16
2.1. INICIALIZACIÓN	20
2.2. ESTADO IDLE	22
2.3. ESTADO ACTIVO	24
3 RESULTADOS	26
4 CONCLUSIONES	34
REFERENCIAS	35
BIBLIOGRAFÍA	36

# LISTA DE FIGURAS

	<b>Pag.</b>	
Figura 1	Conexión del controlador a la memoria LPDDR3	16
Figura 2	Ingreso de funciones a la LPDDR3	16
Figura 3	Circuito digital para ingreso de funciones a la LPDDR3	17
Figura 4	Entradas y salidas de la máquina de estados	18
Figura 5	Contador de flancos de reloj	18
Figura 6	Inicialización del protocolo de la memoria LPDDR3	21
Figura 7	Estados posible a acceder desde el estado Idle	23
Figura 8	Estados posible a acceder después del ingreso del estado activo	24
Figura 9	Inicialización de la memoria LPDDR3, primera parte	26
Figura 10	Comienzo del entrenamiento del bus CA	28
Figura 11	Final del entrenamiento del bus CA	29
Figura 12	Configuración de la memoria LPDDR3	29
Figura 13	Escritura de una ráfaga de datos en la memoria LPDDR3	30

Figura 14	Lectura de una ráfaga de datos en la memoria LPDDR3	31
Figura 15	Función de baja potencia	31
Figura 16	Función de apagado interno de la memoria	32
Figura 17	Función de Refresco	32
Figura 18	Función de Auto-Refresco	33

# LISTA DE TABLAS

	<b>Pag.</b>
Tabla 1	Direccionamiento de la memoria LPDDR3 13
Tabla 2	Pines de la memoria LPDDR3 definición y descripción. 14
Tabla 3	Señales del controlador 19
Tabla 4	Señales de comando del controlador 19
Tabla 5	Información que entrega la LPDDR3 después de insertar el MRR0 27
Tabla 6	Envío de datos del bus CA al bus DQ, MRW 41 28
Tabla 7	Envío de datos del bus CA al bus DQ, MRW 48 28

## RESUMEN

**TITULO:** Diseño de una máquina de estados para el control de una interfaz de la memoria LPDDR3<sup>1</sup>.

**PALABRAS CLAVE:** Lpddr3, Diagrama de estados, restablecer, Activo, Escritura, Lectura, Precarga y Máquina de estados.

**AUTOR:** Julián David Gamboa García<sup>2</sup>.

### DESCRIPCIÓN:

Este documento presenta una implementación del controlador compatible con el protocolo de una memoria LPDDR3 usando máquinas de estados. El controlador diseñado se implementó para la tercera generación de memorias de bajo consumo de potencia con el doble de la tasa de transferencia de datos. El controlador está diseñado para usarse en aplicaciones que requieren un alto rendimiento de acceso a la memoria, altas velocidades de transmisión y aplicaciones de bajo consumo energético.

Introduciendo el funcionamiento y utilización de los diferentes tipos de memorias RAM y su arquitectura general. Se realizaron pruebas con el archivo de simulación que provee el fabricante de las memorias RAM, este archivo se utiliza la Memoria LPDDR3 viene con configuraciones con 8 bancos de memoria, este controlador es capaz de soportar capacidades de 4, 6 u 8 GB. Las funciones que se describen en este documento son: la activación de un banco, escritura y lectura en un banco, modo de bajo consumo, registro de lectura y escritura, precarga. y refrescar la memoria.

Realizando una analogía de la memoria LPDDR3 con un bloc de notas, sin el controlar el usuario tendría que revisar que el bloc de notas este limpio para comenzar a escribir, Nuestro controlador realiza esta función en la etapa de inicialización de la memoria, una vez terminada la etapa este entra en espera y el usuario recibe la orden para realizar cualquier función.

---

<sup>1</sup>Trabajo de grado modalidad de investigación

<sup>2</sup>Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones. Director: MsC. Ckristian Ricardo Esteban Durán Blanco. Codirector: Ph.D. Elkim Felipe Roa Fuentes

## ABSTRACT

**TITLE:** Design of a state machine for the control of a LPDDR3 memory interface<sup>1</sup>.

**KEYWORDS:** LPDDR3, Reset, Active, Write, Read, Precharge, Finite state machine.

**AUTHOR:** Julián David Gamboa García<sup>2</sup>.

### DESCRIPTION:

This document presents an implementation of the driver compatible with the protocol of an LPDDR3 memory using state machines. The controller has been implemented for the third generation of low-power memory with double data transfer rate 3. The driver is designed for use in applications that require high performance memory access, high transmission speeds and low power applications.

Introducing the operation and use of the various types of RAM and its general architecture. It was tested with the simulation file that test the manufacturer of the RAM, this file is used the Memory LPDDR3 comes with configurations with 8 memory banks, this controller can support capacities of 4, 6 or 8 GB. The functions described in this document are: activation of a bank, writing and reading in a bank, power down, deep power down reading and writing record, preload, mode register read and write, refresh and self refreshing.

By making an LPDDR3 memory analogy with a block of notes without control the user checking that the block of notes is clean to start writing in the bloc, our control device perform this stage in the initialization, Once is finish the initialization the controller enter in stage of standby and the user receives the order to perform any function.

---

<sup>1</sup>Bachelor Thesis

<sup>2</sup>Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones. Director: MSc. Ckristian Ricardo Esteban Durán Blanco. Codirector:Ph.D. Elkim Felipe Roa Fuentes

# INTRODUCCIÓN

En los últimos años se ha venido experimentando un fenómeno llamado Internet de las Cosas (IoT por sus siglas en inglés), el cual refleja una tendencia a la interconexión de dispositivos entre sí para enviar y recibir información. Los usuarios cada vez generan una mayor cantidad de datos que deben ser procesados de manera eficiente [9]. Por esta razón, la memoria dinámica de acceso aleatorio síncrona –SDRAM–, y en especial las de bajo consumo de energía como la LPDDR3, es usada en tecnologías móviles como smartphones, tablets, ultra-books y computadores portátiles por sus funciones de baja potencia y alta transferencia de datos (1600 Mhz, el doble de su frecuencia de reloj) [6].

Normalmente las memorias SDRAM son utilizadas con el propósito de procesar las instrucciones de un microprocesador a grandes velocidades, y poder ejecutar un programa en el menor tiempo posible [10]. Para el uso de la LPDDR3 junto al microprocesador es necesario tener un controlador que pueda crear un puente de comunicación entre el núcleo de procesamiento y la memoria SDRAM. Dentro de sus principales tareas están: proveer el proceso de inicialización de la memoria, el correcto control de las señales, el manejo simplificado de las funciones, mantener los tiempos entre comandos para ajustarse con el protocolo de la LPDDR3 y también tiene la lógica necesaria para leer, escribir y acceder a las funciones de bajo consumo de la LPDDR3. De esta manera, al tener el controlador incorporado al microprocesador es posible disminuir la latencia de todo el sistema y reducir su consumo de energía [7], siendo esto clave para aplicaciones IoT.

En adición, el controlador presentado en este artículo pretende ser implementado en una futura versión del microcontrolador de código abierto (hardwre abierto) [8], el cual fue desarrollado en la Universidad Industrial de Santander por el Grupo de Investigación en Sistemas Integrados, ONCHIP. Esto significa que todas las personas pueden ver, estudiar o modificar los diseños del microprocesador, incluyendo el modulo diseñado [3].

Finalmente, para poder validar resultados de simulación de la máquina de estados, se contó con un modelo de simulación de la LPDDR3. Éste fue proporcionado por la compañía Micron, la cual es productora de este tipo de memorias, y facilita una descripción en Verilog de uno de sus módulos. Por lo tanto, todo el controlador fue implementado usando el mismo lenguaje, apoyado en su dominio actual (tiene cerca del 60 % del mercado mundial del diseño digital) y en el hecho de que su sintaxis es cercana a la del lenguaje C [2].

# 1. MEMORIA LPDDR3

La memoria de acceso aleatorio (por sus siglas en inglés, RAM), utilizada en el microcontrolador está destinada al almacenamiento de información temporal que será utilizada por el procesador para realizar cálculos u otro tipo de operaciones lógicas [10]. En el espacio de direcciones de memoria RAM se ubican además los registros de trabajo del procesador y los de configuración, trabajo de los distintos periféricos del microcontrolador. Se denomina de "*acceso aleatorio*" porque se puede leer o escribir en una posición de memoria con un tiempo de espera igual para cualquier posición, no siendo necesario seguir un orden para acceder a la información de la manera más rápida posible.

Existen dos variedades de RAM que se puede utilizar en un microcontrolador, estática y dinámica. Una RAM estática (SRAM, static RAM) se construye internamente empleando circuitos similares a los flip-flop D básicos. Estas memorias tienen la propiedad de que su contenido se conserva en tanto se siga alimentando el circuito [11].

Las RAM dinámicas (DRAM), en cambio, no usan flip-flops. En vez de ello, una DRAM es una matriz de celdas, cada una de las cuales sólo necesita un transistor y un diminuto condensador por bit (en comparación con seis transistores por bit que maneja la SRAM) [10]. Los condensadores pueden cargarse o descargarse, lo que permite almacenar ceros y unos. Puesto que la carga eléctrica tiende a fugarse, cada bit de una DRAM debe refrescarse (cargarse de nuevo) cada pocos milisegundos para evitar que los datos se pierdan. Puesto que el proceso de refresco corre a cargo de la lógica externa, las DRAM requieren interfaces más complejas que las estáticas para ello se construye el controlador de la memoria, las DRAM tienen una densidad muy alta (muchos bits por chip). Por esta razón, las memorias principales casi siempre se construyen con DRAM. Sin embargo, esta gran capacidad tiene un precio: las DRAM son lentas (por decenas de nanosegundos).

La combinación de una caché de SRAM y una memoria principal de DRAM intenta combinar las propiedades ventajosas de cada una, por lo tanto viene la memoria SDRAM (DRAM sincrónica) esta memoria es un híbrido de RAM estática y dinámica, esta memoria es controlado por un solo reloj sincrónico. Con la idea de aumentar la velocidad de transferencia de la memoria se ideó crear una con el doble tasa de transferencia (DDR, Double Data Rate), este tipo de memorias reciben y transmiten datos en ambos flancos del reloj, tanto en el flanco positivo como en el flanco negativo. Para poder transferir datos al doble de la frecuencia de reloj.

El funcionamiento de una memoria RAM no es complicada haciendo alusión a que la memoria RAM y referirnos a ella como el bloc de notas del procesador, es porque ésta funciona como apoyo de los registros del procesador. Cualquier programa en ejecución está alojado en memoria; las instrucciones van siendo pasadas a los registros para su ejecución de forma secuencial, y los datos son pasados también a los registros para su manipulación.

Tabla 1: Direccionamiento de la memoria LPDDR3

Componentes		1Gb	2Gb	4Gb	6Gb	8Gb	12Gb	16Gb
Número de bancos		8	8	8	8	8	8	8
Número de bits para dirección del banco		3	3	3	3	3	3	3
Tiempo para refrescar todos los banco en $\mu s$		7.8	3.9	3.9	3.9	3.9	3.9	3.9
x16	Número de bits para la fila	13	14	14	15	15	15	15
	Número de bits para la columna	10	10	11	11	11	12	12
x32	Número de bits para la fila	13	14	14	15	15	15	15
	Número de bits para la columna	9	9	10	10	10	11	11

Fuente: [5] Adaptado de JEDEC, Low Power Double Data Rate 3. JEDEC SOLID STATE TECHNOLOGY ASSOCIATION, Aug. 2015.

La memoria de acceso aleatorio consta de cientos de miles de pequeños capacitadores que almacenan cargas. Al cargarse, el estado lógico del capacitador es igual a 1; en el caso contrario, es igual a 0, lo que implica que cada capacitador representa un bit de memoria. Teniendo en cuenta que se descargan, los capacitadores deben cargarse constantemente (el término exacto es refrescar) a intervalos regulares, lo que se denomina ciclo de refresco .

Cada capacitador está acoplado a un transistor (tipo MOS), lo cual posibilita la recuperación o modificación del estado del capacitador. Estos transistores están dispuestos en forma de tabla (matriz), de modo que se accede a la caja de memoria (también llamada punto de memoria) mediante una línea y una columna.

Los microcontroladores no solo necesitan que la memoria sea de alta velocidad, también es necesario que sea de bajo consumo de energía (esto se introduce para los consumidores de tecnologías móviles). Como en las tecnologías móviles llevan una batería es necesario tener presente el consumo energético del componente. Por lo tanto se introduce la LPDDR la cual tiene funciones para el ahorro energético y también está diseñadas para bajo consumo energético.

La memoria LPDDR3 está configurada internamente con 8 bancos de memoria, cada banco tiene un número determinado de filas y columnas dependiendo de la capacidad de almacenamiento de la memoria LPDDR3 como se muestra en la tabla 1. Estos dispositivos pueden guardar en cada posición entre 16 y 32 bits (dependiendo de la memoria) [1] con una velocidad de reloj de 800 MHz, esto significa que trabaja a 1600 MHz.

Tabla 2: Pines de la memoria LPDDR3 definición y descripción.

Pines de la memoria LPDDR3		
Nombre	Tipo	Descripción
CK_t, CK_c	Entrada	<b>Reloj</b> (por sus siglas en inglés, CK): Reloj diferencial de entrada.
CKE	Entrada	<b>Habilitador del reloj</b> (por sus siglas en inglés, CKE) El habilitador del reloj activa o desactiva la entrada de la señal del reloj y los buffers.
CS_n	Entrada	<b>Selector de Chip</b> (por sus siglas en inglés, CS): CS_n es considerada parte del código de comando, CS-n es mostrada en el flanco positivo del reloj.
CA0-CA9	Entrada	<b>DDR entradas de comandos/dirección</b> (por sus siglas en inglés, CA) bus de entrada unidireccional de dirección y comandos. Es considerado parte del código de comandos
DQ0-DQ31	Entrada o salida	<b>Datos bidireccionales de entrada y salida</b> (por sus siglas en inglés, DQ): Bus Bi-direccional de datos.
DQS0_t- DQS3_t, DQS0_n- DQS3_n	Entrada o salida	<b>Señal de datos:</b> Es un bus bi-direccional (usado para muestrear la señal DQ cuando se leen y escriben datos) y diferencial (DQS_t y DQS_c). Este bus se utiliza de salida como lectura de datos y entrada como escritura de datos.
DM0-DM3	Entrada	<b>Enmascaramiento de datos:</b> (Por sus siglas en inglés, DM) Señal que enmascara los datos de escritura. DM es muestreada en ambos flancos de de DQS_t.
ODT	Entrada	<b>Terminación del chip:</b> (por sus siglas en inglés, ODT) Esta señal habilita y deshabilita la terminación en el bus DQ de la memoria LPDDR3, acorde al específico modo de las propiedades de registro

Fuente: [5] Adaptado de JEDEC, Low Power Double Data Rate 3. JEDEC SOLID STATE TECHNOLOGY ASSOCIATION, Aug. 2015.

La memoria LPDDR3 está estandarizada por el Consejo Conjunto de Ingeniería de Dispositivos Electrónicos (JEDEC, Joint Electron Device Engineering Council). JEDEC publicó un libro donde estandarizó la memoria [5]. En el estándar explica los pines con

las que trabaja la memoria LPDDR3, en la tabla 2 se hace un resumen, con estos pines se manejan las funciones del protocolo.

El protocolo de la memoria LPDDR3 tiene diferentes modos de operación con los que funciona la memoria LPDDR3. El estándar muestra estas funciones, y como acceder a cada una de ellas, el controlador también intenta acceder de forma más sencilla a la funciones.

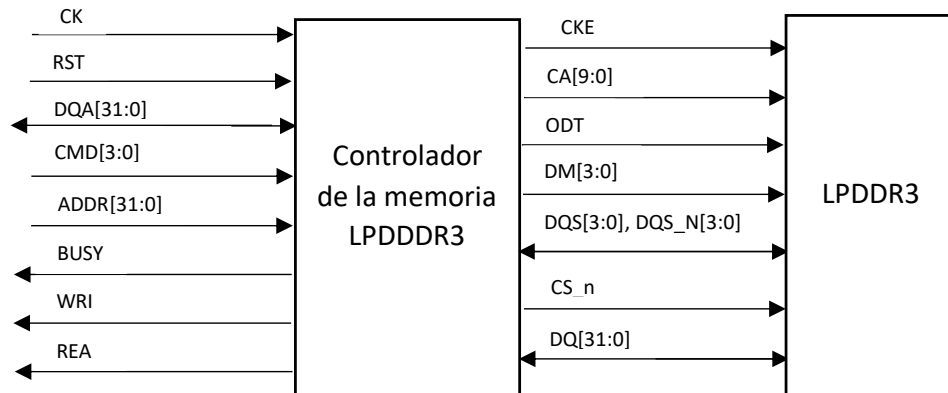
Para el uso de la memoria LPDDR3 se diseña un controlador. Este se ocupa del encendido de forma automática de la memoria como también una activación la escritura y lectura de datos en los bancos.

## 2. CONTROLADOR DE LA MEMORIA LPDDR3

El controlador de la memoria LPDDR3 es un circuito digital que se encarga de gestionar el flujo de datos entre el procesador y la memoria LPDDR3. puede ser separado o integrado en el chip. Los controladores de memoria contienen los elementos necesarios para leer y escribir en la DRAM. Según las normas de JEDEC [5].

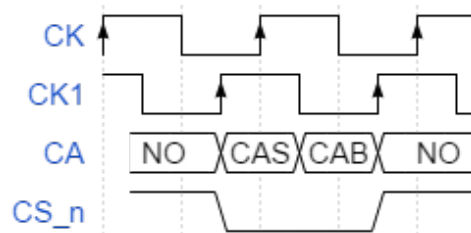
En la figura 1 se muestra las señales que se conectan entre el controlador y la memoria LPDDR3, estas señales son las mostradas en la tabla 2, además de las señales que se utilizan para manejar el controlador, que se especifican en la tabla 3.

Figura 1: Conexión del controlador a la memoria LPDDR3



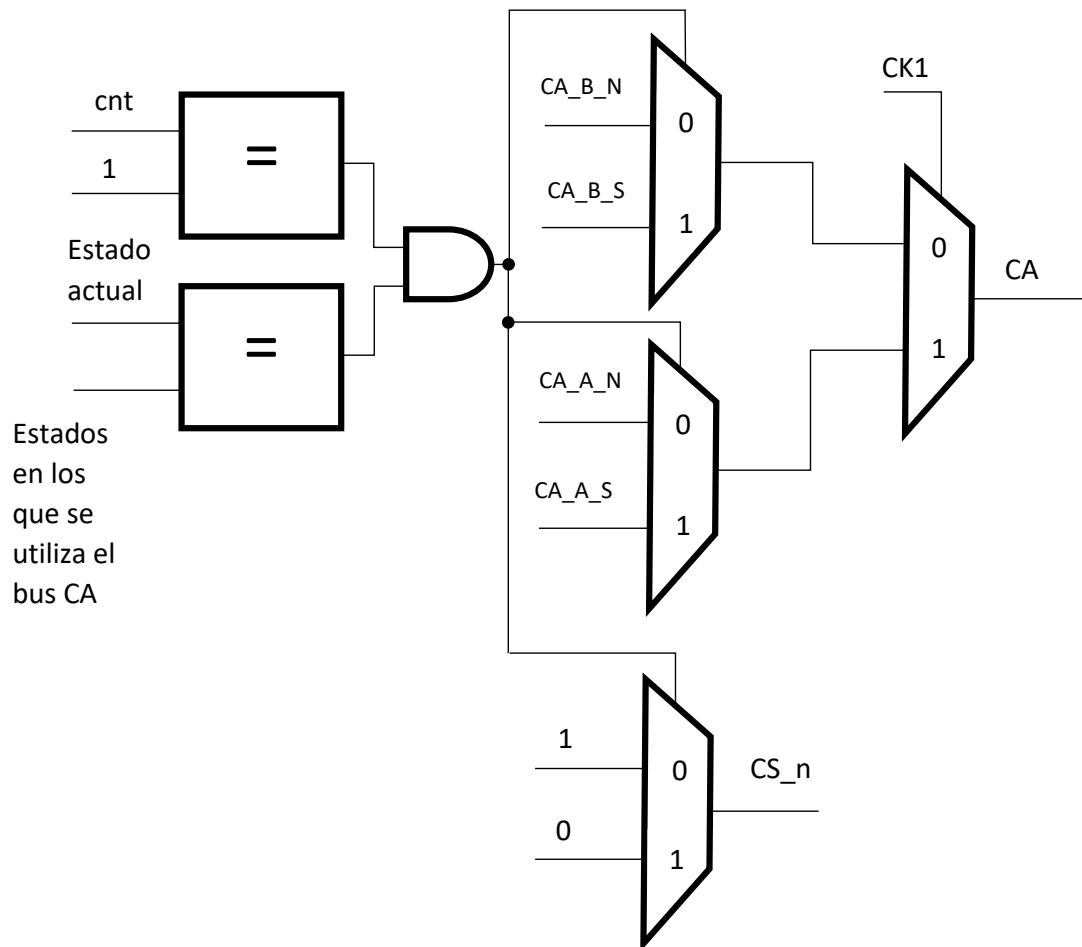
Para poderle ingresar funciones a la LPDDR3 es necesario que el controlador envíe una trama de bits por el flanco de subida y otra trama de bits por el flanco de bajada en un ciclo de reloj mientras se mantiene el bit CS\_n en un estado lógico bajo, como se muestra en la Figura 2.

Figura 2: Ingreso de funciones a la LPDDR3



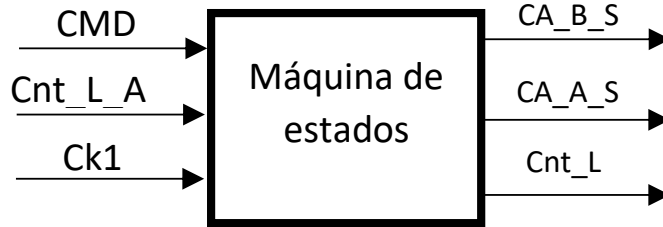
Para lograr enviar las dos tramas de bits se utiliza el circuito lógico mostrado en la Figura 3. Este circuito tiene dos módulos comparadores, cuando se cumplen se envía la trama de bits. Además de mantener el bit CS\_n en un estado lógico bajo. Las señales CA\_B\_S y CA\_A\_s son enviados de la maquina de estados. las señales CA\_B\_N y CA\_A\_N están guardados en registros y son tramas de bits para enviar comandos no operacionales.

Figura 3: Circuito digital para ingreso de funciones a la LPDDR3



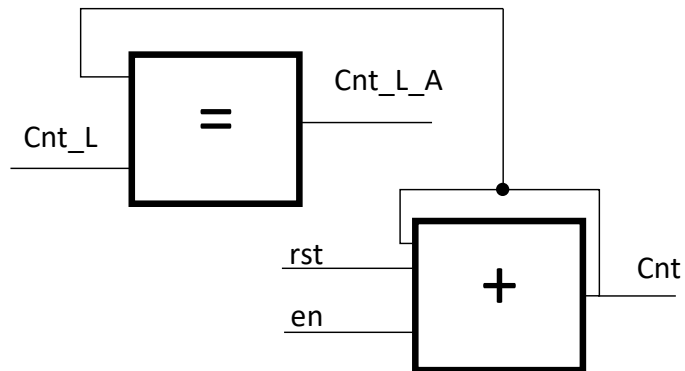
La maquina de estados tiene las señales mostradas en la figura 4 con las cuales controla los circuitos lógicos. Esta máquina de estados se le ingresa el bus CMD para poder elegir en cual estado acceder. las señales CA\_B\_S, CA\_A\_S y Cnt\_L son señales de salida. Las señales CA\_B\_S y CA\_A\_S son utilizadas para las el ingreso de funciones a la memoria LPDDR3. como las funciones tienen un tiempo limite de permanencia se utiliza la señal Cnt\_L para decirle al controlador que no cambie de estado.

Figura 4: Entradas y salidas de la máquina de estados



Para poder cambiar de estados es necesario tener el bit *Cnt\_L\_A* en un estado lógico alto. La máquina de estados coloca el limite del modulo de contador de flancos de reloj mostrada en la figura 5.

Figura 5: Contador de flancos de reloj



En la tabla 3 se muestra un resumen de las señales que maneja el controlador. Los Buses principales que manejan al controlador son la de *CMD* (Comandos) y *ADDR* (direccionamiento), con estas dos buses de señales se puede manejar a la memoria LPDDR3. Mientras que los bits *WRI* (escritura), *BUSY* (ocupado) y *REA* (lectura) son utilizadas para que el microcontrolador sepa cuando debe escribir datos o cuando tiene puede leer los datos que le envía la memoria, también cuando pueda ingresarle otro comando.

Tabla 3: Señales del controlador

Señales	Características
DQA[31:0]	Bus de datos que viene del microcontrolador al controlador.
CMD[3:0]	Con este bus se inserta la función deseada de la memoria.
ADDR[31:0]	ADDR[2:0]: Estos 3 bits son sirven para seleccionar el banco a usar.
	ADDR[17:3]: Estos bits sirven para seleccionar la fila del banco donde se va a leer o escribir un dato.
	ADDR[28:18]: Estos bits sirven para seleccionar la columna del banco donde se va a leer o escribir un dato.
BUSY	Sí está en estado alto el pin, no se puede enviar ningún comando al controlador y quiere decir que la memoria LPDDR3 está ejecutándose alguna función ingresada anteriormente
WRI	Cuando ese bit está en alto se puede escribir los datos del microcontrolador en el bus DQA.
REA	Cuando ese bit está en alto se puede leer los datos de la memoria LPDDR3 en el bus DQA.

Para poder iniciar alguna función del protocolo de la memoria LPDDR3 primero debe que estar la señal de *BUSY* en estado "0" para poderle ingresar un comando. En la tabla 4 se da un resumen de los valores binarios que puede tomar el bus *CMD* y a los que ingresa. Cuando se ingresa el comando *RD*, se espera a que el bit *REA* esté en estado alto, para poderle poderle empezar a enviar datos por el bus DQ a la memoria LPDDR3.

Tabla 4: Señales de comando del controlador

CMD	valor binario	Características
NOP	0000	No inicializa ningún comando del protocolo de la memoria LPDDR3 y solo se mantiene sin ninguna operación.
RESET	0001	Inicializa la función de reinicio del protocolo de la memoria LPDDR3.
MRR	0010	Con este comando inicializa el registro de modo de lectura.
MRW	0011	Con este comando inicializa el registro de modo de escritura.
ACT	0100	El controlador inicializa función para activar un banco de la memoria LPDDR3
PR	0101	Precarga un solo banco.
PRA	0110	Precarga todos los bancos
WR	0111	Escribe una ráfaga de datos.
RD	1000	Lee una ráfaga de datos.
PD	1001	Entra en modo de bajo consumo
PDX	1010	Sale del modo de bajo consumo
SREF	1011	Refresca todos los bancos de la memoria LPDDR3.
SREFX	1100	Salida del modo de refresco de la memoria LPDDR3.
DPD	1101	El controlador inicializa la función de bajo consumo previo al apagado de la memoria LPDDR3.
DPDX	1110	El controlador sale del apagado después de inicializar el DPD.
REF	1111	Refresca todos los bancos pero de uno a la vez de forma secuencial empezando desde el banco 0.

Si se desea ingresar un nuevo comando se debe esperar a que el bit *BUSY* vuelva a estado bajo. Pero mientras el bit *BUSY* está en alto no se puede ingresar a alguna diferente función que la que se está ejecutando, el mismo controlador no lo permite por más que se le ingrese un comando al microcontrolador.

## 2.1. INICIALIZACIÓN

Antes de la operación normal, la SDRAM LPDDR3 debe ser inicializada. La inicialización es utilizada para configuración de la LPDDR3, como es la frecuencia a la que se maneja, el tipo de ráfaga utilizada en la operación de escritura o lectura y la calibración de la impedancia de salida.

En la figura 6 muestra como es el proceso que realiza la máquina de estados para inicializar la memoria LPDDR3. Este proceso siempre tiene que suceder después de haber energizado correctamente la memoria LPDDR3.

Para comenzar la inicialización del protocolo memoria LPDDR3 se debe tener en cuenta que los niveles de tensión de las fuentes de alimentación de la misma sean estables y en el nivel de tensión requerido, Estas tensiones se deben establecer antes de un tiempo máximo de 20 [ms], en este tiempo CKE debe mantenerse en estado bajo, las salidas del dispositivo deben permanecer en alta impedancia y las entradas de la memoria deben estar en un estado alto o bajo para que no ocurra un *latch-up*.

Al estabilizarse la alimentación de la memoria se debe esperar un tiempo mínimo de 100 [ns] para que la entrada CKE cambie de un estado bajo a alto, para esto el Reloj diferencial debe permanecer estable por lo menos 5 ciclos de reloj antes de que la entrada CKE cambie de estado.

Cuando el estado de la entrada CKE cambie se debe esperar mínimo un tiempo de 200 [ $\mu$ s] para poder ingresar el comando ***MRW RESET***, este comando coloca el dispositivo en un estado de auto-inicialización en la secuencia de inicialización. Este comando reinicia todos los modos de los registros a sus valores predeterminados.

Al cumplirse el tiempo de introducción del comando ***MRW RESET*** y termine su funcionamiento se pueden introducir comandos ***MRR***, estos comandos se utilizan para saber la información básica de la memoria LPDDR3 con la que se trabaja, el valor del sensor de temperatura o la ID de fabricación de la memoria.

Figura 6: Inicialización del protocolo de la memoria LPDDR3



Si la inicialización se realiza a altas velocidades (mayores de una frecuencia de  $55\text{ MHz}$ ), entonces el **CA Training** puede ser necesario para cumplir con la configuración y mantener los tiempos. Como se desea trabajar con una frecuencia de ( $800\text{ MHz}$ ), se decide utilizar de manera predeterminada el **CA Training** en la secuencia de inicialización.

Como las entradas de los pines CA operan al doble velocidad de transmisión de datos, esto puede dificultar el control de la memoria LPDDR3 para satisfacer los tiempos de configuración a altas frecuencias, por eso un mecanismo de *CA Training* es utilizado.

El *CA Training* se divide en el ingreso de tres diferentes comandos *MR41*, *MR48* y *MR42*. Cuando se ingresa el comando *MR41* es para dar inicio a la calibración de los pines CA0, CA1, CA2, CA3, CA5, CA6, CA7 y CA8. Cuando se ingresa el comando *MR48* es para la calibración de los pines CA4 y CA9. Para dar finalización de ambas calibraciones se ingresa el comando *MR42*. Después del *CA Training* se ingresa el comando *MR10* para la *ZQ Calibration* (ZQC) este comando es usado para calibrar la impedancia de salida. El dispositivo está listo para su uso al cumplir el tiempo necesario para terminar el *ZQ Calibration*.

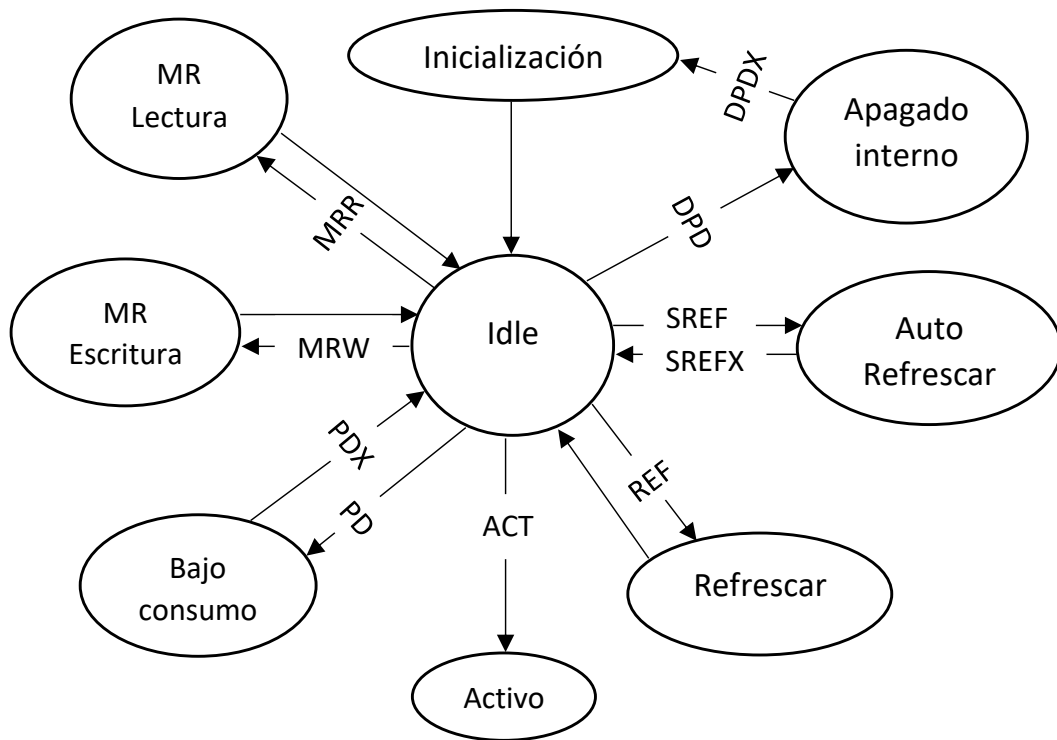
En la configuración LPDDR3 se ingresan tres diferentes comandos de MRW el *MR1*, *MR2* y *MR3*. El comando *MR1* se utiliza para configurar el valor del *nWR* donde es el número de ciclos de reloj los cuales determinan cuando empezar la operación de la precarga para una ráfaga de escritura. El comando *MR2* se utiliza para establecer la frecuencia de la LPDDR3 esto ayuda a cuadrar el número de ciclos antes de realizarse la lectura o escritura de la memoria LPDDR3. El comando *MR3* es para la configuración de la impedancia de los pines de entrada y salida.

## 2.2. ESTADO IDLE

Al terminar la inicialización este entra en el estado Idle, en este estado se puede utilizar cualquier comando valido y todos los bancos están precargados. A continuación se explican cada uno de los estados, los cuales pueden ser accedidos desde el estado Idle, en la figura 7 hace un breve resumen de la forma a acceder a los diferentes estados y su funcionamiento.

1. *apagado interno*: antes de entrar a este estado todos los bancos deben estar en idle con no actividad en el bus de datos. En este estado todas las fuentes de alimentación internas de la SDRAM son desactivadas y los datos al. Al salir de este estado, la SDRAM debe ser re-inicializada como se describe en la figura 7.
2. *Bajo consumo en modo Idle*: este estado a diferencia del *Apagado interno* no reinicia la SDRAM y puede volver al estado Idle sin que tenga que volver a inicializar la SDRAM.

Figura 7: Estados posible a acceder desde el estado Idle



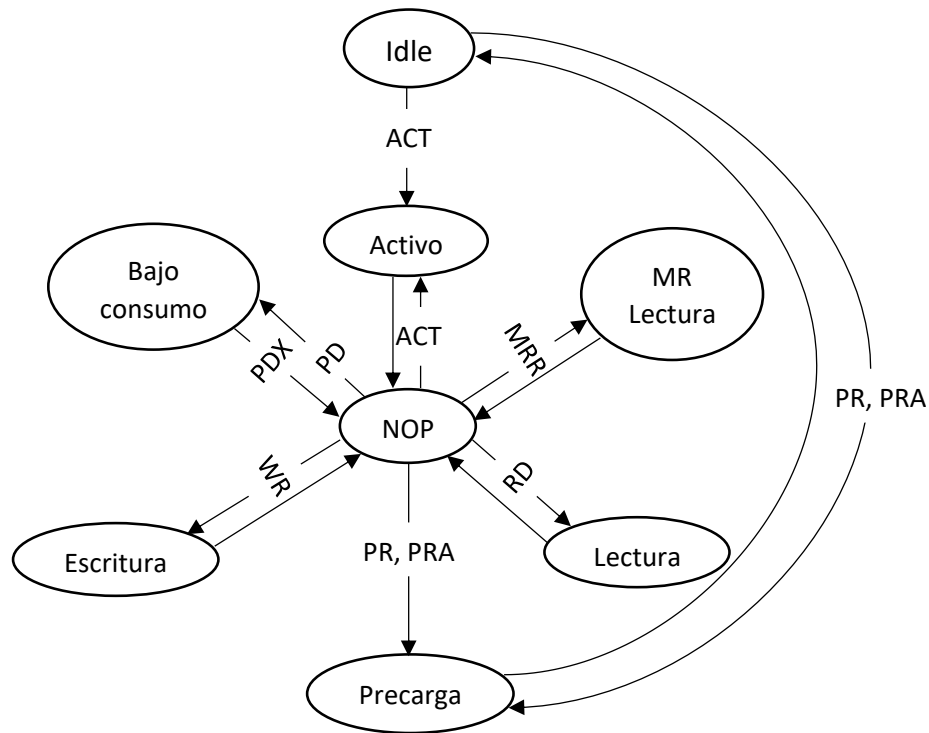
3. **Modo de registro de escritura:** Es usado para escribir datos de configuración en los registros. Solo puede ser usado cuando la SDRAM está en Idle.
4. **Modo de registro de Lectura:** Es usado para leer datos de configuración en los registros y el estado de los datos.
5. **Refrescar:** Este comando es introducido para la lectura periódica de información de un banco de memoria, e inmediatamente reescribir la información leída en el mismo banco sin modificaciones.
6. **Auto-Refresco:** Este comando puede utilizarse para retener datos en la LPDDR3 SDRAM sin relojes externos.
7. **Activo:** El comando **Activo** se utiliza para activar el banco en el cual se va a trabajar, debe ser aplicado antes de cualquier operación **Lectura** o **Escritura** (escritura) pueda ser ejecutada. El dispositivo puede aceptar un comando de lectura o escritura después de haber introducido el comando **Activo**. Este comando

solo se pueden activar 4 bancos al mismo tiempo y en este comando se ingresa la fila en el banco que se va a operar.

## 2.3. ESTADO ACTIVO

Al activarse un banco de la memoria LPDDR3, o haber tenido banco activo, la memoria LPDDR3 no puede acceder a algunas funciones que podía acceder anteriormente en estado *IDLE*. Después de entrar a este estado ya se pueden leer y escribir datos.

Figura 8: Estados posible a acceder después del ingreso del estado activo



En la figura 8 se muestran los estados posibles a acceder desde el estado *Activo*, como se puede observar, después de activar un banco donde se puede escribir o leer siempre se debe ingresar el comando *precarga* para poder activar un banco diferente o salir del estado *activo* y regresar al estado *Idle*. A continuación se explicara cada uno de los estados en los cuales se puede acceder desde el estado Activo.

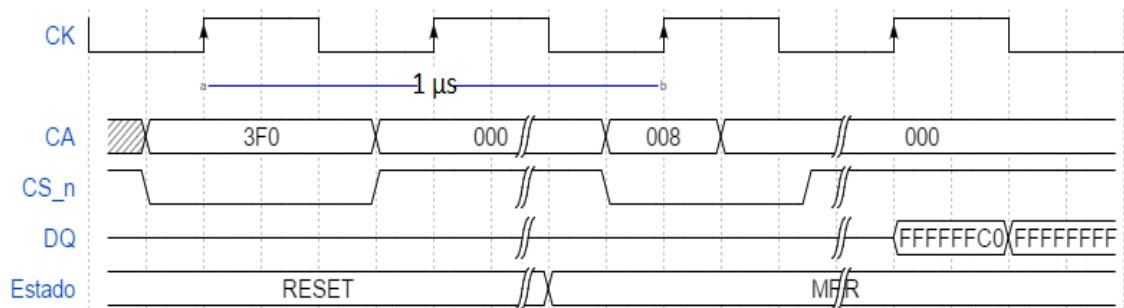
1. ***NOP:*** (No operacional) Al terminar de ingresar el comando ***Active*** el controlador ingresa directamente a este estado donde espera el ingreso de diferentes comandos.
2. ***Modo de resgistro de lectura:*** Este estado tiene la misma funcionalidad que el estado ***Modo de resgistro de lectura*** solo que puede ser accedido desde la activación de un banco y poder leer la configuración de la memoria LPDDR3. Pero como se puede observar desde el estado ***Activo*** no se pueden variar ningún parámetro de la LPDDR3 que se hace al ingresar algún comando ***MRW***.
3. ***Bajo consumo de energía:*** Para el Ahorro de energía se accede a este estado, sin la necesidad de volver a inicializar la LPDDR3, este estado es el mismo que el estado ***Idle Power Down***.
4. ***Precargar:*** este comando es usado para precargar o cerrar un banco que a estado activo. El comando precarga puede ser utilizado para precargar cada banco independientemente o todos los banco al mismo tiempo.
5. ***Lectura:*** En este estado se ingresa el banco (no se puede ingresar un banco no activo) y la columna en la cual se leerá el dato contenido.
6. ***Escritura:*** En este estado se ingresa el banco (no se puede ingresar un banco no activo) y la columna en la cual se escribirá el dato deseado.

### 3. RESULTADOS

Para mostrar la inicialización de la memoria LPDDR3 se divide en 3 partes en la primera parte se muestra como el controlador inicia las funciones RESET y MRR, Para que la memoria LPDDR3 comience se le ingrese cualquier comando el controlador tiene que ingresarle una serie de bits por el bus CA y mientras los envía CKE tiene que estar en nivel bajo. Para comenzar con la inicialización el controlador ingresa la función RESET, modo de registro de escritura, este se ingresa por el bus CA de la memoria como se muestra en la figura 9.

Después de haber ingresado el comando RESET, se ingresa el comando MRR este tiene un valor diferente en el flanco de subida como en el de bajada esto se muestra en la figura 9, modo de registro de lectura, con este comando nos entrega información de la memoria LPDDR3, la memoria se ejecuta dependiendo del MRR ingresado, en la inicialización es obligatorio ingresar el MR0 e cual nos dice la información del dispositivo, esto lo hace mediante el bus DQ. Como nos muestra en la figura 9, en el bus DQ nos entrega en los primeros 8 bits información de la LPDDR3, según el estándar de la JEDEC [5].

Figura 9: Inicialización de la memoria LPDDR3, primera parte



En la figura 9 muestra el resultado que nos entrega la memoria LPDDR3, este los entrega por los primeros 8 bits del bus DQ. En la tabla 5 nos muestra la información de que significa cada bit, a continuación se resumen:

1. Bit 0, tiene un valor de cero, esto significa que el DAI ya está completo.
2. Bit 1 y 2, tienen un valor de 0, estos dos bit no tienen importancia.

3. Bit 3 y 4, tienen un valor de 00, el auto test RZQ no está soportado en la memoria LPDDR3.
4. Bit 5, tiene un valor de 0, no tiene información relevante.
5. Bit 6, soporta el WL.
6. Bit 7, soporta el RL3.

Tabla 5: Información que entrega la LPDDR3 después de insertar el MRR0

Bit 0	0 <sub>B</sub> : DAI completo	Estado de la auto inicialización del dispositivo (por sus siglas en inglés, DAI)
	1 <sub>B</sub> : DAI incompleto	
Bit 1	No indican información	
Bit 2		
Bit 3 y 4	00 <sub>B</sub> : No da soporte a la auto calibración	RZQ información del test de calibración de la impedancia de los pines de salida
	01 <sub>B</sub> : El pin ZQ puede estar flotando	
	10 <sub>B</sub> : El pin ZQ puede estar conectado a GND	
	11 <sub>B</sub> : Auto calibración completada y sin errores	
Bit 5	No se utiliza	
Bit 6	0 <sub>B</sub> : No tiene soporte	Ancho de escritura (por sus siglas en inglés, WL)
	1 <sub>B</sub> : Tiene soporte	
Bit 7	0 <sub>B</sub> : No tiene soporte	Ancho de Lectura de 3 (por sus siglas en inglés, RL3)
	1 <sub>B</sub> : Tiene soporte	

Fuente: [5] Adaptado de JEDEC, Low Power Double Data Rate 3. JEDEC SOLID STATE TECHNOLOGY ASSOCIATION, Aug. 2015.

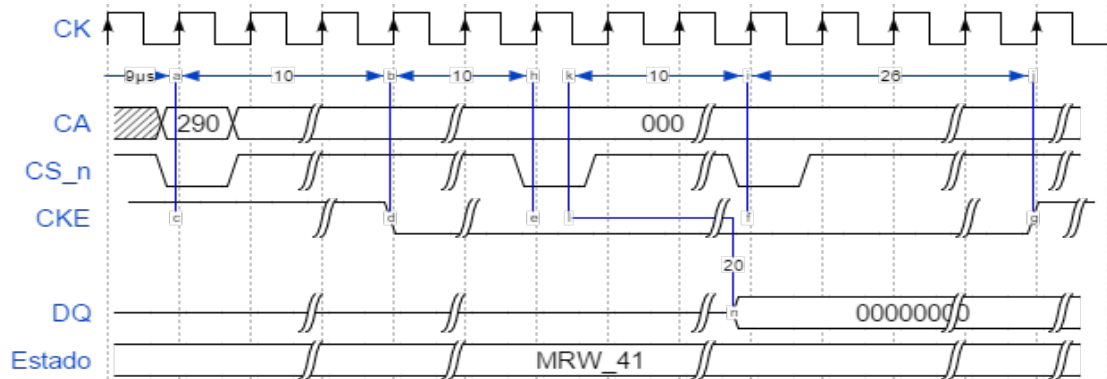
En la segunda parte de la inicialización el controlador empieza la función del CA Training, como se muestra en la figura 10. para dar comienzo al entrenamiento del bus CA se debe enviar el comando MRW 41, como se explicó anteriormente, para saber si funcionó correctamente en el estado MRW\_41\_caxr y MRW\_41\_cayr se manda un valor de 0 por el bus CA haciendo que la respuesta por el DQ sea igualmente 0. En la tabla 6 se explica que que bits se transmiten del bus CA al DQ cuando al mandarse las dos funciones.

Tabla 6: Envío de datos del bus CA al bus DQ, MRW 41

Bits del bus CA	CA0	CA1	CA2	CA3	CA5	CA6	CA7	CA8
Bits del bus DQ	DQ0	DQ2	DQ4	DQ6	DQ8	DQ10	DQ12	DQ14
	DQ1	DQ3	DQ5	DQ7	DQ9	DQ11	DQ13	DQ15

Fuente: [5] Adaptado de JEDEC, Low Power Double Data Rate 3. JEDEC SOLID STATE TECHNOLOGY ASSOCIATION, Aug. 2015.

Figura 10: Comienzo del entrenamiento del bus CA



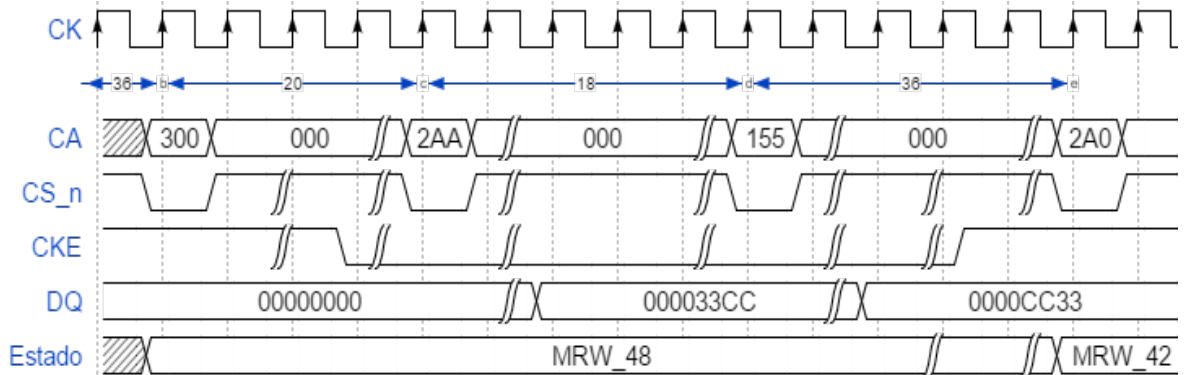
El MRW41 entrena bus CA del bit 0 al 3 y del 5 al 8, para el bit 4 y 8 se utiliza el comando MRW 48 en la figura 11 se muestra como es ingresado y la respuesta que se recibe. En el estado MRW\_48\_caxr se ingresa en hexadecimal 2AA o en binario 1010101010 nos interesan los bits 4 y 9 del bus CA los cuales tienen un valor de 1 y 1 respectivamente, los bits 0 y 1 del bus DQ aparecerá el valor del bit 4 del bus CA y los bits 8 y 9 del bus DQ aparecerá el valor del bit 9 del bus CA. En la tabla 7 se resume lo anterior.

Tabla 7: Envío de datos del bus CA al bus DQ, MRW 48

Bits del bus CA	CA4	CA9
Bits del bus DQ	DQ0	DQ8
	DQ1	DQ9

Fuente: [5] Adaptado de JEDEC, Low Power Double Data Rate 3. JEDEC SOLID STATE TECHNOLOGY ASSOCIATION, Aug. 2015.

Figura 11: Final del entrenamiento del bus CA

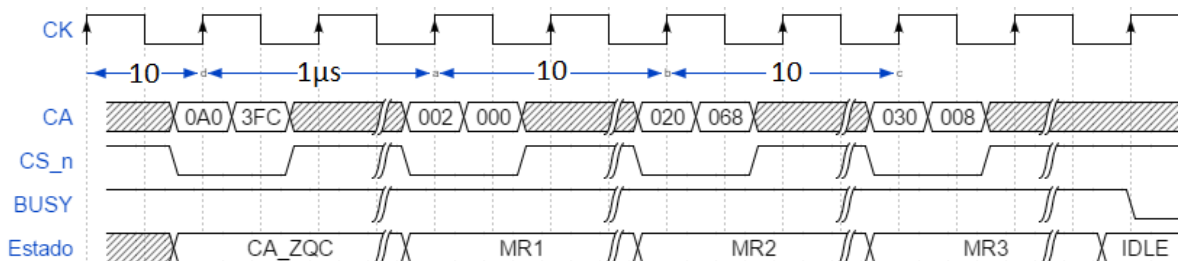


En la tercera parte de la inicialización el controlador empieza la configuración de la memoria LPDDR3, con las funciones de registro de modo de escritura MRW 1, MRW 2 y MRW 3, como se muestra en la figura 12.

El MRW 1 se utiliza para definir el número de datos que se pueden enviar en una ráfaga (el protocolo de la memoria LPDDR3 por defecto tiene 8 datos por ráfaga y los demás están reservados), el número de bits que toca esperar después de enviar la función de escritura o lectura para ingresar el dato o recibirlo, se utiliza el modo por defecto del protocolo de la memoria LPDDR3 que serían 10 ciclos de reloj y 8 en la ráfaga de datos.

El MRW 2 se utiliza para definir la frecuencia a la que se va a trabajar el protocolo de la memoria LPDDR3 por defecto se utiliza la de 800 [MHz] y en por finalizar se ingresa el registro MRW 3 el cual se ingresa para configurar la impedancia de los pines de entrada y salida se utiliza la de por defecto de 40  $\Omega$ .

Figura 12: Configuración de la memoria LPDDR3

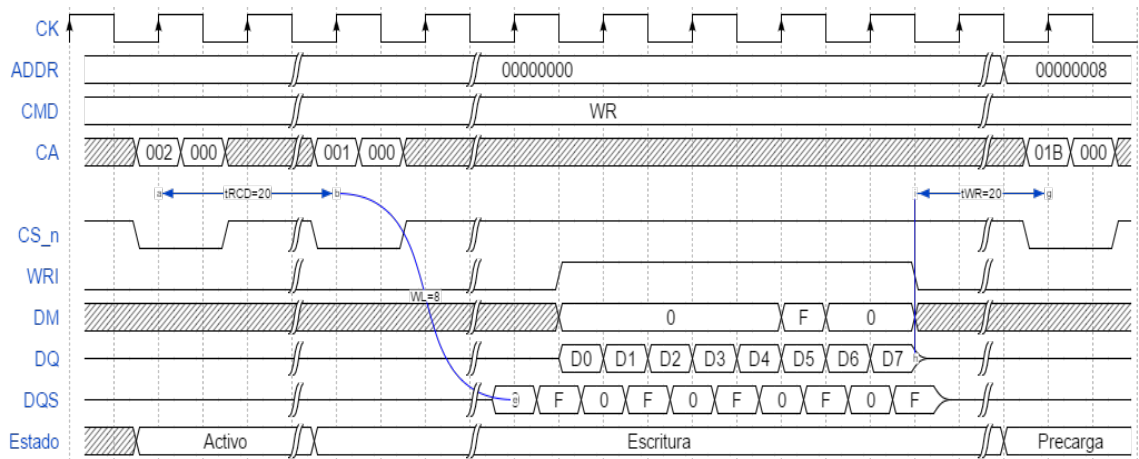


Para la escritura de un dato en la memoria LPDDR3 se utiliza el comando Active para activar un banco y al mismo tiempo asigna la fila en la cual se escribirá la ráfaga de

datos, después comienza el comando de escritura donde se especifica la columna. En una Ráfaga se va aumentando el valor de la columna. Además si no se quiere escribir un dato se pone en alto el valor del pin DM, como se muestra en la figura 13.

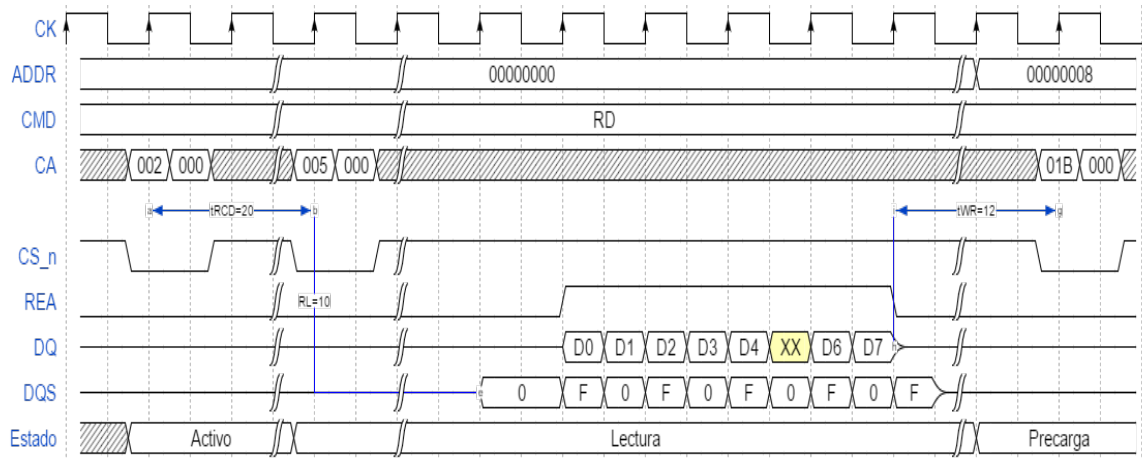
En la figura 13 se puede observar que ADDR está todo en 0 esto quiere decir que se activó el banco 0, se escribió en la fila 0 y columna 0. esto quiere decir que empezará a escribir desde la columna 0 y terminará en la columna 7, además de eso en dos columnas no se escribirán, en la columna 4 y 5 haciendo que sí se leen a partir de esa columna no se tendrá ningún valor. El bus DQS se utiliza para muestrear el bus DQ en la memoria LPDDR3

Figura 13: Escritura de una ráfaga de datos en la memoria LPDDR3



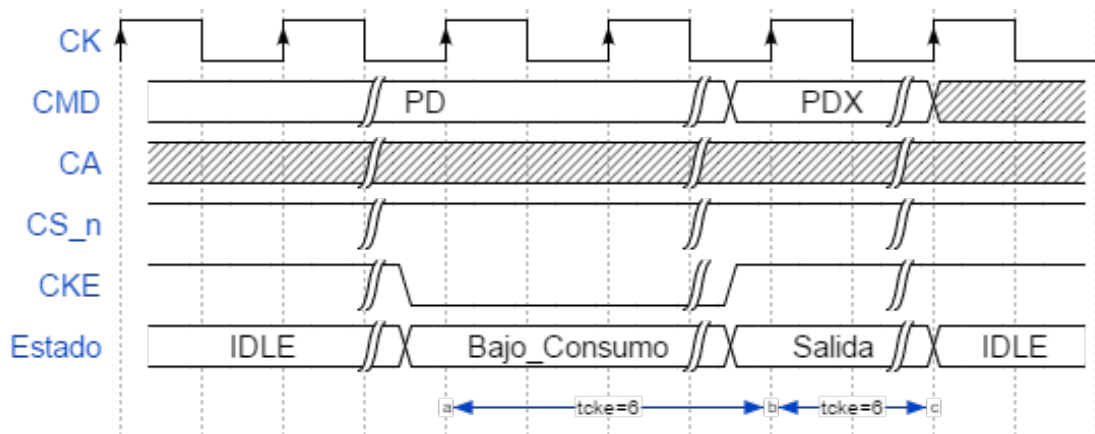
En la Lectura de un dato de la memoria LPDDR3 también se necesita utilizar el comando Active después de un comando precarga. Al terminar de activar el banco se inicializar el comando de lectura, como se muestra en la figura 14, el dato enmascarado no se escribió en la memoria LPDDR3. Esta ráfaga de lectura empezó desde la columna 0 haciendo y terminando en la columna 7, de esta forma en la columna 4 y 5 no tengan valor.

Figura 14: Lectura de una ráfaga de datos en la memoria LPDDR3



Una función importante del protocolo de la memoria LPDDR3 es que pueda seguir funcionando pero que no consuma tanta energía, entonces se introduce la función de baja potencia, esta se inicializa al colocar la señal CKE en un nivel lógico bajo, esto se demuestra en la figura 15. Como se podrá observar el bus CA no es utilizado para ingresar la función de la LPDDR3 el controlador no tiene que poner en estado bajo el CS\_n ni tampoco ingresar valores por el bus CA.

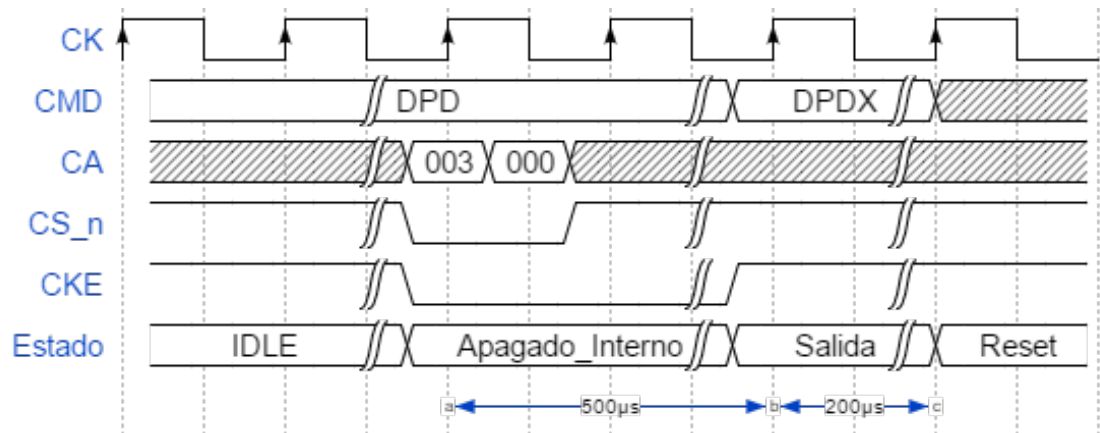
Figura 15: Función de baja potencia



Antes de apagar la memoria LPDDR3, se debe apagar internamente, por eso se debe inicializar la función Apagado interno (Deep Power Down). Al igual que la función de bajo consumo la señal CKE se pone en un nivel lógico bajo, pero también se introduce

la señal CA y CS\_n para diferenciar de la otra función, como se muestra en la figura 16. Pero también se puede salir de esta función al inicializar el comando de salida.

Figura 16: Función de apagado interno de la memoria



Como la memoria tiene que refrescarse se pueden utilizar dos comandos el de auto-refrescar y el de refrescar. El comando de refrescar solo refresca un banco a la vez y de forma secuencial empezando desde el banco 0 como se muestra en la Figura 17. El comando de auto-refresco refresca todos los bancos, el ingreso de la función se muestra en la Figura 18

Figura 17: Función de Refresco

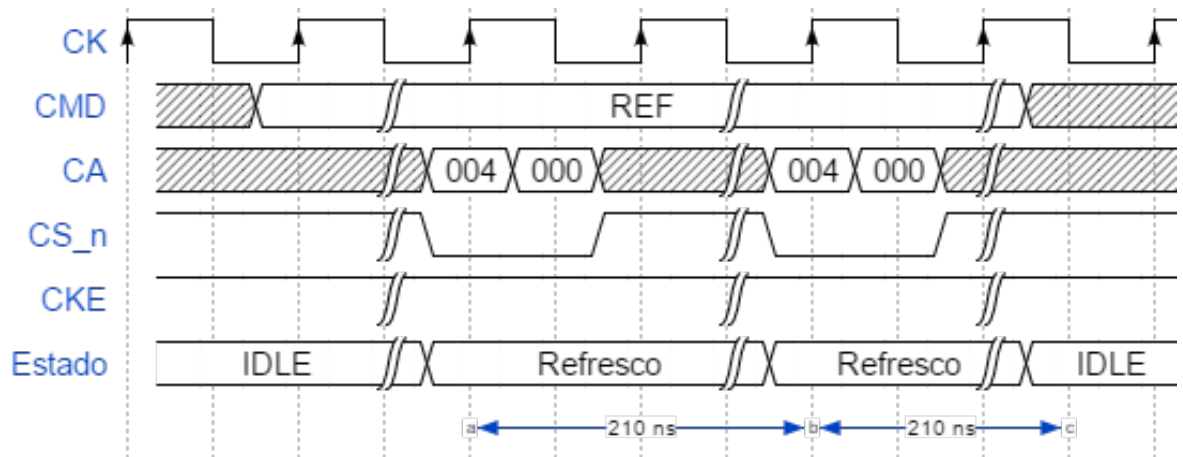
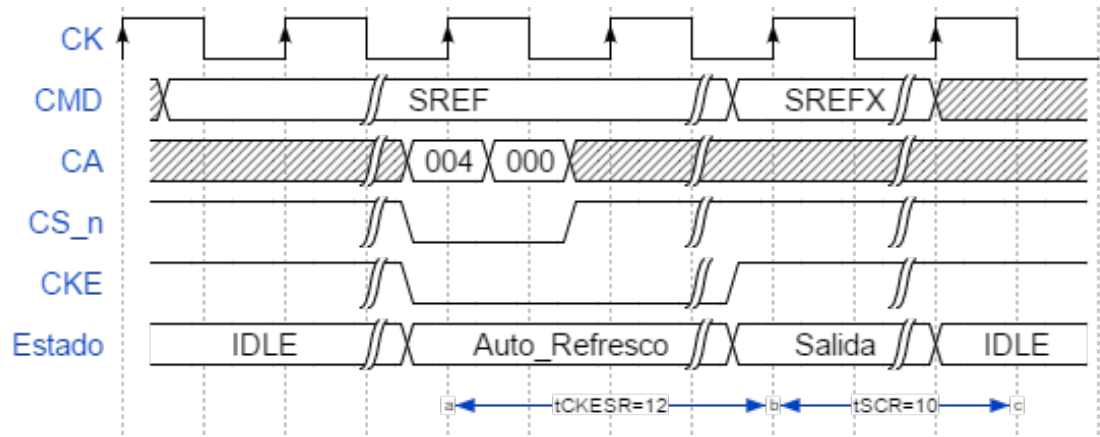


Figura 18: Función de Auto-Refresco



El circuito digital fue sintetizado con una tecnología de 180 [nm], el circuito tiene 1027 componentes lógicos en un área aproximada de 10700,41 [ $\mu m^2$ ] con una densidad de potencia de 8,34 [ $\mu W/MHz$ ].

## 4. CONCLUSIONES

Este trabajo presenta el diseño de una máquina de estados para el control de la memoria LPDDR3. Con el archivo de simulación se comprobó el funcionamiento del controlador, empezando con la secuencia predefinida de pasos para inicializar la LPDDR3 y configurar varios parámetros para el funcionamiento normal de la SDRAM. Otras funciones añadidas fueron la activación de un banco de memoria, escritura y lectura de datos en diferentes posiciones, acceso a la función de bajo consumo de energía y apagado interno de la LPDDR3.

Una gran parte de este controlador puede ser utilizado para diferentes versiones de la memoria LPDDR por lo que no tienen cambios significativos, haciendo posible una migración a partir de este diseño [4]. Para trabajos futuros se puede incluir un módulo de auto-refresco donde se inicialice el comando de forma automática sin que el microprocesador ejecute la función, y se puede sintetizar con una tecnología de 45nm o inferior.

## REFERENCIAS

- [1] CRISP, J. Introduction to microprocessors and microcontrollers. Newnes, Ed., vol. 2, p. 72.
- [2] DOULOS. Designers guide verilog. [en línea] disponible en: [https://www.doulos.com/knowhow/verilog\\_designers\\_guide/](https://www.doulos.com/knowhow/verilog_designers_guide/).
- [3] GIBB, A. *Building Open Source Hardware*, vol. 1. Addison-Wesley Professional, 2014.
- [4] JEDEC. Migrating to low power double data rate 3 (lpddr3). JEDEC SOLID STATE TECHNOLOGY ASSOCIATION.
- [5] JEDEC. Low power double data rate 3 (lpddr3). JEDEC SOLID STATE TECHNOLOGY ASSOCIATION.
- [6] PAPER, M. W. Ecc brings reliability and power efficiency to mobile devices, 2017. [en línea]. Disponible en: <https://www.micron.com/resource-details/28c643e4-9f86-49cd-9471-f386a0b812ca>.
- [7] PC, M., AND SMITH, W. *Maximum PC Guide to Building a Dream PC*. Que, 2004.
- [8] PETLEY, G. Vlsi and asic technology standard cell library design, open source standard cell libraries. [Online] Available: <http://www.vlsitechnology.org/html/libraries.html>, September 2015.
- [9] SUNDMAEKER, H., GUILLEMIN, P., FRIESS, P., AND WOELFFLÉ, S., Eds. *Vision and Challenges for Realising the Internet of Things*. Publications Office of the European Union, Luxembourg, 2010.
- [10] TANENBAUM, A. S. Organizacion de computadoras: Un enfoque estructurado. PHH, Ed., vol. 4, p. 153.
- [11] WEISS, G. G. B. Introduction to microcontrollers. V. U. of Technology Institute of Computer Engineering Embedded Computing Systems Group, Ed., p. 23.

# BIBLIOGRAFÍA

CRISP, John. Introduction to Microprocessors and Microcontrollers. 2 ed. Burlington,,: Elsevier, 2004. 288p.

ELLIS, John. Life Without Termination – Delivering Successful LPDDR/LPDDR2 Interfaces. DesignCon, 2011

SMITH, Douglas. HDL Chip Design: A practical guide for designing, synthesizing and simulating ASICs an FPGAs using VHDL or Verilog. Madison,: Doone Publications, 1996.p. 448.

JEDEC. Low Power Double Data Rate 3. Solid State Technology Association, 2015.

MORRIS, Mano. Arquitectura de computadoras. México: Prentice Hall Hispanoamericana, 1999.p. 641.

SAMIR, Palnitkar. verilog HDL - A guide to Digital Design And Synthesis. Sun Microsystems Inc, 2003.