



Banco de pruebas de degradación de memorias NAND Flash

Proyecto de investigación

Por

César Arguello Rubio

En cumplimiento a los requerimientos para la obtención de
la Especialidad de Tecnólogo en Mecatrónica

Revisor académico: Dagoberto de León Gordillo

Santiago de Querétaro, Qro., México, Septiembre 2020.



CENTRO DE INGENIERIA Y DESARROLLO INDUSTRIAL
CENTRO DE INFORMACION Y DOCUMENTACION TECNOLOGICA

AUTORIZACIÓN
PUBLICACIÓN EN FORMATO ELECTRÓNICO DEL TRABAJO TERMINAL,
ESPECIALIDAD

Fecha: 03/02/2021

El que suscribe
 Alumno (a) César Arguello Rubio
 CURP AURC940522HQTRBS08 CVU 918994

ORCID 0000-0002-5881-2816

Correo electrónico (opcional)
c.arguello@posgrado.cidesi.edu.mx

Egresado (a) de Especialidad de Tecnólogo en Mecatrónica

Autor del Trabajo Terminal, Especialidad titulado
Banco de pruebas de degradación de memorias NAND Flash

Por medio del presente documento autorizo¹ en forma gratuita y permanente a que el Trabajo terminal, Especialidad arriba citada sea divulgada y reproducida para publicarla mediante almacenamiento electrónico que permita el acceso al público a leerla y conocerla visualmente, así como a comunicarla públicamente en Página Web.
 La única contraprestación que condiciona la presente autorización es la del reconocimiento del nombre del autor en la publicación que se haga de la misma.

Atentamente



 César Arguello Rubio
 Nombre y firma del estudiante

¹ Lev Federal de Derechos de Autor
 Para obtener tu ORCID regístrate en: <https://orcid.org/register>

Contenido temático

Introducción.....	1
Antecedentes	2
Planteamiento del problema.....	5
Objetivo	6
Justificación	7
Metodología	8
Marco teórico.....	10
Memorias NAND Flash.....	10
Raspberry Pi 3®	15
GPIO de Raspberry Pi®.....	16
Sistema operativo Raspberry Pi OS.....	17
Python con Raspberry Pi®.....	18
Actualizar la versión de Raspbian® en Raspberry Pi®.....	18
Registrador de datos	20
GRAFCET	21
Desarrollo del Banco de Pruebas de degradación de memorias NAND Flash	22
Interfaz gráfica de usuario	24
Selección del tipo de señal que se desea transmitir al dispositivo NAND Flash.....	25
Graficar datos a transmitir al dispositivo NAND Flash.....	25
Cambiar color de fondo de la interfaz.....	28
Lectura de ID del dispositivo NAND Flash.....	29
Borrar un bloque del dispositivo NAND Flash.....	33
Lectura de una página del dispositivo NAND Flash.....	39

Escritura de una página del dispositivo NAND Flash.	42
Mapear bloques defectuosos.	46
Proceso de degradación del dispositivo NAND Flash.	49
Resultados	53
Conclusiones.....	69
Bibliografía.....	70

Lista de figuras

Figura 1 Transistor MOSFET (efecto de campo metal-óxido-semiconductor) con puerta flotante.....	2
Figura 2 Niveles de almacenamiento en una celda en memorias NAND Flash SLC, MLC y TLC.....	4
Figura 3 Banco de pruebas de degradación de memorias NAND Flash.....	8
Figura 4 Especificaciones para el banco de pruebas de degradación de memorias NAND Flash.....	9
Figura 5 Organización del dispositivo con encapsulado de matriz única. (Micron®, 2020).	11
Figura 6 Organización del dispositivo con encapsulado de dos matrices. (Micron®, 2020).	13
Figura 7 Arquitectura del dispositivo NAND Flash Micron® de 256 Gb modelo 29F256G08CJABB, organizado con bloques (Micron®, 2020).	14
Figura 8 Tarjeta electrónica Raspberry Pi® modelo B (RaspberryPi®, 2020).....	15
Figura 9 GPIO Raspberry Pi® modelo B. (RaspberryPi®, 2020).....	16
Figura 10 Raspberry Pi Imager.....	17
Figura 11 Consola de comandos de Raspberry Pi®.	19
Figura 12 Esquemático PCB del Registrador de datos.	20
Figura 13 Diagrama tipo GRAFCET básico.	21
Figura 14 Conexiones físicas del banco de pruebas.	23
Figura 15 Interfaz gráfica de usuario en Raspberry Pi®.....	24
Figura 16 Selección de modelo de memoria NAND Flash.	24
Figura 17 Selección de señal a transmitir al dispositivo NAND Flash.....	25
Figura 18 Diagrama de control para graficar datos a transmitir al dispositivo NAND Flash.	26
Figura 19 Grafica de una señal seno de 2 Hz.	26
Figura 20 Grafica de una señal diente de sierra 2 Hz.	26
Figura 21 Grafica de una señal aleatoria.....	27

Figura 22 Grafica de una señal constante con amplitud de 126.	27
Figura 23 Selección de color de fondo para la interfaz gráfica.	28
Figura 24 Selección de color de fondo “Coral” para la interfaz gráfica.	28
Figura 25 Obtener ID del dispositivo NAND Flash mediante la interfaz gráfica de usuario.	29
Figura 26 Diagrama de control para la lectura de ID del dispositivo NAND Flash.....	29
Figura 27 Comportamiento de señales para enviar un comando al dispositivo NAND Flash (BROADCOM, 2008).	31
Figura 28 Comportamiento de señales para leer el ID del dispositivo NAND Flash (BROADCOM, 2008).	32
Figura 29 Código ID del dispositivo NAND Flash MT29F256G08CJABB.	32
Figura 30 Borrar un bloque del dispositivo NAND Flash mediante la interfaz gráfica de usuario.	33
Figura 31 Diagrama de control para borrar un bloque del dispositivo NAND Flash.	34
Figura 32 Comportamiento de señales para enviar los bytes de dirección al dispositivo NAND Flash (BROADCOM, 2008).	35
Figura 33 Comportamiento de señales para borrar un bloque del dispositivo NAND Flash (BROADCOM, 2008).	36
Figura 34 Comportamiento de señales para LEER EL ESTADO del dispositivo NAND Flash (BROADCOM, 2008).	38
Figura 35 Bloque borrado del dispositivo NAND Flash mediante la interfaz gráfica de usuario.	38
Figura 36 Leer una página del dispositivo NAND Flash mediante la interfaz gráfica de usuario.	39
Figura 37 Diagrama de control para leer una página del dispositivo NAND Flash.	40
Figura 38 Comportamiento de señales para leer una página del dispositivo NAND Flash (BROADCOM, 2008).	42
Figura 39 Página leída del dispositivo NAND Flash mediante la interfaz gráfica de usuario.	42
Figura 40 Escribir en una página del dispositivo NAND Flash mediante la interfaz gráfica de usuario.	43

Figura 41 Diagrama de control para escribir a una página del dispositivo NAND Flash....	43
Figura 42 Comportamiento de señales para escribir en una página del dispositivo NAND Flash (BROADCOM, 2008).....	44
Figura 43 Bytes enviados a una página del dispositivo NAND Flash mediante la interfaz gráfica de usuario.....	45
Figura 44 Diagrama de control para mapear bloques defectuosos del dispositivo NAND Flash.	47
Figura 45 Mapear bloques defectuosos del dispositivo NAND Flash mediante la interfaz gráfica de usuario.....	48
Figura 46 Barra de progreso de mapeo de bloques malos del dispositivo NAND Flash en la interfaz gráfica de usuario.	48
Figura 47 Degradación de memoria NAND Flash mediante la interfaz gráfica de usuario.	49
Figura 48 Diagrama de control para el proceso de degradación, cálculo y recopilación de los resultados obtenidos.	50
Figura 49 Grafica de datos recibidos del dispositivo NAND Flash en la interfaz gráfica de usuario.....	51
Figura 50 Indicadores de proceso de ejecución de la secuencia Borra-Escribir-Leer del dispositivo NAND Flash en la interfaz gráfica de usuario.....	52
Figura 51 Paro controlado de la secuencia Borra-Escribir-Leer del dispositivo NAND Flash en la interfaz gráfica de usuario.....	52
Figura 52 Resultados obtenidos del banco de pruebas empleando una señal diente de sierra de 1 Hz: número de bytes detectados con errores por ciclo.	54
Figura 53 Resultados obtenidos del banco de pruebas empleando una señal diente de sierra de 1 Hz: número de bits detectados con errores por ciclo.....	55
Figura 54 Resultados obtenidos del banco de pruebas empleando una señal diente de sierra de 1 Hz: RBER por ciclo.....	55
Figura 55 Resultados obtenidos del banco de pruebas empleando una señal 0 constante: número de bytes detectados con errores por ciclo.	61
Figura 56 Resultados obtenidos del banco de pruebas empleando una señal 0 constante: número de bits detectados con errores por ciclo.....	62

Figura 57 Resultados obtenidos del banco de pruebas empleando una señal 0 constante: RBER por ciclo.	62
Figura 58 Comparación de resultados obtenidos para una señal tipo diente de sierra y una señal constante 0.	68

Lista de tablas

Tabla 1 Descripción de señales de un dispositivo NAND Flash. (Micron®, 2020).....	12
Tabla 2 Esquema de direccionamiento del dispositivo NAND Flash MLC de 256 Gb. (Micron®, 2020).....	14
Tabla 3 Parámetros de tiempos mínimos y máximos para el envío de comandos al dispositivo NAND Flash (BROADCOM, 2008).....	30
Tabla 4 Parámetros de tiempos mínimos y máximos para leer el ID dispositivo NAND Flash (BROADCOM, 2008).....	31
Tabla 5 Parámetros de tiempos mínimos y máximos para el envío de los bytes de dirección al dispositivo NAND Flash (BROADCOM, 2008).....	35
Tabla 6 Parámetros de tiempos mínimos y máximos para borrar un bloque del dispositivo NAND Flash (BROADCOM, 2008).....	36
Tabla 7 Parámetros de tiempos mínimos y máximos para verificar el estado del dispositivo NAND Flash (BROADCOM, 2008).....	37
Tabla 8 Parámetros de tiempos mínimos y máximos para leer una página del dispositivo NAND Flash (BROADCOM, 2008).....	41
Tabla 9 Parámetros de tiempos mínimos y máximos para escribir en una página del dispositivo NAND Flash (BROADCOM, 2008).....	45
Tabla 10 Bloques defectuosos del dispositivo Micron® modelo MT29F256G08CJABB..	53
Tabla 11 Número de bloque y página para prueba de degradación con señal diente de sierra 1 Hz.....	54
Tabla 12 Bytes con errores detectados de 10 ciclos consecutivos del bloque 454.....	56
Tabla 13 Bytes con errores detectados de 10 ciclos consecutivos del bloque 1963.....	57
Tabla 14 Bytes con errores detectados de 10 ciclos consecutivos del bloque 2784.....	58
Tabla 15 Bytes con errores detectados de 10 ciclos consecutivos del bloque 5654.....	59
Tabla 16 Bytes con errores detectados de 10 ciclos consecutivos del bloque 7721.....	60
Tabla 17 Número de bloque y página para prueba de degradación con señal constante 0..	61
Tabla 18 Bytes con errores detectados de 10 ciclos consecutivos del bloque 93.....	63
Tabla 19 Bytes con errores detectados de 10 ciclos consecutivos del bloque 3538.....	64

Tabla 20 Bytes con errores detectados de 10 ciclos consecutivos del bloque 6372.	65
Tabla 21 Bytes con errores detectados de 10 ciclos consecutivos del bloque 7209.	66
Tabla 22 Bytes con errores detectados de 10 ciclos consecutivos del bloque 6243.	67

Introducción

Los dispositivos de almacenamiento NAND Flash han encontrado su uso en un número cada vez mayor de aplicaciones de informática y dispositivos electrónicos, han ido reemplazando los discos duros convencionales, debido a su alta tasa de transferencia de datos, durabilidad y bajo consumo de energía (MicronTechnologies®, 2006).

A medida que ha continuado la búsqueda de productos de menor potencia, más ligeros y más robustos, las memorias de tipo NAND Flash se han convertido en la opción de almacenamiento líder para una amplia gama de aplicaciones.

Diversos equipos instrumentados de CIDESI emplean memorias de tipo NAND Flash, esto para aprovechar su mayor densidad y menor costo para aplicaciones de alto rendimiento frente a dispositivos de almacenamiento como memorias de tipo flash NOR y discos duros convencionales.

Los dispositivos NAND Flash se componen de celdas, las cuales son las encargadas de almacenar la información. Las celdas NAND Flash son un 60% más pequeña que las celdas de las memorias de tipo flash NOR, lo que proporciona densidades más altas, lo cual es requerido para los dispositivos de bajo consumo de hoy en día.

Los dispositivos NAND Flash pueden usar tecnología de celda de un solo nivel SLC, donde cada celda puede almacenar un dígito binario o tecnología multinivel como MLC y TLC, donde cada celda puede almacenar múltiples dígitos binarios, la ventaja principal de los dispositivos NAND Flash que almacenan más de un bit por celda es que pueden tener una mayor capacidad de almacenamiento a un menor costo por unidad de almacenamiento.

Una demanda cada vez mayor de capacidad de almacenamiento junto con la implementación de la tecnología Flash MLC y TLC, ha generado la necesidad de crear métodos de detección y corrección de errores más potentes, esto es porque estos dispositivos se degradan a medida que se escribe y se borra información, en consecuencia, aumenta la probabilidad de error en el almacenamiento de información en el dispositivo.

Antecedentes

Físicamente dentro de una memoria NAND Flash se encuentran dos chips, uno de control y otro chip que es una matriz NAND Flash, las matrices NAND Flash pueden contener múltiples planos dependiendo de la capacidad de almacenamiento, los planos se dividen en bloques, los bloques en páginas y las páginas en celdas, la cuales son el medio de almacenamiento y, dependiendo de la tecnología, pueden almacenar uno o más bits.

Las celdas están constituidas por compuertas lógicas NAND, las cuales están conformadas por transistores de tipo MOSFET (efecto de campo metal-óxido-semiconductor) con una puerta adicional llamada puerta flotante, esta puerta flotante se encuentra entre dos capas de aislamiento y puede almacenar electrones incluso cuando se corta la energía (ver Figura 1).

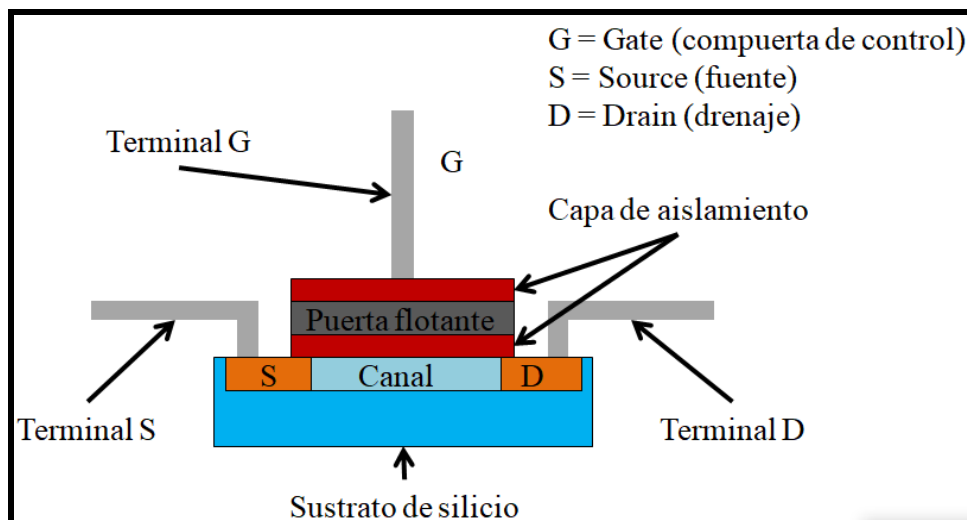


Figura 1 Transistor MOSFET (efecto de campo metal-óxido-semiconductor) con puerta flotante.

Para programar una celda se debe aplicar un voltaje a la puerta de control (G) mientras que la fuente (S) y el drenaje (D) están conectados a tierra (ver Figura 1), a partir de un cierto valor de voltaje (dependiendo de las características eléctricas del transistor) se genera un canal entre la fuente (S) y el drenaje (D) formado de electrones del sustrato de silicio, lo cual permite la circulación de corriente entre ambas terminales, cuanto mayor sea el voltaje aplicado a la terminal de control G más ancho se hará el canal entre la fuente y el drenaje, ya que la puerta flotante está aislada eléctricamente, puede almacenar una carga eléctrica incluso cuando la energía de la puerta de control se quita, es decir, la puerta flotante se comporta como un capacitor.

Para borrar la carga almacenada en la puerta flotante, un alto voltaje debe ser aplicado a través de la fuente (S) y drenaje (D) mientras se aplica un voltaje negativo a la puerta de control (G), esto hace que los electrones retrocedan a través de la capa de aislamiento anulando el canal entre la fuente y el drenaje.

Para leer el valor almacenado en la celda se aplica un voltaje de referencia a través de la fuente (S) y el drenaje (D) y luego la corriente se mide a la salida de la terminal de drenaje, la cantidad de corriente varía dependiendo de si la puerta flotante contiene carga o no, es decir, si la celda está programada o no.

Las celdas NAND Flash pueden ser programadas por diferentes cantidades de voltajes aplicadas a la puerta de control G, dependiendo de los umbrales de voltaje son diferentes cantidades de información binaria que pueden ser asociadas a cada celda, las celdas pueden tener tecnología SLC (Single Level Cell), MLC (Multi Level Cell), TLC (Triple Level Cell) en la cual es posible almacenar 1, 2, y 3 bit por celda de memoria respectivamente. En la Figura 2 se puede observar los niveles de almacenamiento de energía según la tecnología del dispositivo NAND Flash.

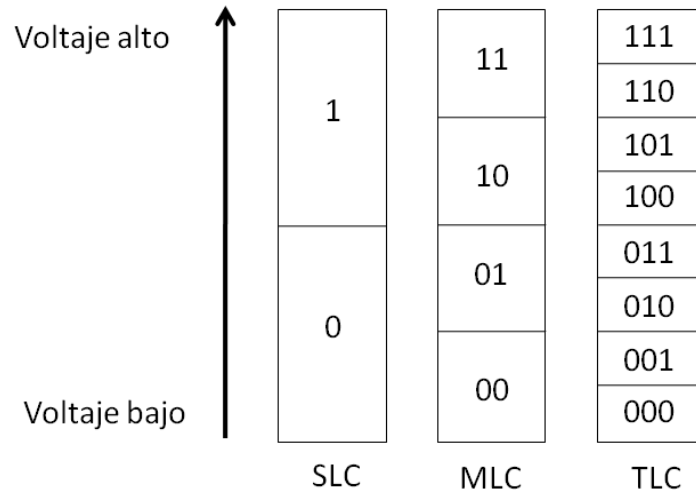


Figura 2 Niveles de almacenamiento en una celda en memorias NAND Flash SLC, MLC y TLC.

Cada una de estas tecnologías tiene un gran impacto en el rendimiento y durabilidad del dispositivo NAND Flash. Esto es porque cada vez que se programa o borra una celda, el movimiento de los electrones daña la capa de aislamiento debajo de la puerta flotante y comienza a desgastarse, a medida que se desgastan las capas aislantes comienza a empeorar la capacidad de retener cargas eléctricas en la puerta flotante, es decir, reduce la retención de datos, en consecuencia, aumenta la probabilidad de error.

Planteamiento del problema

Para dispositivos de almacenamiento de tipo NAND Flash se tiene una probabilidad de error en las celdas donde se almacenan los datos. Esta probabilidad de error depende de:

- Tipo de tecnología del dispositivo (SLC, MLC, TLC).
- Degradación de la memoria, la cual se degrada con los ciclos de escritura y borrado. Esta degradación aumenta la probabilidad de error en los bits.
- Bloques inválidos, los cuales son bloques que el mismo fabricante recomienda no usar.

De lo anterior es indispensable contar con un banco de pruebas que degrade el dispositivo NAND Flash, aplicando una serie de ciclos de escritura y lectura de información para degradarlos, con esto es posible obtener el parámetro BER (Bit Error Rate). El parámetro BER se refiere a la probabilidad de error.

Conocer el BER de los dispositivos NAND Flash permitirá elegir adecuadamente un código de corrección de errores adecuado para futuras aplicaciones. Proponer algún ECC que pueda ser capaz de adaptarse o cambiar su capacidad de corrección de acuerdo a la degradación de la memoria.

Objetivo

Desarrollar un banco de pruebas de degradación de memorias NAND Flash con el fin de degradar la memoria y conocer la relación entre ciclos de borrado/escritura y la probabilidad de error, utilizando un sistema a base de una tarjeta Raspberry Pi® y un Registrador de datos.

Justificación

Debido a que diversos quipos instrumentados de CIDESI cuentan con sistemas de almacenamiento basados en memorias NAND Flash es importante comprender el comportamiento de la probabilidad de error en los bits de almacenamiento a medida que se degrada el dispositivo NAND Flash, entender la probabilidad de error de un dispositivo NAND Flash permitirá seleccionar adecuadamente un código de detección y corrección de errores para futuras aplicaciones, para lograr esto, en este trabajo se desarrolla un banco de pruebas de degradación de memorias NAND Flash, el banco de pruebas está basado en una tarjeta Raspberry Pi® y un Registrador de datos que emplea una memoria NAND Flash MLC Micron® 29F256G08CJABB y un Microcontrolador MSP430F2618 de Texas Instruments®.

Metodología

En la Figura 3 se muestra el diagrama del banco de pruebas que es requerido para este proyecto, para el cual es necesario desarrollar una programa en Python en Raspberry Pi® (c) para generar una interfaz gráfica de usuario, la cual se visualizará con la ayuda de un monitor externo (a) conectado al puerto HDMI de Raspberry Pi®, la interfaz es controlada por el usuario a través de un teclado y un mouse (b) conectados a los puertos USB disponibles en la tarjeta Raspberry Pi®, la interfaz gráfica posibilita la comunicación a través de UART a 115200 baudios con el registrador de datos (d), para esto se utilizan los pines disponibles Rx y Tx del GPIO de Raspberry Pi® (pin 8 y 10 respectivamente). El Registrador de datos emplea una memoria NAND Flash MLC Micron® 29F256G08CJABB (f) y un Microcontrolador MSP430F2618 de Texas Instruments® (e), el microcontrolador se comunicará con el dispositivo NAND Flash a través del protocolo ONFI.

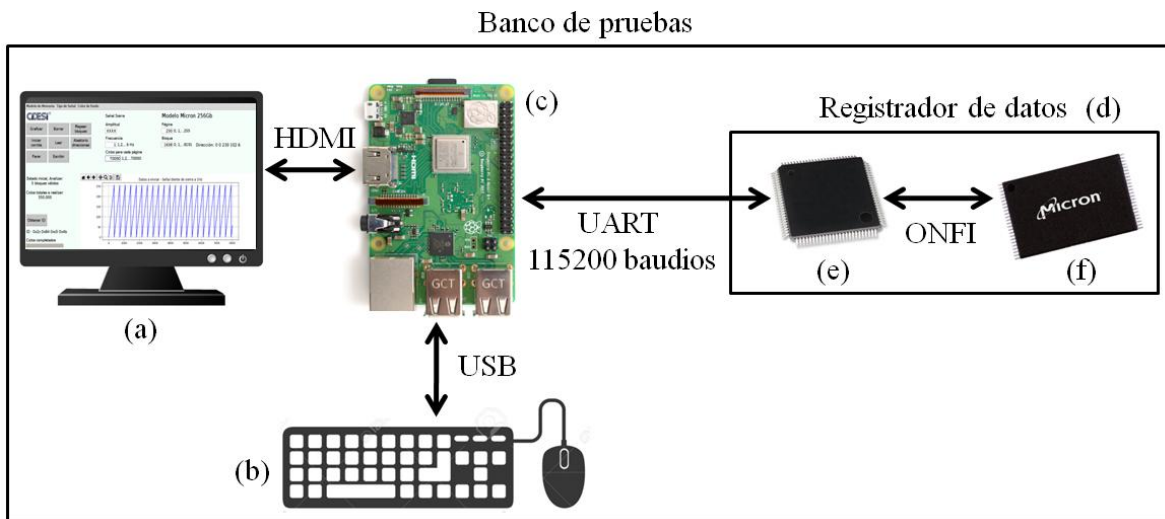


Figura 3 Banco de pruebas de degradación de memorias NAND Flash.

En la Figura 4 se muestran los requerimientos específicos del banco de pruebas de degradación de memorias NAND Flash.

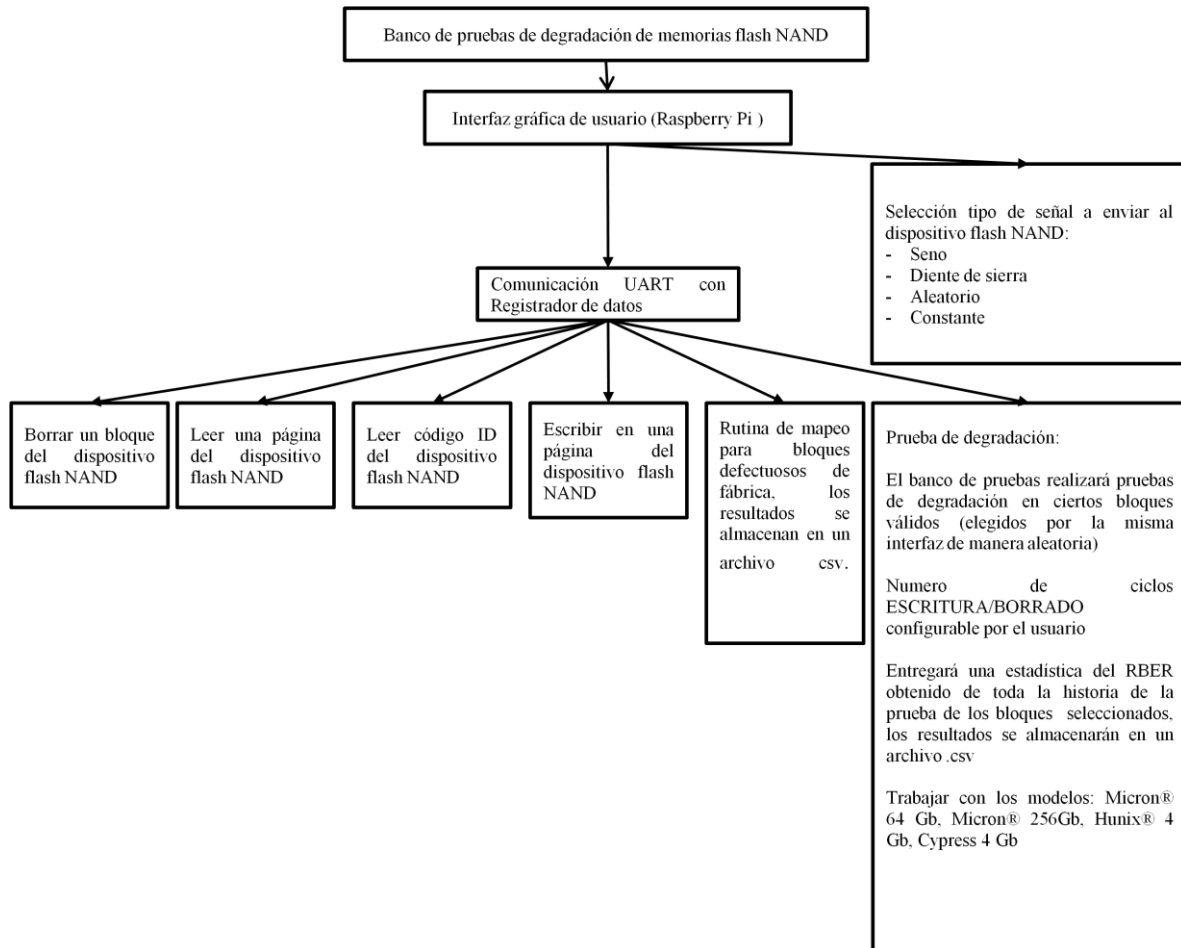


Figura 4 Especificaciones para el banco de pruebas de degradación de memorias NAND Flash.

Marco teórico

Memorias NAND Flash

La memoria NAND Flash es un tipo de memoria no volátil, lo que significa que puede almacenar una carga independientemente que el chip esté o no esté conectado a una fuente de alimentación. Como se mencionó anteriormente los dispositivos NAND Flash pueden tener tecnología SLC (Single Level Cell), MLC (Multi Level Cell), TLC (Triple Level Cell) en la cual es posible almacenar 1, 2, y 3 bit por celda de memoria respectivamente.

Una mayor cantidad de almacenamiento de bits por celda ofrece mayores capacidades de almacenamiento, esto es porque en el mismo espacio físico se almacena una mayor cantidad de información, lo que trae como ventaja un menor costo por unidad de almacenamiento. Por otro lado, una mayor capacidad de almacenamiento de bits por celda ofrece un rendimiento bastante inferior, así también el número de ciclos de ESCRITURA/BORRADO que toleran las celdas se ve fuertemente reducido, es decir, el tiempo de vida útil es menor.

Esto se debe a que el desgaste que ocurre en las celdas afecta en una mayor medida a celdas que almacenan una cantidad mayor de bits por el hecho de que entre mayor tasa de bits almacenados por celda, las lecturas de voltaje deben ser mucho más precisas. Además de que son más lentos los procesos de lectura y escritura, esto se debe a que el proceso de interpretar cual es un cero y cual es un uno es más lento, siendo las memorias NAND Flash con tecnología SLC las más veloces y duraderas.

Los bloques de un dispositivo NAND Flash son las entidades más pequeñas que pueden ser borradas, la operación BORRADO establece todos los bits de un bloque en uno (todos los bytes en FFh). La operación ESCRITURA es necesaria para cambiar los bits borrados de 1 a 0, para ésta operación la entidad más pequeña que puede ser escrita es un byte (MicronTechnologies®, 2006).

Los diferentes dispositivos NAND Flash están disponibles en diferentes configuraciones y densidades. El registrador de datos del banco de pruebas está basado en un microcontrolador MSP430 y una memoria NAND Flash modelo MT29F256G08CJABB, a continuación se muestran las características del dispositivo NAND Flash utilizado:

- MT – Tecnología Micron®
- 29F – Dispositivo NAND Flash
- 256G – Densidad de 256GB
- 08 – Ancho de datos de 8 bits
- C – Tecnología MLC (2 bits por celda)
- J – Encapsulado con dos dispositivos NAND Flash embebidos, es decir doble señal \overline{CE} y $\overline{R/B}$ con los pines I/O comunes para ambos
- A – Voltaje de operación de 2.7 – 3.6 Volts
- BB – Interfaz Síncrona/Asíncrona

Las memorias de tipo NAND Flash proporciona una disposición de pines consistente para todos los dispositivos y densidades, esta disposición de pines permite a los diseñadores utilizar densidades más bajas y migrar a densidades más altas sin ningún cambio de hardware en la placa de circuito impreso (Micron®, 2020). En la Tabla 1 se muestra la descripción de las señales de un dispositivo NAND Flash de 8 bits de bus de datos. Para un dispositivo con una matriz NAND Flash se cuenta con una señal \overline{CE} y $\overline{R/B}$, se compone de un total de 15 pines como se puede apreciar en la Figura 5.

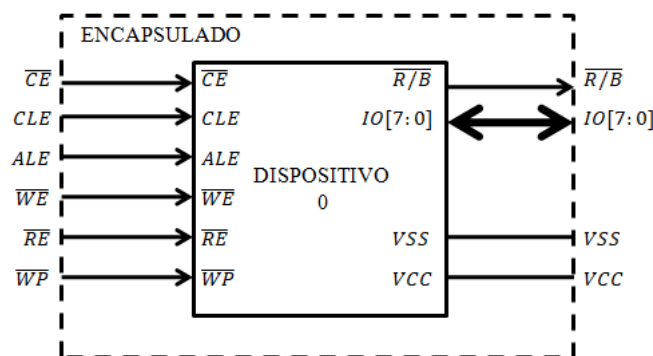


Figura 5 Organización del dispositivo con encapsulado de matriz única. (Micron®, 2020).

Tabla 1 Descripción de señales de un dispositivo NAND Flash. (Micron®, 2020).

Símbolo	Señal	Descripción
ALE	Address latch enable	Cuando ALE se encuentra en alto, los registros de dirección se bloquean en NAND Flash en el flanco ascendente de la señal \overline{WE} .
\overline{CE}	Chip enable	Permite activar el chip, en otras palabras, disponer de los pines de entrada/salida, si no se activa la señal CE , el dispositivo NAND Flash permanece en modo de espera y no responde a ningún control de señales.
CLE	Command latch enable	Cuando CLE se encuentra en alto, los comandos se bloquean en el registro de comando NAND Flash en el flanco ascendente de la señal \overline{WE} .
$\overline{R/B}$	Read/Busy	Si el dispositivo NAND Flash está ocupado con una operación de borrado, programación o lectura, la señal $\overline{R/B}$ se afirmará en estado bajo. La señal $\overline{R/B}$ es de drenaje abierto y requiere una resistencia pull-up.
\overline{RE}	Read enable	La señal \overline{RE} habilita los búferes de datos de salida.
\overline{WE}	Write enable	La señal \overline{WE} es responsable de marcar (como reloj) los datos, la dirección o los comandos en el dispositivo NAND Flash

Los encapsulados con más de una matriz NAND Flash cuentan con más de una señal \overline{CE} y $\overline{R/B}$, en este caso para habilitar la matriz NAND Flash a la que se desea acceder se hace por medio de la activación de la señal \overline{CE} (Chip Enable). En la Figura 6 se muestra un esquema de un encapsulado con dos matrices NAND Flash, esta interfaz de hardware cuenta con una disposición de pines estándar con la ventaja de poseer una mayor capacidad de almacenamiento.

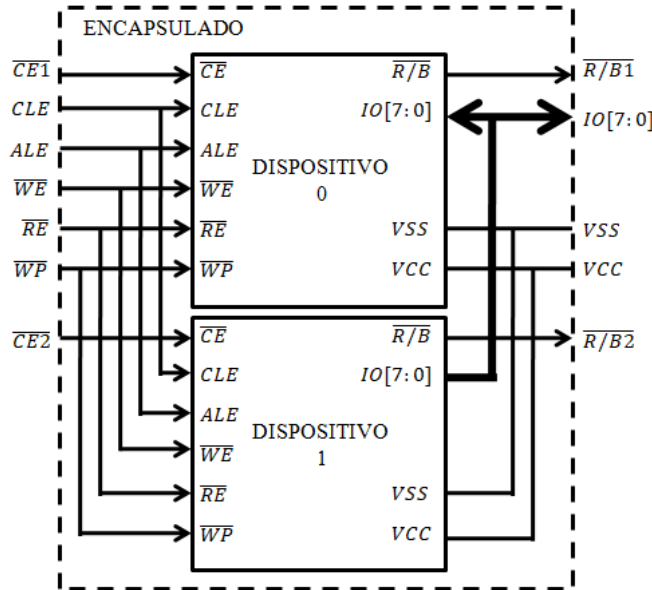


Figura 6 Organización del dispositivo con encapsulado de dos matrices. (Micron®, 2020).

La memoria NAND Flash es muy parecido a una unidad de disco duro, ya que ambos dispositivos son adecuados para el almacenamiento de datos secuenciales, como imágenes, videos, audio, etc. Además, como una unidad de disco duro, una memoria de tipo NAND Flash tiene bloques defectuosos de fábrica, de igual modo ambos dispositivos requieren de un código de corrección de errores (ECC) para mantener la integridad de los datos.

El dispositivo NAND Flash Micron® de 256Gb MT29F256G08CJABB está organizado con 16,384 bloques, con 256 páginas por bloque, cada página cuenta con 8,640 bytes para almacenamiento, que consta de un área de datos de 8192 bytes y un área de reserva de 448 bytes, dicha área de reserva se usa normalmente para ECC, nivelación de desgaste y otras funciones de software generales, aunque físicamente es igual que el resto de la página. La Figura 7 muestra la arquitectura del dispositivo NAND Flash Micron® de 256Gb. Para este modelo el tiempo de lectura de una página completa es de máximo 75µs, el tiempo de programación es de de una página completa es de 1,300µs y el tiempo de borrado de un bloques es de máximo 3.8 ms, tiene una resistencia de 3,000 ciclos de PROGRAMACIÓN/BORRADO (Micron®, 2020).

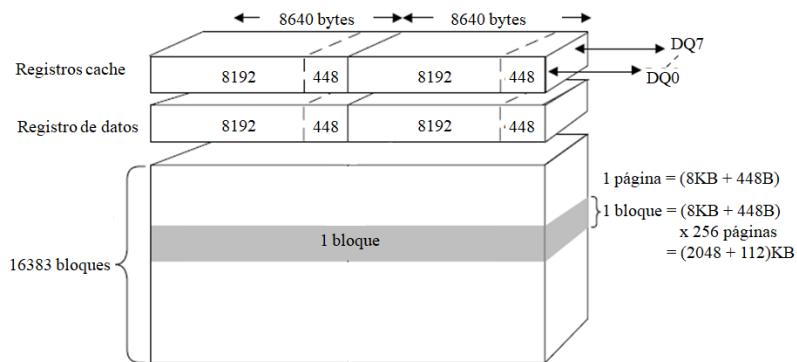


Figura 7 Arquitectura del dispositivo NAND Flash Micron® de 256 Gb modelo 29F256G08CJABB, organizado con bloques (Micron®, 2020).

La matriz de direccionamiento para el dispositivo NAND Flash Micron® MT29F256G08CJABB, se muestra en la Tabla 2. El primer y segundo ciclo de dirección especifican la dirección de la columna, que especifica el byte de inicio dentro de la página, PA [7:0] especifica la dirección de la página dentro del bloque y BA [19:15] especifica la dirección del bloque. Mientras que la dirección completa de 5 bytes es requerida para la mayoría de las operaciones, PROGRAMAR UNA PÁGINA y LEER UNA PÁGINA de la memoria NAND Flash solo son necesarios el primer y segundo byte (o ciclos). La operación de borrado de un bloque requiere solo los tres bytes menos significativos (tercero, cuarto y quinto) para seleccionar el bloque.

Tabla 2 Esquema de direccionamiento del dispositivo NAND Flash MLC de 256 Gb. (Micron®, 2020).

Ciclos	I/O7	I/O6	I/O5	I/O4	I/O3	I/O2	I/O1	I/O0
Primero	CA7	CA6	CA5	CA4	CA3	CA2	CA1	CA0
Segundo	BAJO	BAJO	CA13	CA12	CA11	CA10	CA9	CA8
Tercero	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
Cuarto	BA15	BA14	BA13	BA12	BA11	BA10	BA9	BA8
Quinto	BAJO	BAJO	BAJO	LA0	BA19	BA18	BA17	BA16

Raspberry Pi 3®

Raspberry Pi® es una tarjeta electrónica de un tamaño muy reducido, la cual se considera como un ordenador completo, ya que cuenta con periféricos de entrada y salida (USB, HDMI, JACK AUDIO, ETHERNET). En la Figura 8 se puede observar la placa Raspberry Pi® 3 modelo B, utilizada para este proyecto.

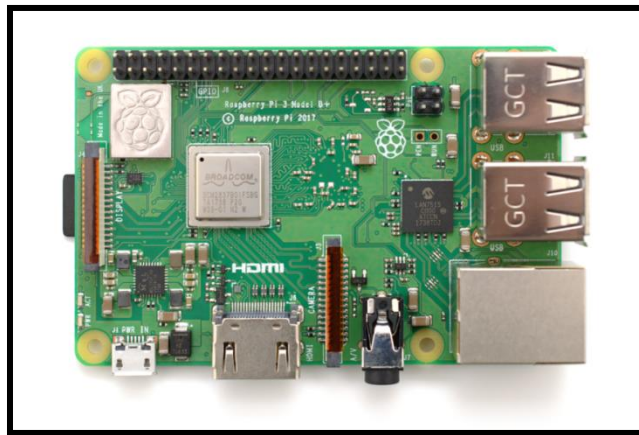


Figura 8 Tarjeta electrónica Raspberry Pi® modelo B (RaspberryPi®, 2020).

La tarjeta Raspberry Pi® utilizada para éste proyecto cuenta con las siguientes características tecnológicas: CPU ARMv8 a 1.2GHz de 64-bit, 1 GB de RAM, WiFi 802.11 b/g/n, Bluetooth 4.1, Puerto HDMI, 4 puertos USB, Slot para Micro SD y un puerto Ethernet. Cabe mencionar que éste dispositivo no cuenta con una memoria interna, esto es debido a que la placa cuenta con una ranura para una tarjeta Micro SD en la cual se instala el sistema operativo, ya que ésta tarjeta Micro SD será el principal medio de almacenamiento, es importante contar con una tarjeta Micro SD de clase 10 o superior con una buena capacidad de almacenamiento. El puerto micro USB que tiene Raspberry Pi® 3 se utiliza para alimentar la tarjeta, una vez que Raspberry Pi® es alimentada ésta se enciende automáticamente, el puerto HDMI se utiliza para obtener la imagen del escritorio en un monitor externo, los cuatro puertos de USB se utilizan para conectar dispositivos periféricos como teclado y ratón.

GPIO de Raspberry Pi®.

Una característica muy poderosa de Raspberry Pi® es que dispone de una fila de pines GPIO (Entrada/Salida de uso general) a lo largo del borde superior de la placa. Los pines GPIO cuentan con una gran serie de funcionalidades ya que se pueden utilizar como pines digitales de entrada o salida y algunos específicos para establecer protocolos de comunicación. En la Figura 9 se muestra la configuración de los pines GPIO con los que cuenta la tarjeta Raspberry Pi® 3 (RaspberryPi®, 2020).

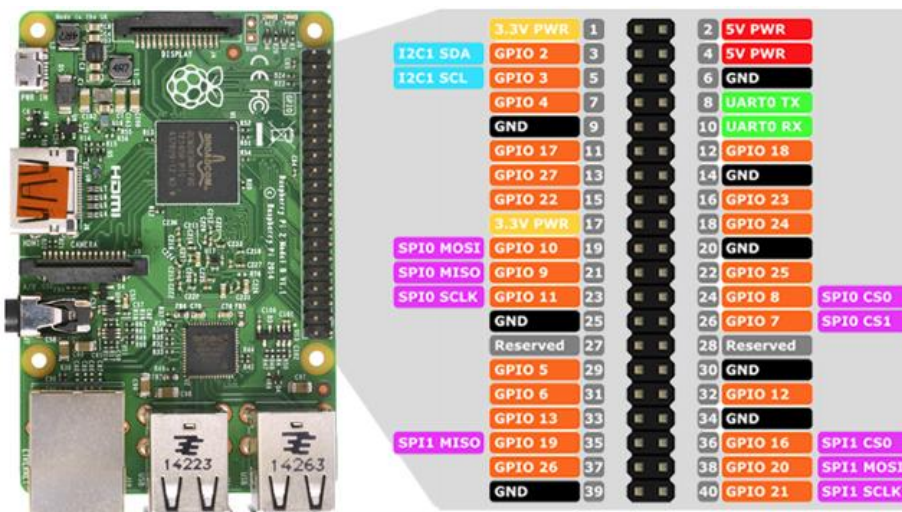


Figura 9 GPIO Raspberry Pi® modelo B. (RaspberryPi®, 2020).

Sistema operativo Raspberry Pi OS.

Raspberry Pi OS® (antes Raspbian®) es el sistema operativo por excelencia para la tarjeta Raspberry Pi®, el sistema operativo está basado en Linux y es de licencia abierta, para instalar este sistema operativo en Raspberry Pi® es necesario instalar “Raspberry Pi Imager” (ver Figura 10) en la computadora, con este software se instala el sistema operativo Raspberry Pi OS y se puede descargar desde: <https://www.raspberrypi.org/software/operating-systems/#raspberrypi-os-32-bit>, la herramienta Raspberry Pi Imager® funciona como herramienta para instalar el sistema operativo en la tarjeta Micro SD.

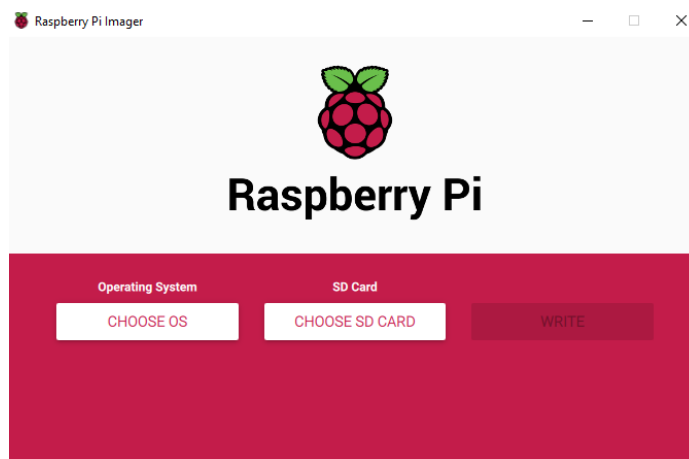


Figura 10 Raspberry Pi Imager.

Python con Raspberry Pi®.

Por defecto, Python viene instalado en el sistema operativo Raspberry Pi OS®. Al utilizar Python en la tarjeta Raspberry Pi® se tiene la ventaja de poder utilizar los pines GPIO de la tarjeta, con esto es posible comunicar el mundo digital con el mundo físico mediante electrónica y programación. Python es un lenguaje de programación de propósito general, esto quiere decir que es posible desarrollar una gran cantidad de aplicaciones en diferentes sectores, por ejemplo, inteligencia artificial, desarrollo de sitios web, software multiplataforma, desarrollo de videojuegos, aplicaciones de escritorio, tareas de ciencia de datos, etc. Esto es porque Python no tiene un enfoque definido por lo cual es posible crear distintas aplicaciones. Este lenguaje de programación usa un tipado dinámico porque una misma variable puede tomar valores de distinto tipo en distintos momentos. Python se ha convertido en uno de los lenguajes de programación más usados a nivel mundial según Stack Overflow. Al ser multiplataforma se puede ejecutar en distintos sistemas operativos, incluyendo el sistema operativo de Raspberry Pi®. Python se considera como un lenguaje de alto nivel, ya que posee instrucciones o sentencias que son escritas con palabras similares a los lenguajes humanos, además de que posee una de las sintaxis más amigables dentro del mundo de la programación lo que facilita su entendimiento (CodigoFácil, 2012).

Actualizar la versión de Raspbian® en Raspberry Pi®.

Para realizar una actualización al sistema operativo de la tarjeta Raspberry Pi® Es necesario ejecutar una serie de comandos desde la consola de comandos, esto ya que el mecanismo de ejecución del software para la actualización del sistema está integrado en el mismo sistema operativo Raspbian®. En la Figura 11 se muestra la consola de comandos de Raspberry Pi®.

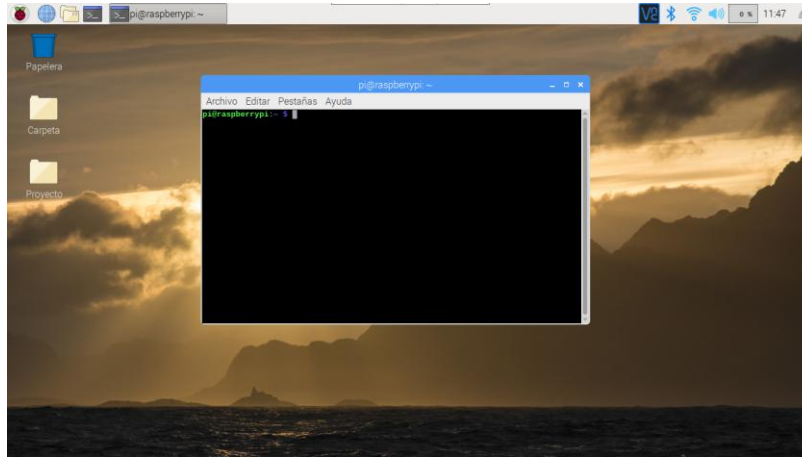


Figura 11 Consola de comandos de Raspberry Pi®.

Es necesario ejecutar cada uno de los siguientes comandos en la consola virtual de Raspberry Pi®:

- `sudo apt-get -y update`
- `sudo apt-get -y dist-upgrade`
- `sudo apt-get-y autoremove`
- `sudo apt-get autoclean`
- `sudo reboot`

Con esto se realiza la actualización de todos los paquetes del sistema, después se limpia el repositorio de paquetes local y, finalmente, se reinicia la Raspberry Pi®. Es muy recomendable realizar esta actualización del sistema a la Raspberry Pi® si se nota que el sistema en general se vuelve lento.

Registrador de datos

El registrador de datos consiste de un microcontrolador MSP430 de Texas Instruments® y una memoria NAND Flash Micron ® modelo MT29F256G08CJABB la cual tiene una capacidad de almacenamiento de 256 Gb, la comunicación entre el microcontrolador y la memoria flash NAND se realiza mediante el protocolo ONFI 2.0.

El microcontrolador tiene un módulo de acceso directo a la memoria DMA que permite ejecutar instrucciones del CPU en paralelo a la transferencia de información que en nuestro caso es vía UART a 115200 baudios. El DMA permite que la CPU realice las instrucciones mientras que el controlador DMA llena un vector de datos que es el que se escribe en la memoria, la **¡Error! No se encuentra el origen de la referencia.** muestra el esquemático del registrador de datos que consta de la memoria NAND Flash (a), el micro controlador MSP430F2618 (B), pines de comunicación UART (C), pines de programación (D) y alimentación (E).

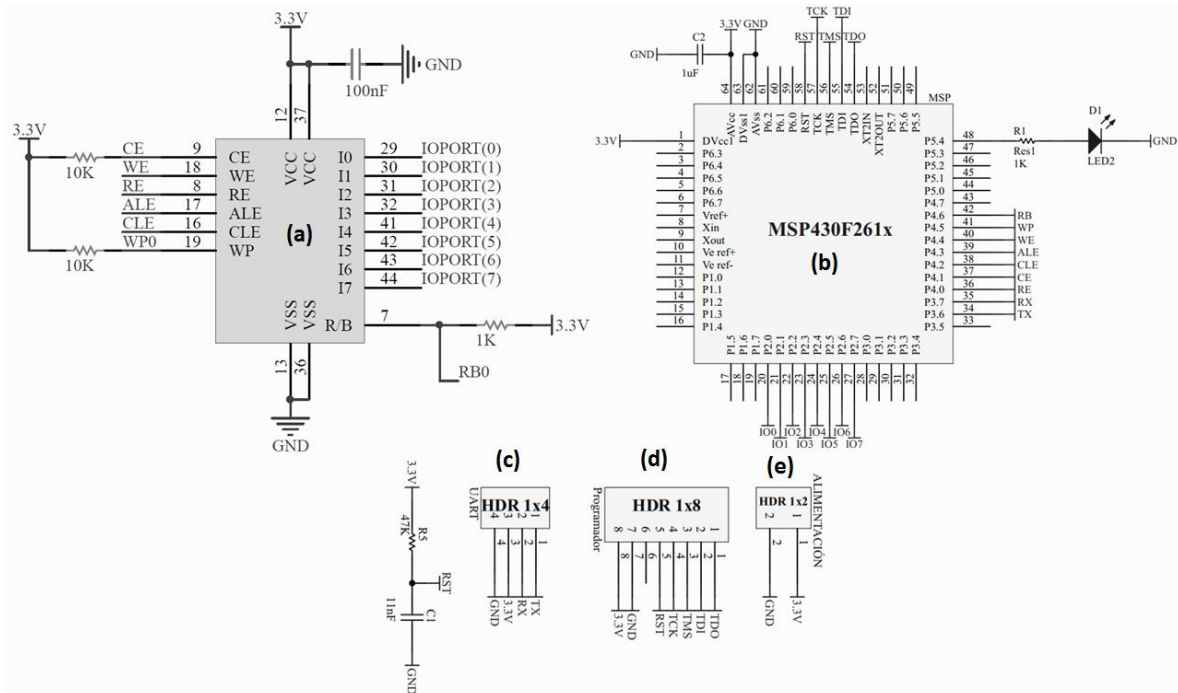


Figura 12 Esquemático PCB del Registrador de datos.

GRAFCET

Para la realización del programa con Python en Raspberry Pi® se utilizará el método de diagramación de flujo lógico GRAFCET (Gráfico Funcional de Control de Etapas y Transiciones) ya que es un sistema muy visual y sencillo de implementar. El GRAFCET se compone de varios elementos, estos elementos se distribuyen en etapas, acciones asociadas y transiciones, todo a través de procesos secuenciales. En un sistema utilizando GRAFCET siempre se debe arrancar con una etapa inicial. En seguida se pueden agregar etapas normales, las acciones asociadas a las etapas es la acción que se ejecutará, las transiciones son el paso evolutivo de la etapa previa a la siguiente. Las transiciones son segmentos de condiciones que el sistema debe superar para alcanzar la siguiente etapa. Estas condiciones pueden ser, comparaciones booleanas, comparaciones aritméticas, etc. Para establecer el flujo, se utilizan líneas de enlace, estas pueden ser verticales u horizontales y dependerá directamente de los elementos anteriores. En la Figura 13 se muestra un ejemplo de un diagrama de tipo GRAFCET para ilustrar su funcionamiento.

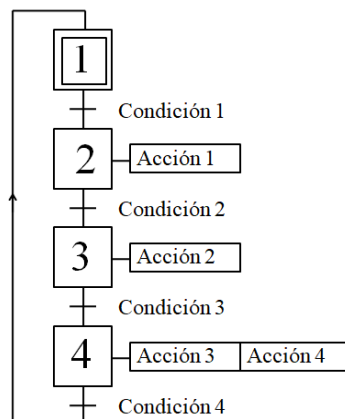


Figura 13 Diagrama tipo GRAFCET básico.

Las etapas se muestran con las formas de cuadrados y están enumeradas, en donde, la etapa 1 se designa como la etapa inicial, es decir, donde empezará el programa, una etapa puede ejecutar más de una acción como se puede mostrar en la etapa 4.

Desarrollo del Banco de Pruebas de degradación de memorias NAND Flash

A continuación se muestran las librerías utilizadas en el código de Python para la realización de la aplicación de escritorio del banco de pruebas de degradación de memorias flash NAND:

- Tkinter: la librería sirve para generar interfaces gráficas para usuario, para la realización de éste proyecto se hace uso de la librería Tkinter por su simplicidad de uso y porque es una librería gratuita, así mismo, Tkinter ofrece una muy amplia gama de herramientas para una gran variedad de tareas (botones, etiquetas, textos, gráficas, barras de progreso, leds indicadores, etc.), así como desplegar mensajes de aviso, error y advertencia para el usuario (Alvarez, 2015).
- Time: permite obtener, manipular y dar formato a fechas y horas que se pueden obtener del sistema, así como hacer cálculos con éstos valores (PythonSoftwareFundation, 2020).
- Pil: permite la edición de imágenes directamente desde Python, dicha librería soporta una gran variedad de formatos (David, 2019).
- Numpy: es un paquete para la computación científica que permite la manipulación de datos desde Python, ésta librería permite trabajar con matrices y vectores, lo cual es muy importante para este proyecto (EMECATRONICOS, 2019).
- Matplotlib: permite generar gráficas a partir de datos contenidos en vectores, para esto es necesario que los vectores estén generados en la extensión matemática de Numpy, dichas gráficas se pueden visualizar en la ventana generada por Tkinter (DonPython, 2018).
- Random: ésta librería permite generar números aleatorios (Piogram, 2020).
- Serial: para éste proyecto es necesario establecer una comunicación UART a 115200 baudios, entre la tarjeta Raspberry Pi® 3 y el Registrador de datos, esto para compartir información entre ambos sistemas, para realizar esto es necesario disponer de la librería PySerial para Python, ésta librería permite una forma muy sencilla de emplear el puerto serie disponible en el GPIO de la tarjeta Raspberry

Pi®, para Raspberry Pi® se hace referencia al nombre del puerto serie como ttyS0 como se indica en el manual de usuario de Raspberry Pi® 3 (MasterCodeOnline, 2017).

- Threading: el módulo permite gestionar operaciones concurrentes dentro de un proceso, dichas operaciones se denominan “Hilos”, al hacer uso de “Hilos” permite que un programa ejecute múltiples operaciones simultáneamente en el mismo espacio de proceso, lo cual es muy útil para reducir el tiempo de ciclo de la aplicación, ejecutando algunas porciones de código que son independientes de forma simultánea (Errodriger, 2018).
- Os: el módulo os provee de varios métodos para trabajar con funcionalidades del sistema operativo, dicha librería se utiliza para conocer información y crear directorios en la memoria de la tarjeta Raspberry Pi® 3 (Ávila, 2014).
- CSV: el formato csv es un formato de importación y exportación de información muy utilizado en las bases de datos y hojas de cálculo, con la librería CSV Python es capaz de escribir y leer información en dicho formato (CoreyShafer, 2017).

En la Figura 14 se puede observar la conexión física del banco de pruebas



Figura 14 Conexiones físicas del banco de pruebas.

Interfaz gráfica de usuario

La interfaz gráfica de usuario desarrollada con Python en Raspberry Pi® (ver Figura 15) es una herramienta que posibilita la comunicación entre el usuario y el registrador de datos. Es posible trabajar con uno de los 4 modelos diferentes de memorias NAND Flash, para esto es necesario seleccionar el modelo con el que desea trabajar desde la interfaz (ver Figura 16) y conectar físicamente el dispositivo NAND Flash correspondiente.

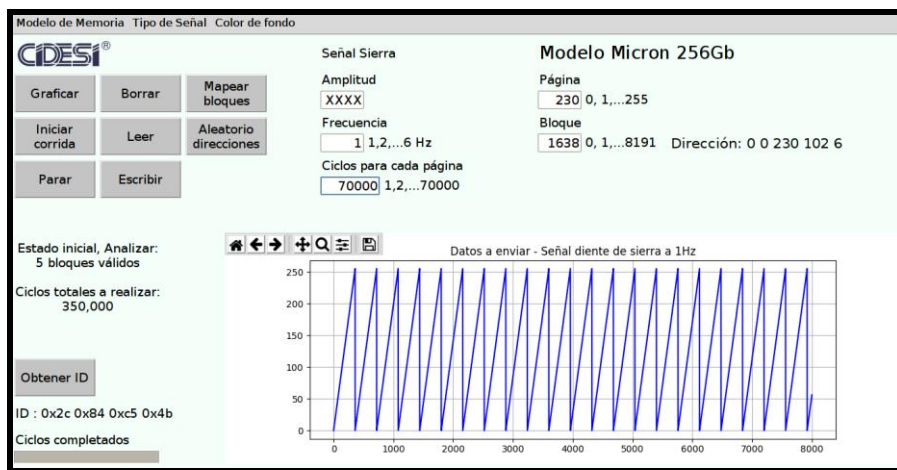


Figura 15 Interfaz gráfica de usuario en Raspberry Pi®.

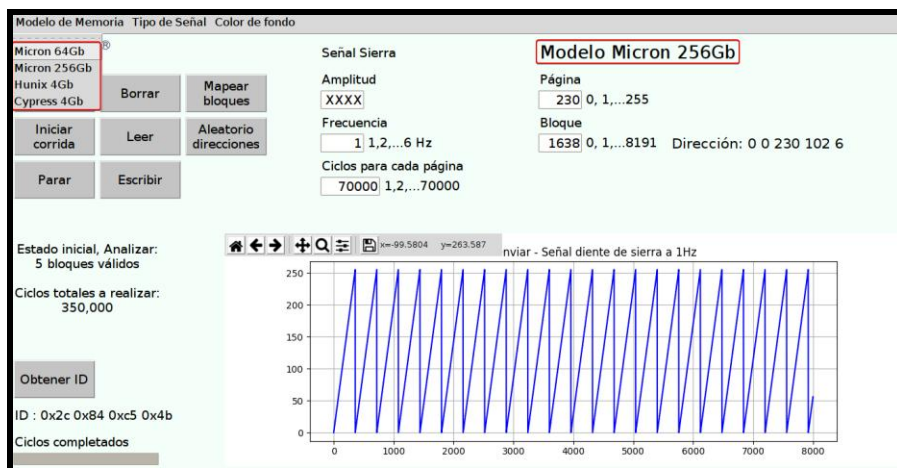


Figura 16 Selección de modelo de memoria NAND Flash.

Selección del tipo de señal que se desea transmitir al dispositivo NAND Flash.

Permite al usuario seleccionar el tipo de señal (ver Figura 17) que será posible escribir a una página de la memoria NAND Flash, es posible enviar cuatro tipos de señales: senoidal, diente de sierra, aleatoria y constante. El número de puntos que se generan corresponde al número de bytes que se pueden escribir en una página del dispositivo NAND Flash según el modelo seleccionado en la interfaz.

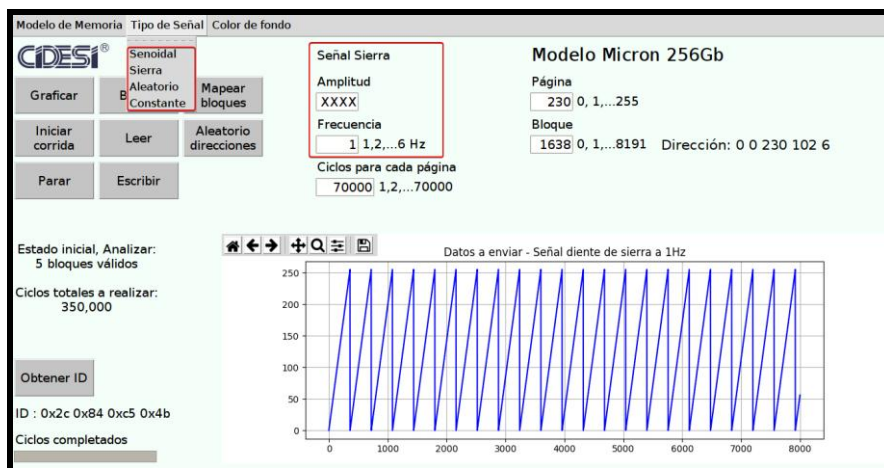


Figura 17 Selección de señal a transmitir al dispositivo NAND Flash.

Graficar datos a transmitir al dispositivo NAND Flash.

Para la señal seno y diente de sierra, el usuario puede configurar la frecuencia en el cuadro de entrada "Frecuencia" (ver Figura 19), la frecuencia para ésta señal tiene como límites de 1 a 6 Hz, la amplitud de la señal será igual a 255. En la Figura 19 se puede observar una señal de tipo seno de 2 Hz, de igual manera en la Figura 20 se muestra una señal de tipo diente de sierra de 2 Hz. El diagrama GRAFCET programado con Python en Raspberry Pi® de la Figura 18 muestra el proceso para graficar una señal en la interfaz gráfica de Raspberry.

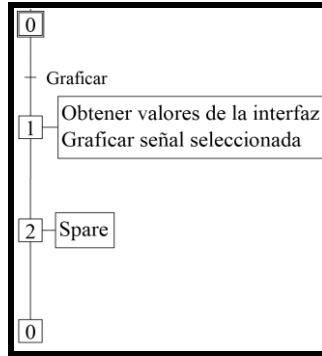


Figura 18 Diagrama de control para graficar datos a transmitir al dispositivo NAND Flash.

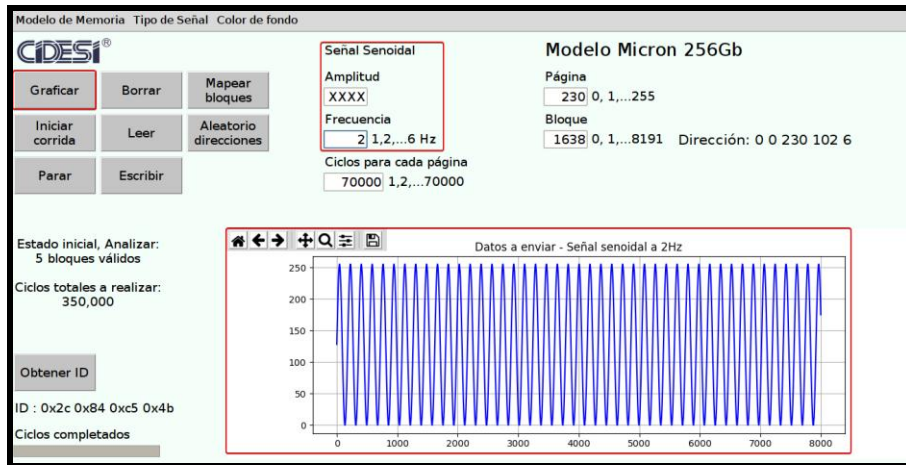


Figura 19 Grafica de una señal seno de 2 Hz.

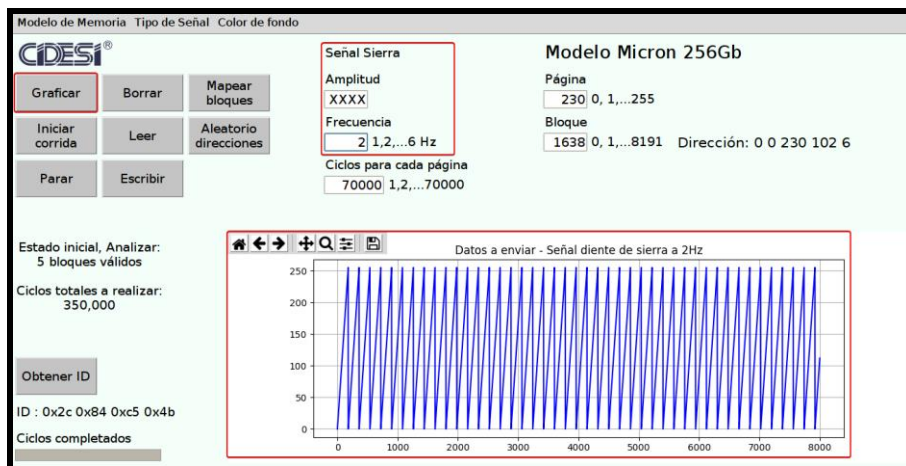


Figura 20 Grafica de una señal diente de sierra 2 Hz.

Para la señal aleatoria no es posible configurar la amplitud ni la frecuencia. La Figura 21 muestra la gráfica de una señal aleatoria de 8000 elementos.

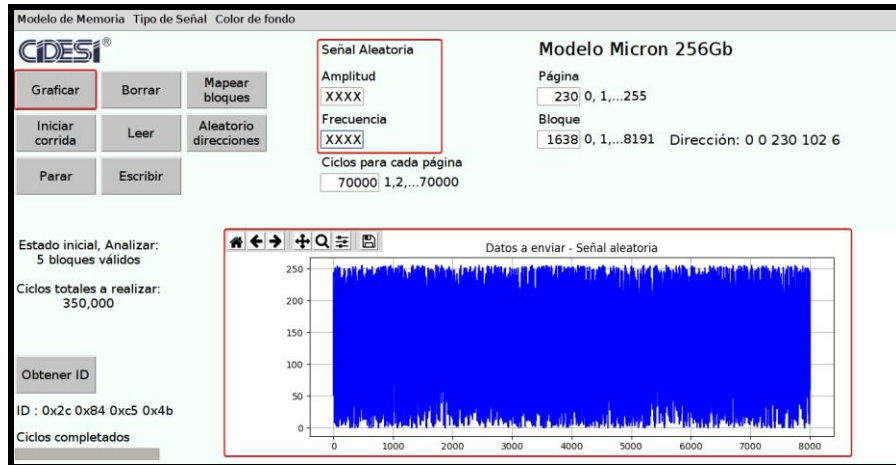


Figura 21 Grafica de una señal aleatoria.

Para la señal constante es posible configurar la amplitud de 0 a 255, La Figura 22 muestra la gráfica de una señal constante de valor 126 de 8000 elementos.

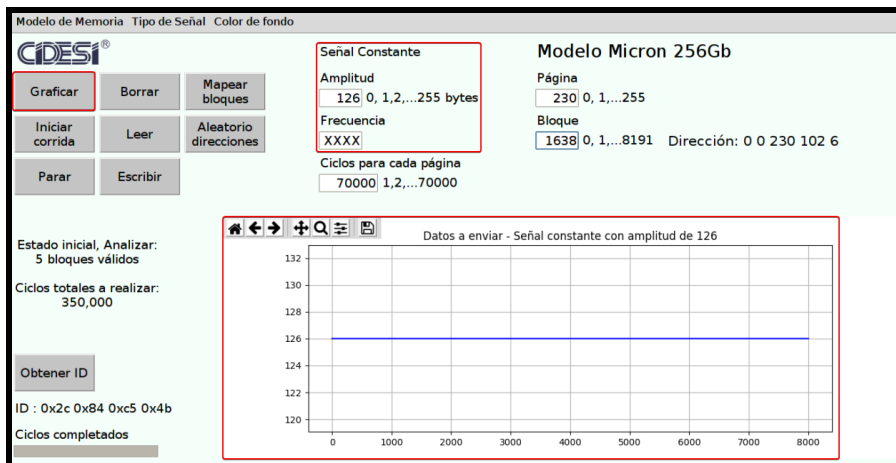


Figura 22 Grafica de una señal constante con amplitud de 126.

Cambiar color de fondo de la interfaz.

Permite al usuario seleccionar el color de fondo (ver Figura 23) que puede tener toda la interfaz, estos pueden ser: “Mint cream”, “Orange red”, “Coral”, “Tan” y “Sky blue”, en la Figura 24 se muestra la interfaz gráfica con un color de fondo “Coral”.

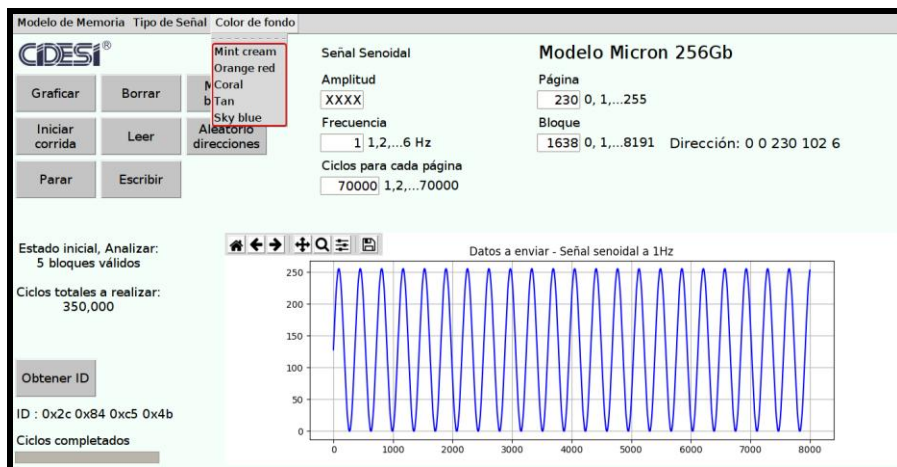


Figura 23 Selección de color de fondo para la interfaz gráfica.

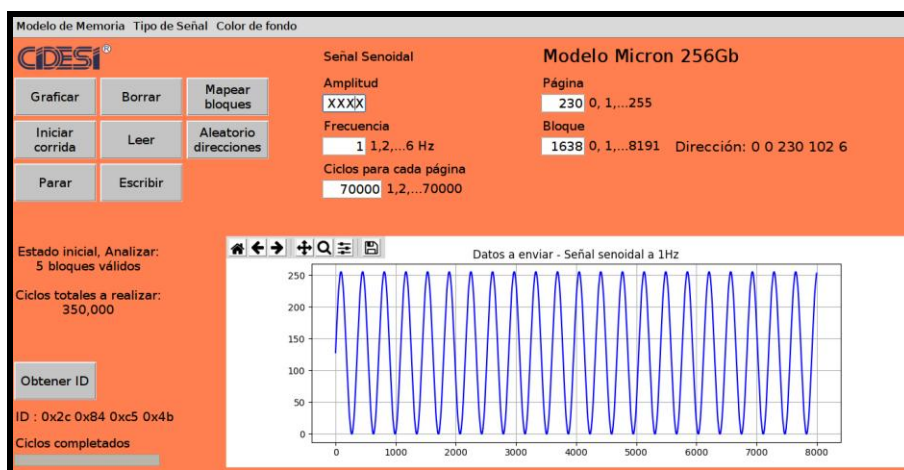


Figura 24 Selección de color de fondo “Coral” para la interfaz gráfica.

Lectura de ID del dispositivo NAND Flash.

En la Figura 25 se puede observar la operación de lectura de ID del dispositivo NAND Flash Micron® modelo MT29F256G08CJABB desde la interfaz gráfica de Raspberry Pi®. El diagrama GRAFCET programado con Python en Raspberry Pi® de la Figura 26 muestra el proceso para obtener el ID del dispositivo NAND Flash conectado.

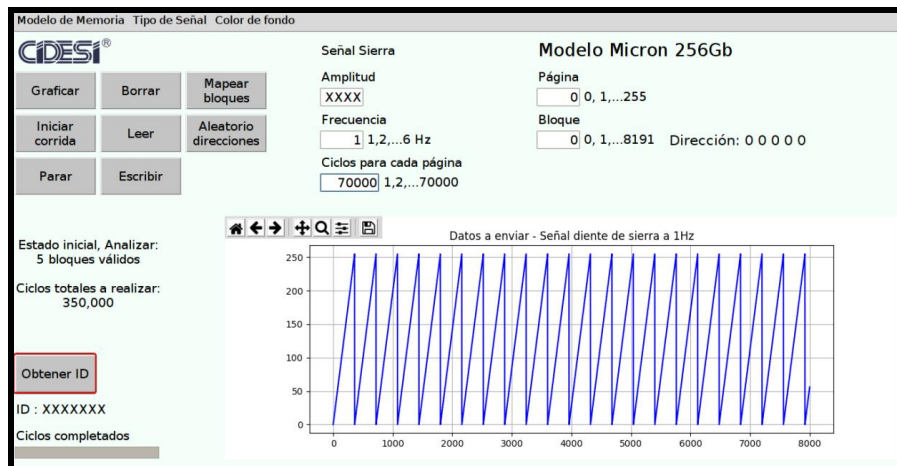


Figura 25 Obtener ID del dispositivo NAND Flash mediante la interfaz gráfica de usuario.

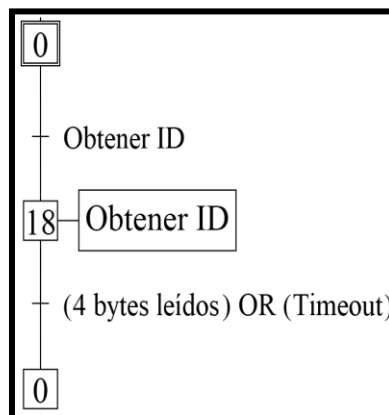


Figura 26 Diagrama de control para la lectura de ID del dispositivo NAND Flash

La operación LEER ID se utiliza para leer los códigos de identificación programados por el fabricante en el dispositivo NAND Flash. Para obtener el ID del dispositivo, Raspberry Pi® envía a través de UART el carácter ASCII “I” al Registrador de datos, el envío del carácter se realiza en la etapa 18 del programa (ver Figura 26). El Microcontrolador MSP430F2618 de Texas Instruments® recibe través de UART el carácter ASCII “I”. Para el envío de los comandos al dispositivo NAND Flash, el microcontrolador ejecuta las señales de control de acuerdo a los tiempos de la Figura 27, los parámetros mínimos y máximos de los tiempos se presentan en la Tabla 3.

Tabla 3 Parámetros de tiempos mínimos y máximos para el envío de comandos al dispositivo NAND Flash (BROADCOM, 2008).

Símbolo	3.3V		Unidad
	Mínimo	Máximo	
tCLS	15	-	ns
tCLH	10	-	ns
tCS	25	-	ns
tCH	10	-	ns
tWP	17	-	ns
tALS	15	-	ns
tALH	10	-	ns
tDS	15	-	ns
tDH	5	-	ns

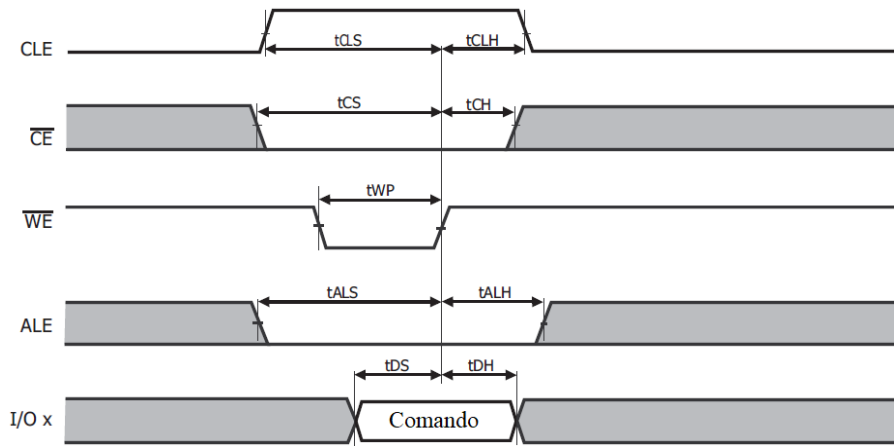


Figura 27 Comportamiento de señales para enviar un comando al dispositivo NAND Flash (BROADCOM, 2008).

Para leer el código ID del dispositivo NAND Flash, el microcontrolador ejecuta las señales de control de acuerdo a los tiempos de la Figura 28, los parámetros mínimos y máximos de los tiempos se presentan en la Tabla 4.

Tabla 4 Parámetros de tiempos mínimos y máximos para leer el ID dispositivo NAND Flash (BROADCOM, 2008).

Símbolo	3.3V		Unidad
	Mínimo	Máximo	
tAR	10	-	ns
tREA	-	20	ns

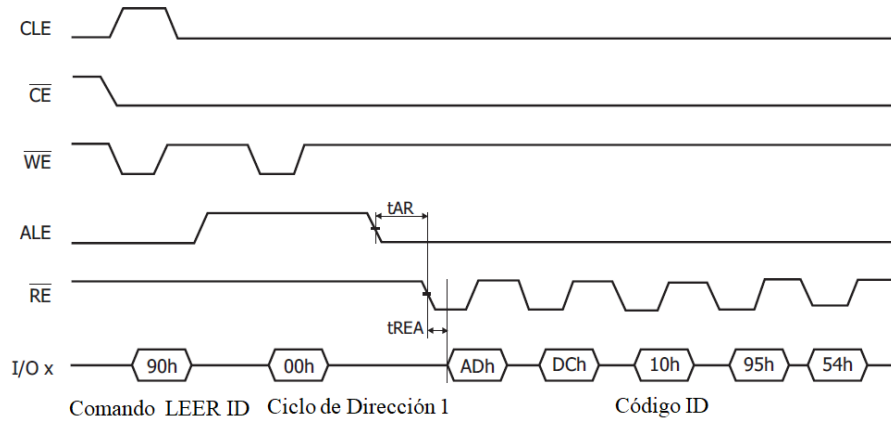


Figura 28 Comportamiento de señales para leer el ID del dispositivo NAND Flash (BROADCOM, 2008).

Una vez obtenido el código ID del dispositivo NAND Flash, el microcontrolador MSP430 envía el código ID a Raspberry Pi a través de UART. En la Figura 29 se puede observar el código ID obtenido del dispositivo NAND Flash Micron® modelo MT29F256G08CJABB.

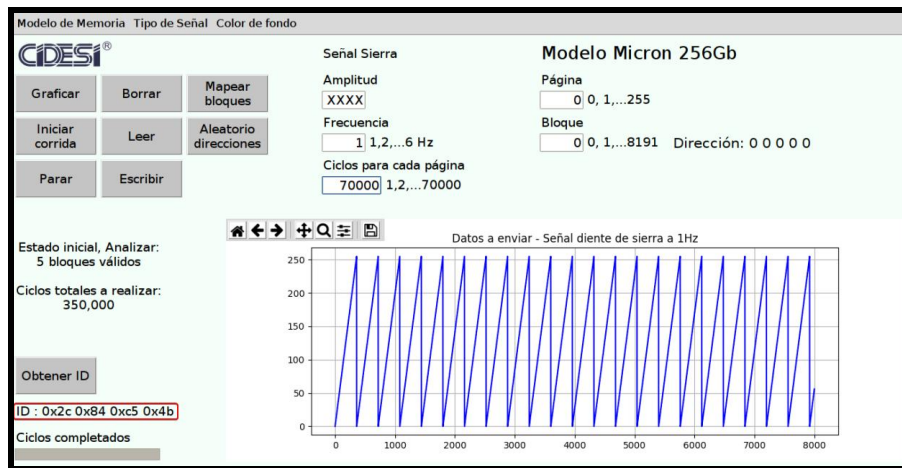


Figura 29 Código ID del dispositivo NAND Flash MT29F256G08CJABB.

Borrar un bloque del dispositivo NAND Flash.

Para realizar esta operación a un bloque del dispositivo NAND Flash es necesario especificar el número del bloque en el cuadro de entrada “Bloque”, la acción empieza cuando se presiona el botón “Borrar” (ver Figura 30). La acción de éste botón solo se realizará cuando la interfaz no se encuentre ocupada realizando alguna otra operación.

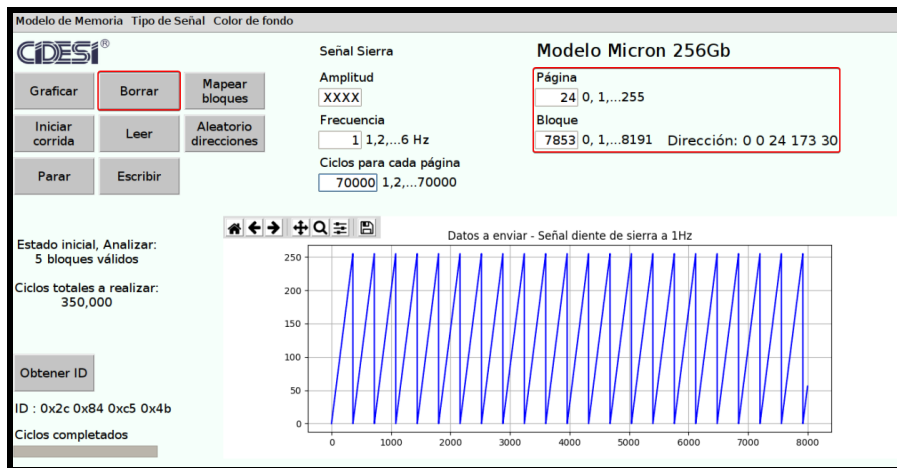


Figura 30 Borrar un bloque del dispositivo NAND Flash mediante la interfaz gráfica de usuario.

Para realizar el borrado de una localidad en la memoria NAND Flash se debe realizar por bloque, que es la mínima unidad de la memoria NAND Flash que puede ser borrada. La operación BORRAR afecta a todas las páginas en el bloque. La operación pone todos los bits del bloque en valor ‘1’.

El diagrama GRAFCET programado con Python en Raspberry Pi® de la Figura 31 muestra el proceso para borrar un bloque del dispositivo NAND Flash conectado.

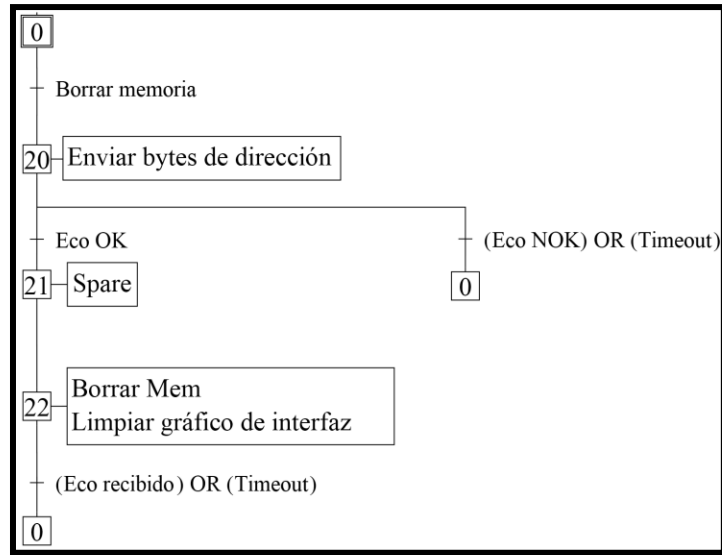


Figura 31 Diagrama de control para borrar un bloque del dispositivo NAND Flash.

Para realizar la operación de borrar un bloque Raspberry Pi® envía a través de UART el carácter ASCII “D” al Registrador de datos. Dicha acción se realiza en la etapa 20 del programa (ver Figura 32), al enviarse el carácter “D” la interfaz espera un eco, esto indica que el Registrador de datos recibió el carácter, y está preparado para recibir los bytes de dirección (número de bloque), seguido de esto la interfaz envía los 5 bytes de dirección correspondientes al bloque seleccionado en la interfaz. Para enviar los bytes de dirección al dispositivo NAND Flash el microcontrolador ejecuta las señales de control de acuerdo a los tiempos de la Figura 32, los parámetros mínimos y máximos de los tiempos se presentan en la Tabla 5.

Tabla 5 Parámetros de tiempos mínimos y máximos para el envío de los bytes de dirección al dispositivo NAND Flash (BROADCOM, 2008).

Símbolo	3.3V		Unidad
	Mínimo	Máximo	
tCLS	15	-	ns
tCS	25	-	ns
tWC	35	-	ns
tWP	17	-	ns
tALS	15	-	ns
tDS	15	-	ns
tWH	15	-	ns
tDH	15	-	ns

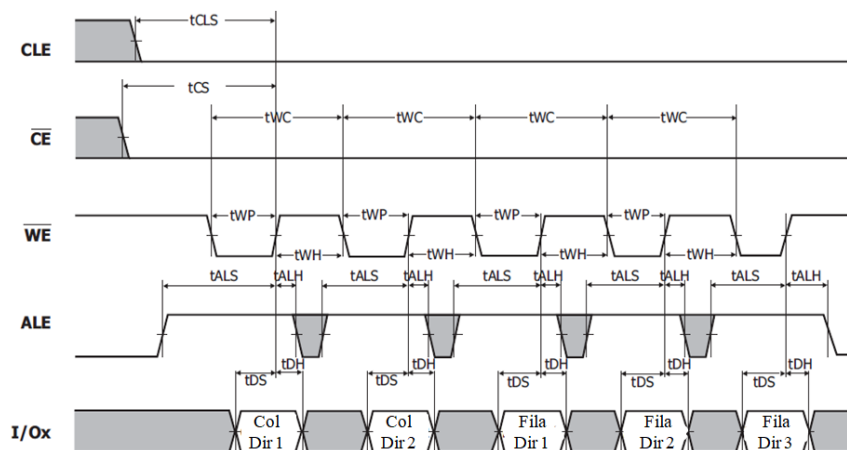


Figura 32 Comportamiento de señales para enviar los bytes de dirección al dispositivo NAND Flash (BROADCOM, 2008).

Recibidos los bytes de dirección la interfaz envía a través de UART el carácter ASCII “b” al Registrador de datos, dicha acción se realizar en la etapa 22 del programa.

El Registrador de datos ordena al dispositivo NAND Flash a borrar el bloque especificado en los bytes de dirección a través de comandos específicos, en caso que la memoria haya sido borrada adecuadamente, el Registrador de datos envía como eco el carácter ASCII “T”, con esto se informa al usuario a través de un cuadro de mensaje que la operación fue exitosa. En caso que el bloque no haya sido borrado adecuadamente se envía como eco el carácter ASCII “W”. Para borrar un bloque del dispositivo NAND Flash el microcontrolador ejecuta las señales de control de acuerdo a los tiempos de la Figura 33, los parámetros mínimos y máximos de los tiempos se presentan en la Tabla 6.

Tabla 6 Parámetros de tiempos mínimos y máximos para borrar un bloque del dispositivo NAND Flash (BROADCOM, 2008).

Símbolo	3.3V		Unidad
	Mínimo	Máximo	
tWB	25	-	ns
tWC	35	-	ns
tBERS	17	-	ns

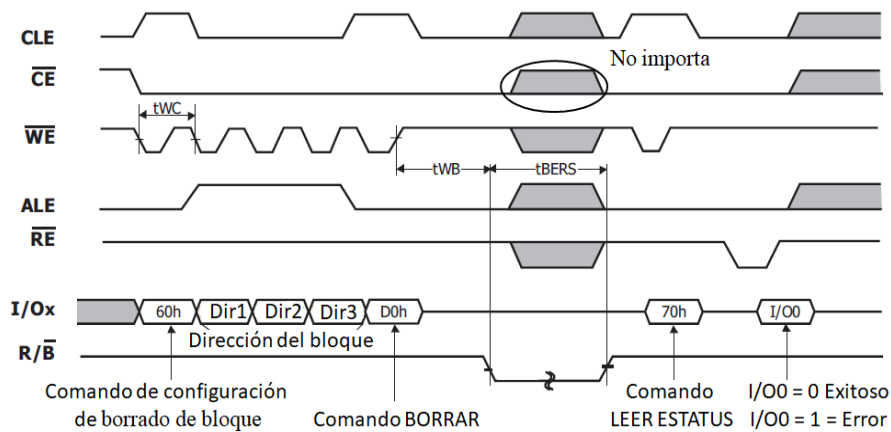


Figura 33 Comportamiento de señales para borrar un bloque del dispositivo NAND Flash (BROADCOM, 2008).

El microcontrolador MSP430 monitorea el estado del dispositivo NAND Flash después de la operación BORRAR BLOQUE, la operación de LEER ESTADO (70h) es un comando que se puede emitir mientras el dispositivo NAND Flash está ocupado. Leer el estado también informa el estado de pasa/falla de las operaciones PROGRAMAR/BORRAR.

Es obligatorio que el estado aprobado se obtenga después de programar o borrar una página de la memoria NAND Flash para garantizar la integridad adecuada de los datos.

Para obtener el estado del dispositivo NAND Flash el microcontrolador MSP430 ejecuta las señales de control de acuerdo a los tiempos de la Figura 34, los parámetros mínimos y máximos de los tiempos se presentan en la Tabla 7.

Tabla 7 Parámetros de tiempos mínimos y máximos para verificar el estado del dispositivo NAND Flash (BROADCOM, 2008).

Símbolo	3.3V		Unidad
	Mínimo	Máximo	
tCLS	12	-	ns
tCLH	5	-	ns
tCS	20	-	ns
tCH	5	-	ns
tWHR	80	-	ns
tCHZ	-	50	ns
tCOH	15	-	ns
tRHOH	15	-	ns
tRLOH	5	-	ns

Lectura de una página del dispositivo NAND Flash.

Para realizar la lectura de una página del dispositivo NAND Flash es necesario especificar el número de página y bloque en los cuadros de entrada “Página” y “Bloque” respectivamente, la acción empieza cuando se presiona el botón “Leer” (ver Figura 36).

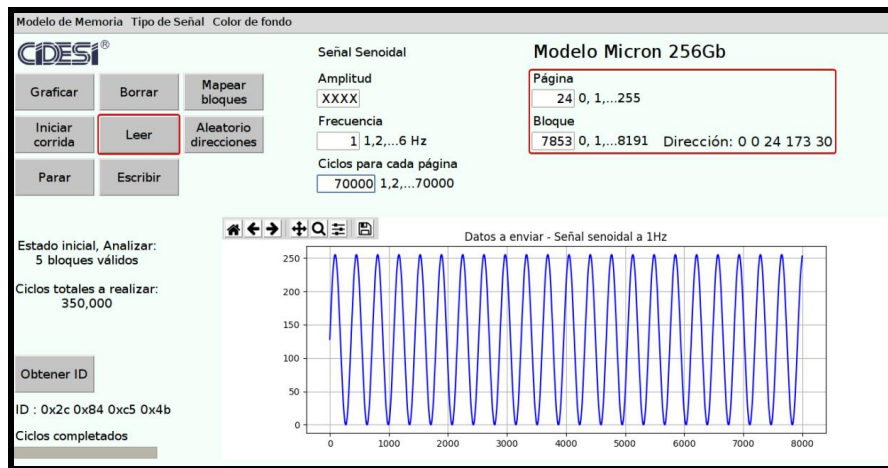


Figura 36 Leer una página del dispositivo NAND Flash mediante la interfaz gráfica de usuario.

El diagrama GRAFCET programado con Python en Raspberry Pi® de la Figura 37 muestra el proceso para leer una página del dispositivo NAND Flash conectado.

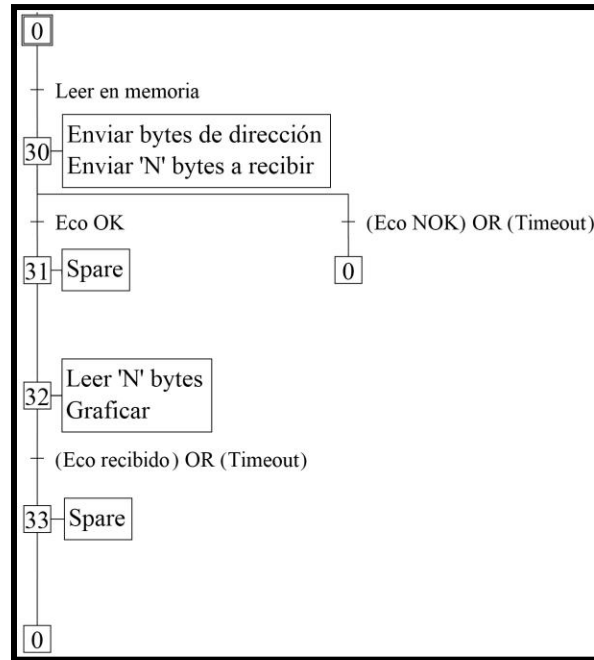


Figura 37 Diagrama de control para leer una página del dispositivo NAND Flash.

La dirección completa de 5 bytes es requerida para realizar la operación de lectura de una página en el dispositivo NAND Flash. Una operación de lectura de una página en una memoria NAND Flash Micron de 256Gb con 8600 bytes por página es de alrededor de 3000 μ s.

Para realizar la operación de lectura de una página del dispositivo NAND Flash Raspberry Pi® envía a través de UART el carácter ASCII “D” al Registrador de datos. Dicha acción se realiza en la etapa 30 del programa (ver Figura 37), al enviarse el carácter “D” la interfaz espera un eco, esto indica que el Registrador de datos recibió el carácter, y está preparado para recibir los bytes de dirección, seguido de esto la interfaz envía los 5 bytes de dirección correspondientes al número de bloque y página seleccionados en la interfaz.

Para enviar los bytes de dirección al dispositivo NAND Flash el microcontrolador ejecuta las señales de control de acuerdo a los tiempos de la Figura 32, los parámetros mínimos y máximos de los tiempos se presentan en la Tabla 5.

Recibidos los bytes de dirección la interfaz envía a través de UART el carácter ASCII “L” al Registrador de datos, dicha acción se realiza en la etapa 32 del programa (ver Figura 37). En la misma etapa del programa se grafican los datos recibidos en caso de que no hubiese algún error de transferencia. La lectura de una página borrada debe contener bytes con valor decimal de 255. Leer más allá del final del dispositivo da como resultado datos no válidos.

Para leer una página del dispositivo NAND Flash el microcontrolador MSP430 ejecuta las señales de control de acuerdo a los tiempos de la Figura 38, los parámetros mínimos y máximos de los tiempos se presentan en la Tabla 8. En la Figura 39 se muestra la gráfica de los bytes leídos del bloque 1767 en la página 222.

Tabla 8 Parámetros de tiempos mínimos y máximos para leer una página del dispositivo NAND Flash (BROADCOM, 2008).

Símbolo	3.3V		Unidad
	Mínimo	Máximo	
tRC	35	-	ns
tREA	-	25	ns
tREH	15	-	ns
tRR	20	-	ns
tCHZ	-	50	ns
tCOH	15	-	ns
tRHOH	15	-	ns

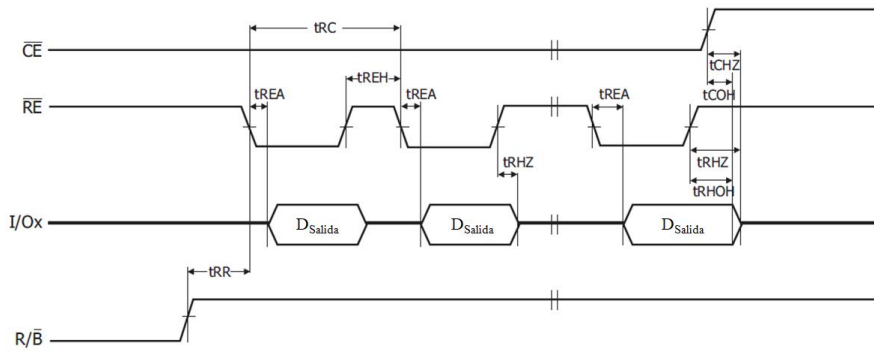


Figura 38 Comportamiento de señales para leer una página del dispositivo NAND Flash (BROADCOM, 2008).

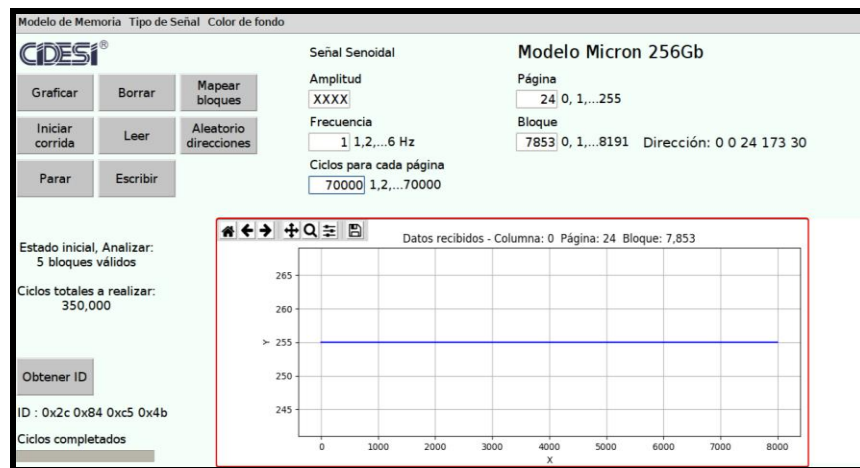


Figura 39 Página leída del dispositivo NAND Flash mediante la interfaz gráfica de usuario.

Escritura de una página del dispositivo NAND Flash.

Para realizar la escritura en una página del dispositivo NAND Flash es necesario especificar el número de página y bloque en los cuadros de entrada “Página” y “Bloque” respectivamente, la acción empieza cuando se presiona el botón “Escribir” (ver Figura 40). La acción de éste botón solo se realizará cuando la interfaz no se encuentre ocupada realizando alguna otra operación. La operación ESCRIBIR solo puede cambiar el estado de los bits a 0 y asumir que el usuario comenzó la escritura en un bloque previamente borrado.

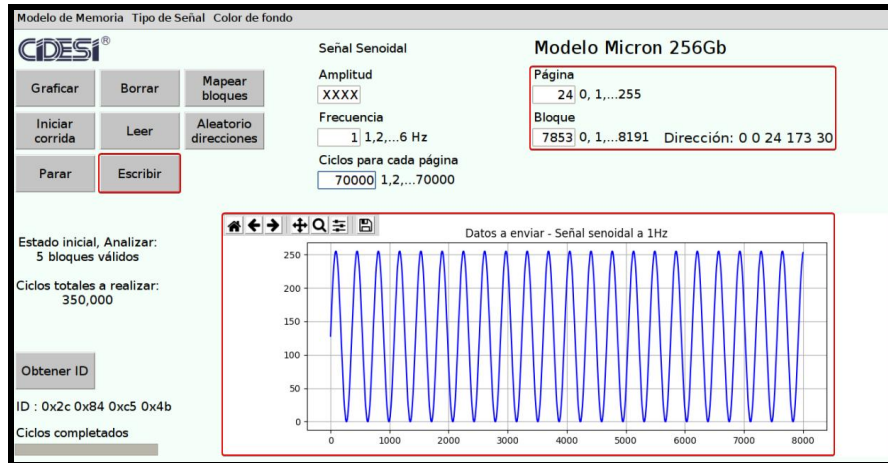


Figura 40 Escribir en una página del dispositivo NAND Flash mediante la interfaz gráfica de usuario.

El diagrama GRAFCET programado con Python en Raspberry Pi® de la Figura 41 muestra el proceso para escribir en una página del dispositivo NAND Flash conectado.

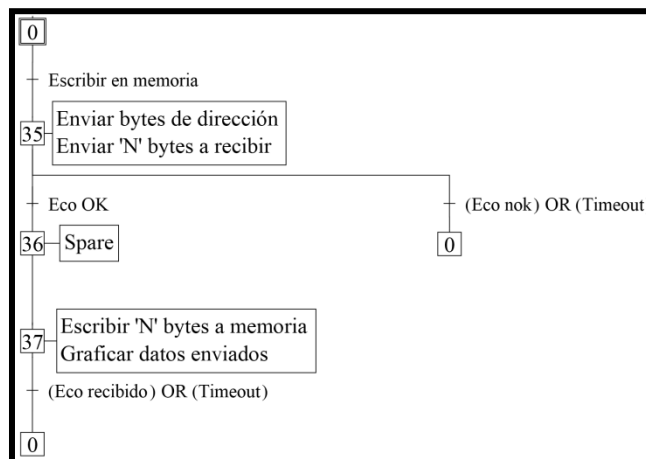


Figura 41 Diagrama de control para escribir a una página del dispositivo NAND Flash.

La dirección completa de 5 bytes es requerida para realizar la operación de ESCRITURA de una página en la memoria NAND Flash. A través de la interfaz se puede generar cuatro señales diferentes como ya se había mencionado anteriormente.

El total de elementos de la señal generada en la interfaz es igual a la cantidad de bytes que se pueden almacenar en una página del dispositivo NAND Flash. Para realizar la operación de escritura en una página en la memoria NAND Flash la interfaz envía a través de UART el carácter ASCII “D” al Registrador de datos. Dicha acción se realiza en la etapa 35 del programa (ver Figura 41). Al enviarse el carácter “D” la interfaz espera un eco, esto indica que el Registrador de datos recibió el carácter, y está preparado para recibir los bytes de dirección. Seguido de esto la interfaz envía los 5 bytes de dirección correspondientes al bloque seleccionado en la interfaz.

Para enviar los bytes de dirección al dispositivo NAND Flash el microcontrolador ejecuta las señales de control de acuerdo a los tiempos de la Figura 32, los parámetros mínimos y máximos de los tiempos se presentan en la Tabla 5. Recibidos los bytes de dirección la interfaz envía a través de UART el carácter ASCII “E” al Registrador de datos, dicha acción se realiza en la etapa 37 del programa (ver Figura 41). En la misma etapa del programa se grafican los datos enviados en caso de que no hubiese algún error de transferencia. Para escribir en una página del dispositivo NAND Flash el microcontrolador MSP430 ejecuta las señales de control de acuerdo a los tiempos de la Figura 42, los parámetros mínimos y máximos de los tiempos se presentan en la Tabla 9.

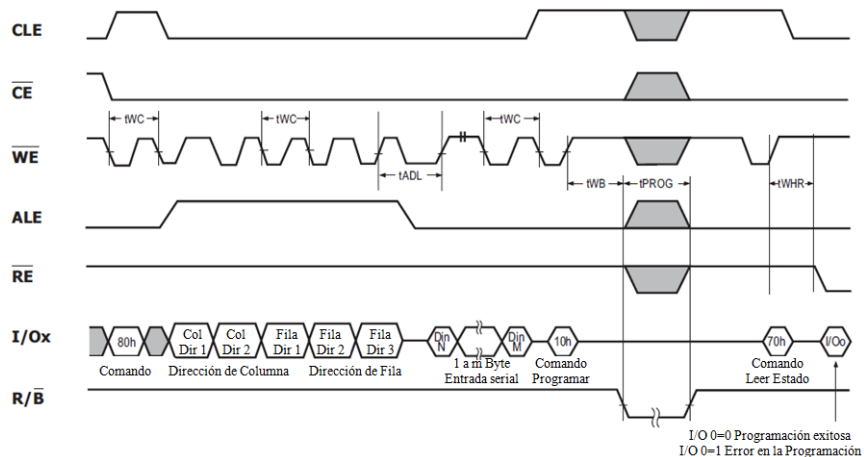


Figura 42 Comportamiento de señales para escribir en una página del dispositivo NAND Flash (BROADCOM, 2008).

Tabla 9 Parámetros de tiempos mínimos y máximos para escribir en una página del dispositivo NAND Flash (BROADCOM, 2008).

Símbolo	3.3V		Unidad
	Mínimo	Máximo	
tPROG	200	700	ns
tWC	25	-	ns
tWB	-	100	ns
tADL	70	-	ns
tWHR	80	-	ns

En la Figura 43 se muestra la gráfica de los bytes enviados a la página 142 del bloque 2440 del dispositivo NAND Flash del Registrador de datos.

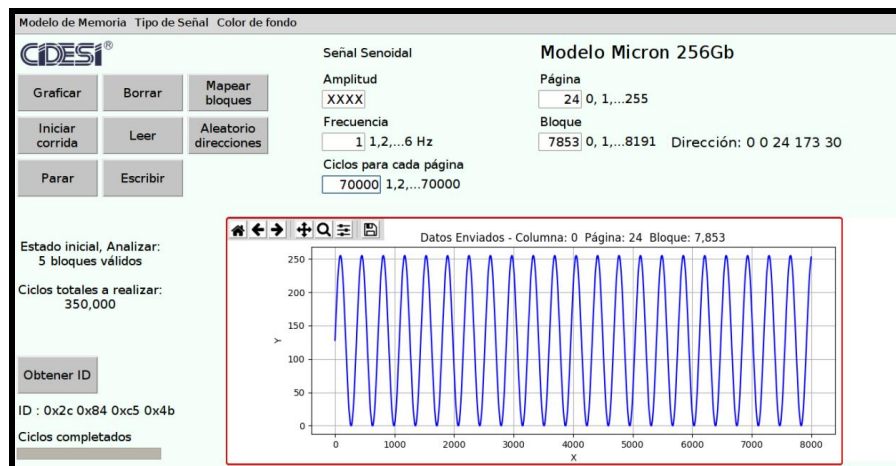


Figura 43 Bytes enviados a una página del dispositivo NAND Flash mediante la interfaz gráfica de usuario.

Mapear bloques defectuosos.

Los bloques no validos son bloques que contienen uno o más bits defectuosos más allá de un ECC. El dispositivo puede contener bloques defectuosos al momento del envío, además se pueden desarrollar bloques defectuosos adicionales con el tiempo. La especificación para un dispositivo NAND Flash de 256 Gb de la marca Micron® establece que puede tener como máximo 100 bloques defectuosos de fábrica, los circuitos internos del dispositivo NAND Flash aíslan cada bloque de otros bloques, por lo que la presencia de un bloque defectuoso no afecta el funcionamiento del resto de la matriz NAND Flash.

El fabricante identifica los bloques defectuosos de cada dispositivo NAND Flash antes del envío, esto para que puedan ser identificados por el usuario ya que es importante que no sean utilizados para almacenamiento de datos. Para el caso de la memoria NAND Flash Micron® de 256 Gb la marca de bloque defectuoso se coloca en la primera ubicación del área libre, es decir, en la página 0 en el área libre de cada bloque defectuoso, la marca de bloque malo es 00h, para otros proveedores de memorias NAND Flash puede ser necesario mapear la marca de bloque defectuoso en más de una página de cada bloque. La fábrica prueba cada dispositivo NAND Flash en una amplia gama de temperaturas y voltajes. Algunos bloques que están marcados como malos de fábrica pueden ser funcionales a ciertas temperaturas o voltajes, pero podrían fallar en el futuro. Si se borra la marca de bloque defectuoso, no se puede recuperar.

La interfaz permite mapear los bloques del dispositivo NAND Flash (ver Figura 45) para determinar qué bloques son defectuosos de fábrica. La operación debe escanear todos los bloques para determinar qué bloques son malos y luego crear un archivo csv de estos bloques defectuosos para una futura referencia. El archivo csv con dicha información se guarda en una carpeta que el usuario puede elegir desde la interfaz de Raspberry Pi®. La operación solo leer el byte donde se puede encontrar la marca de bloque malo.

El diagrama GRAFCET programado con Python en Raspberry Pi® de la Figura 44 muestra el proceso de mapeo de bloques defectuosos del dispositivo NAND Flash conectado.

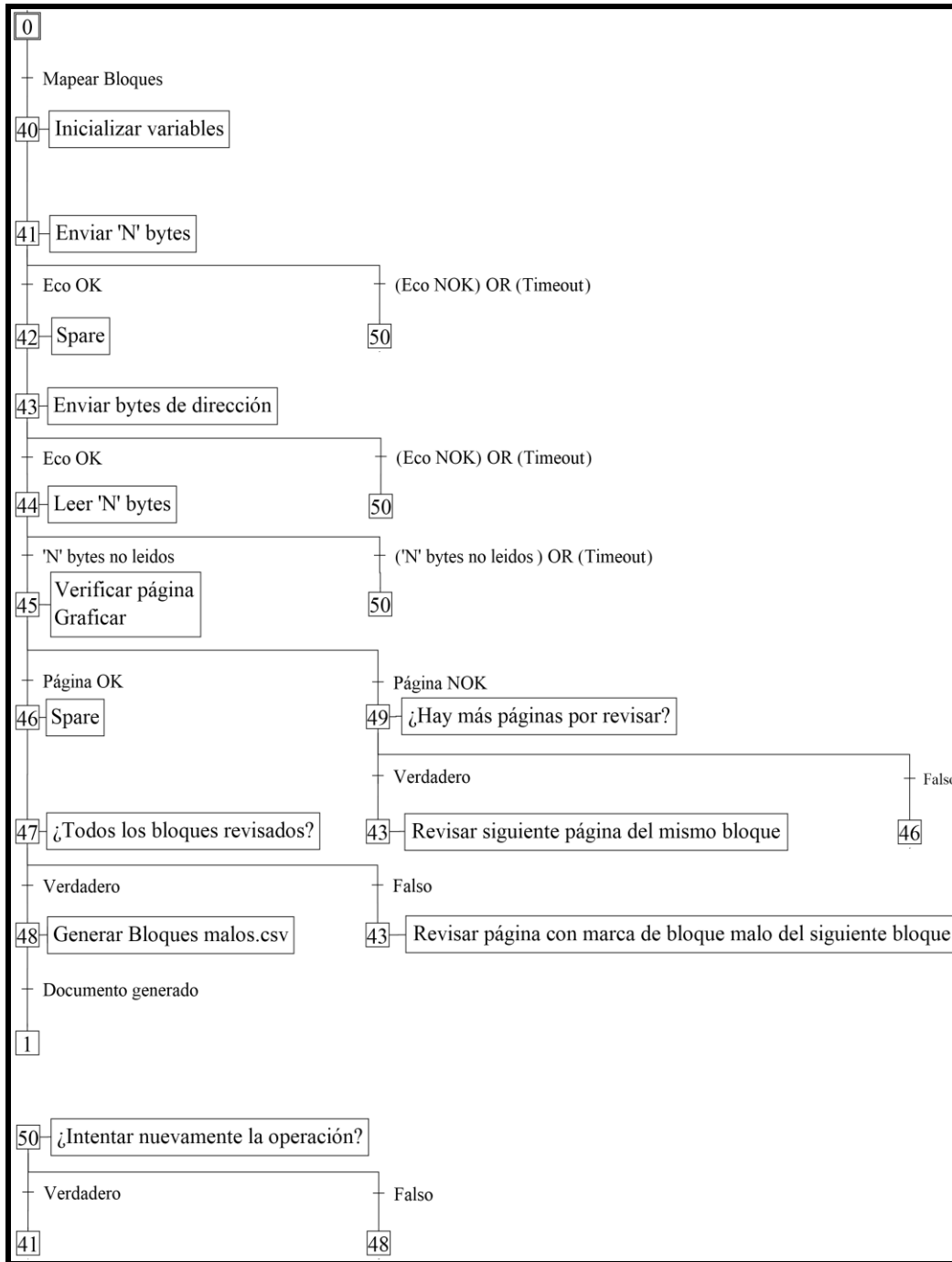


Figura 44 Diagrama de control para mapear bloques defectuosos del dispositivo NAND Flash.

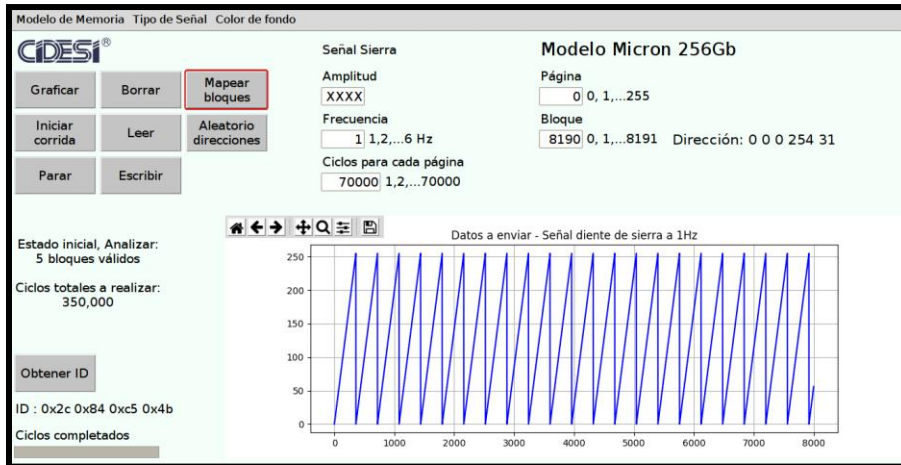


Figura 45 Mapear bloques defectuosos del dispositivo NAND Flash mediante la interfaz gráfica de usuario.

En caso de requerirse detener la secuencia de mapeo de bloques defectuosos basta con presionar el botón “Parar” en la interfaz gráfica. Durante el proceso de mapeo de bloques malos del dispositivo NAND Flash con el banco de pruebas la interfaz muestra el progreso con una barra de progreso de los bloques que se han leído “Bloques leídos” (ver Figura 46).

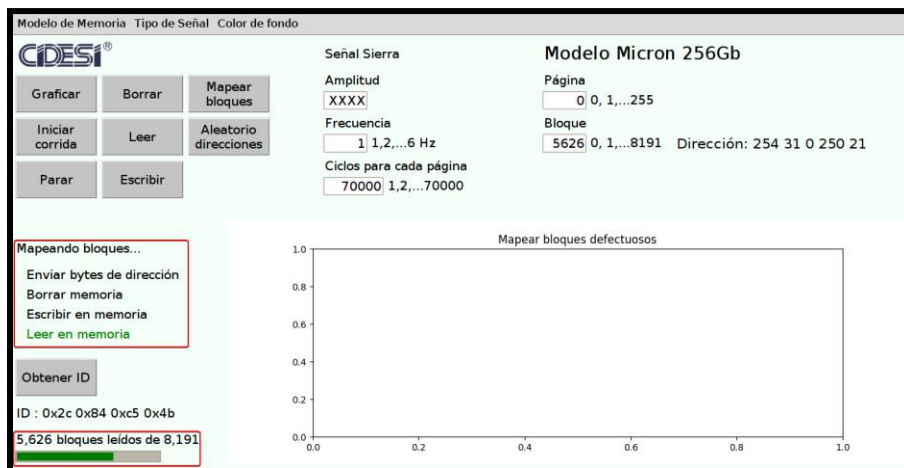


Figura 46 Barra de progreso de mapeo de bloques malos del dispositivo NAND Flash en la interfaz gráfica de usuario.

Proceso de degradación de la memoria NAND Flash.

Al iniciar la secuencia de degradación del dispositivo NAND Flash (ver Figura 41), la interfaz selecciona de manera aleatoria 5 bloques del dispositivo NAND Flash los cuales se encontrarán en bloques válidos del dispositivo NAND Flash, para esto el programa de la interfaz solicita al usuario seleccionar un archivo csv “Bloques malos.csv”, el programa de la interfaz utiliza como referencia la información del archivo “Bloques malos.csv” para seleccionar de forma aleatoria los 5 bloques. Es importante que el usuario seleccione el archivo de bloques malos que hace referencia al dispositivo NAND Flash al cual se realizará la degradación. Es recomendable que se realice un mapeo de bloques defectuosos a un dispositivo NAND Flash nuevo, esto para tener como referencia una tabla de bloques defectuosos marcados de fábrica en un archivo “Bloques malos.csv”. Para cada bloque se repetirá continuamente el proceso una cantidad de veces determinada por el usuario en la entrada “Ciclos para cada página” (ver Figura 47); borrar el bloque en el que se encuentra la página, escribir datos en toda la página, leer la página e identificar errores comparando los datos enviados originalmente con los datos que se han leído, en cada ciclo completado se registran los resultados en el archivo “Resultados.csv”, en el cual se recopila toda la información del proceso. El diagrama GRAFCET de la Figura 48 muestra el proceso de degradación, cálculo y recopilación de los resultados obtenidos.

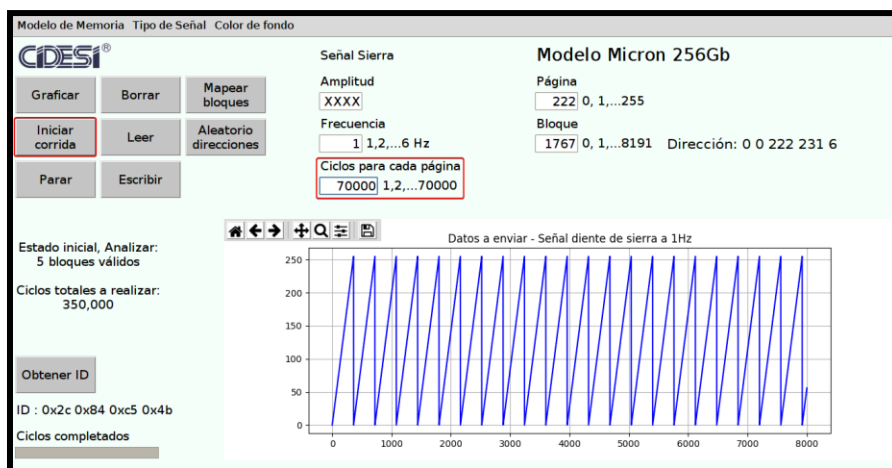


Figura 47 Degradación de memoria NAND Flash mediante la interfaz gráfica de usuario.

Al obtener los resultados finales es posible analizar el comportamiento de error en cada bloque, esto ya que el rendimiento de cada celda del dispositivo NAND Flash se degrada a medida que se repite la operación de borrado/escritura. La interfaz es capaz de graficar los datos leídos del dispositivo NAND Flash en cada ciclo (ver Figura 49), la desventaja es que al programa le toma un cierto tiempo desplegar información en la gráfica de la interfaz consecuentemente se aumenta el tiempo de ciclo del proceso en un 30 % aproximadamente.

El usuario puede activar o desactivar esta función con el botón “Graficar”, adicionalmente la interfaz muestra en todo momento indicadores de texto (ver Figura 50), lo cuales señalan el proceso que realiza el banco de pruebas para degradar el dispositivo NAND Flash.

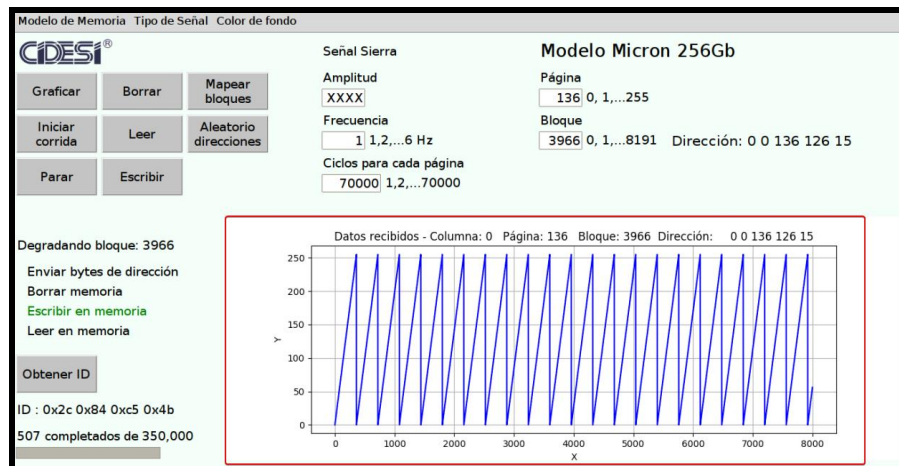


Figura 49 Grafica de datos recibidos del dispositivo NAND Flash en la interfaz gráfica de usuario.

La interfaz muestra el progreso de la secuencia con una barra de progreso de los ciclos que se han completado “Ciclos completados” (ver Figura 50).

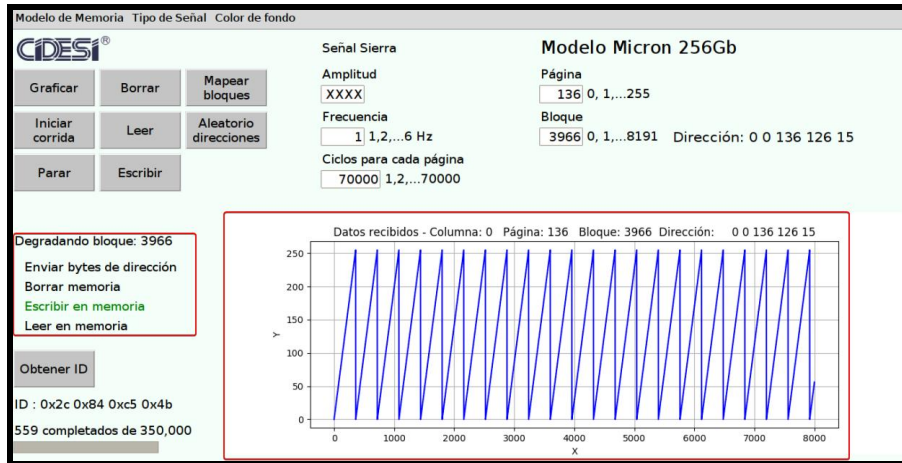


Figura 50 Indicadores de proceso de ejecución de la secuencia Borra-Escribir-Leer del dispositivo NAND Flash en la interfaz gráfica de usuario.

En caso de requerirse detener la secuencia de degradación que hace el banco de pruebas al dispositivo NAND Flash, basta con presionar el botón “Parar” en la interfaz (ver Figura 52). La operación terminará la secuencia en una manera controlada, es decir, en el caso de haber un ciclo en ejecución en el momento de solicitar la parada de secuencia, el programa completará el ciclo para no afectar los resultados finales, para luego detener la secuencia.

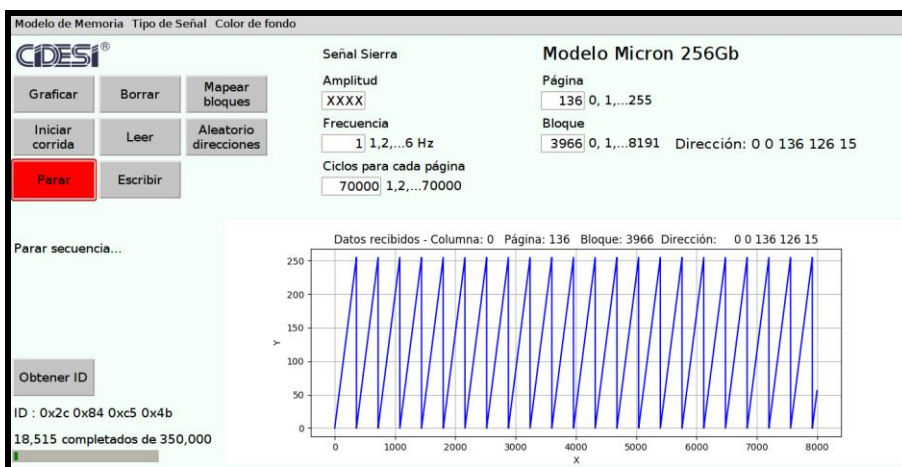


Figura 51 Paro controlado de la secuencia Borra-Escribir-Leer del dispositivo NAND Flash en la interfaz gráfica de usuario.

Resultados

En esta sección se muestran los resultados obtenidos del banco de pruebas degradación de memorias NAND Flash al analizar un total de 10 bloques del dispositivo NAND Flash Micron® modelo MT29F256G08CJABB del registrador de datos, los números de bloque fueron elegidos aleatoriamente por el banco de pruebas. Se realizó un mapeo de bloques defectuosos al dispositivo NAND Flash con el banco de pruebas, el mapeo detectó un total de 4 bloques defectuosos de 8191 bloques escaneados, en la Tabla 10 se muestran los números de bloque defectuosos encontrados.

Tabla 10 Bloques defectuosos del dispositivo Micron® modelo MT29F256G08CJABB.

Número de bloque defectuoso
90
91
4186
4187

Para una primera prueba de degradación se selecciona una señal de diente de sierra de 1 Hz y un total de 50 mil ciclos Borrar-Escribir-Leer a 5 bloques del dispositivo NAND Flash, para cada página se escriben y se leen 8000 bytes.

En la Tabla 11 se muestran los números de bloque y página seleccionados de manera aleatoria por la interfaz.

Tabla 11 Número de bloque y página para prueba de degradación con señal diente de sierra 1 Hz.

Número de bloque	Número de página
454	220
1963	253
2784	24
5654	100
7721	193

En Figura 53 se muestra la cantidad de bytes recibidos que fueron detectados con errores en cada ciclo realizado, así también en la figura 54 se muestra la cantidad de bits recibidos que fueron detectados con mutaciones en los bytes con errores en cada ciclo realizado y en la Figura 55 se muestra el RBER por cada ciclo realizado.

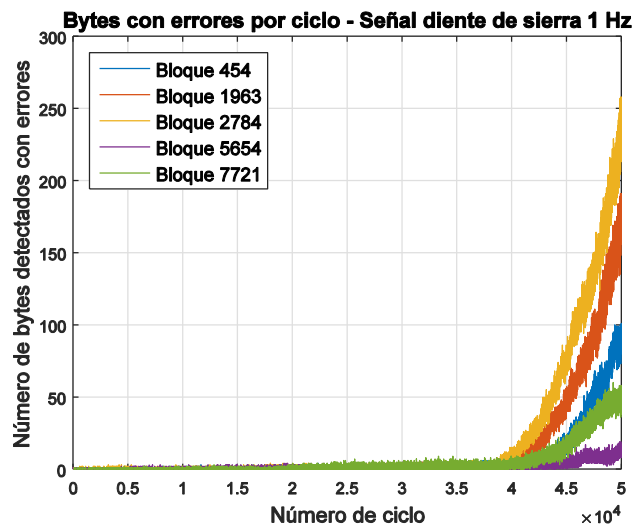


Figura 52 Resultados obtenidos del banco de pruebas empleando una señal diente de sierra de 1 Hz: número de bytes detectados con errores por ciclo.

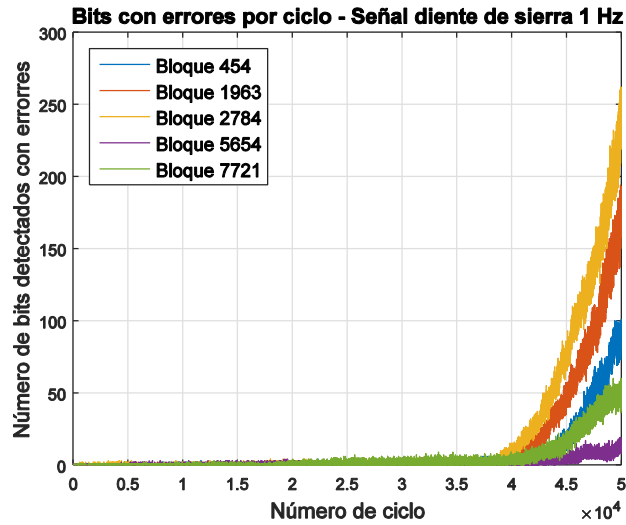


Figura 53 Resultados obtenidos del banco de pruebas empleando una señal diente de sierra de 1 Hz: número de bits detectados con errores por ciclo.

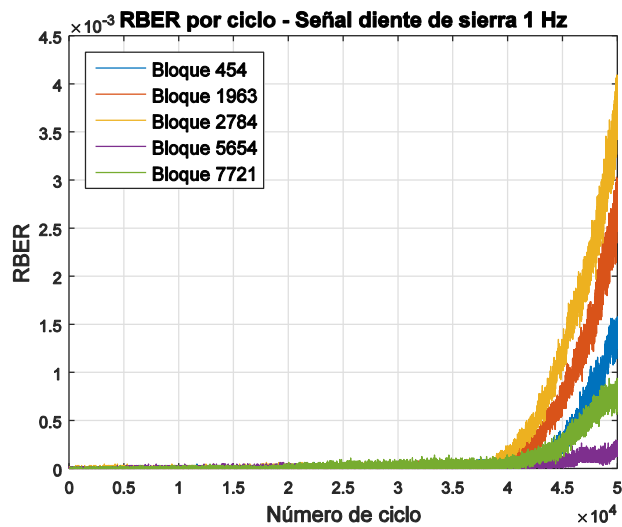


Figura 54 Resultados obtenidos del banco de pruebas empleando una señal diente de sierra de 1 Hz: RBER por ciclo.

En las Tablas 12, 13, 14, 15 y 16 se muestran los bytes que se detectaron con errores (marcados con una “X”) en 10 iteraciones consecutivas. En las tablas se puede observar que la mayoría de los bytes con errores detectados tienen una tendencia de repetición de error con las iteraciones consecutivas, las celdas verdes muestran las posiciones de los bytes con error que se repitieron a lo largo de las 10 iteraciones analizadas.

Tabla 12 Bytes con errores detectados de 10 ciclos consecutivos del bloque 454.

	Número de ciclo									
	48606	48607	48608	48609	48610	48611	48612	48613	48614	48615
Número de byte con error	37	X	X	X	X	X	X	X	X	X
	198	X			X					X
	407	X	X			X			X	X
	483	X	X	X	X	X	X	X	X	X
	589	X	X	X	X	X	X	X	X	X
	605	X			X	X	X		X	
	903	X	X	X	X	X	X	X	X	X
	1139	X		X	X					X
	1173	X	X	X		X	X	X	X	X
	1207	X	X	X	X	X	X	X	X	X
	1221	X	X	X	X	X	X	X		X
	1511	X	X	X	X	X	X	X	X	X
	1656	X	X	X	X	X	X	X	X	X
	1822	X		X					X	X
	1846	X	X	X	X	X		X	X	X
	2211	X	X	X	X	X	X	X	X	X
	2364	X	X	X	X	X	X	X	X	X
	2488	X	X	X	X	X		X	X	X
	2633	X	X	X	X	X	X	X	X	X
	2663	X	X							
	2722	X	X	X	X	X	X	X	X	X
	2768	X	X					X	X	X
	2795	X	X			X	X	X	X	X
	3226	X	X	X	X	X	X	X	X	X
	3227	X	X	X	X	X	X	X	X	X
	3300	X	X	X	X	X	X	X	X	X
	3332	X	X	X	X	X	X	X	X	X
	3617	X								X
	3780	X	X	X	X	X	X	X	X	X
	3918	X	X	X	X	X	X	X	X	X
	4010	X	X	X	X	X	X	X	X	X
	4244	X	X	X	X	X	X	X	X	X
	4283	X	X	X	X			X	X	X
	4436	X	X	X		X		X	X	X
	4466	X	X					X	X	X
	4539	X	X	X	X	X	X	X	X	
	4692	X				X	X			X
	4753	X	X	X	X	X	X	X	X	X
	5342	X	X	X	X		X	X		X
	5348	X	X	X		X			X	X
5396	X	X	X	X		X			X	
5539	X	X	X	X	X	X	X	X	X	
5592	X	X	X	X	X	X	X	X	X	
5645	X	X		X	X		X		X	
5774	X	X	X	X	X	X	X	X	X	
6060	X	X	X	X	X	X	X	X	X	
6400	X		X	X	X	X	X		X	
6882	X									
7024	X	X	X	X	X	X	X	X	X	
7035	X	X	X	X	X	X	X	X	X	
7060	X	X	X	X	X	X	X	X	X	
7251	X									
7311	X	X	X	X	X	X	X	X	X	
7409	X					X				
7526	X	X	X	X	X	X	X	X	X	
7786	X	X	X	X	X	X	X	X	X	
7895	X	X	X	X	X	X	X	X	X	
7950	X	X	X	X	X	X	X	X	X	
7979	X	X	X	X	X	X	X	X	X	

Tabla 13 Bytes con errores detectados de 10 ciclos consecutivos del bloque 1963.

		Número de ciclo									
		45493	45494	45495	45496	45497	45498	45499	blo # 455	45501	45502
Número de byte con error	94	X	X	X	X	X	X	X	X	X	X
	129	X	X	X	X	X	X	X		X	X
	247	X									X
	273	X	X	X		X	X	X	X	X	X
	326	X	X	X	X	X					X
	416	X	X	X	X		X			X	X
	869	X	X	X	X	X	X	X	X	X	X
	1031	X	X	X		X	X	X	X	X	
	1079	X	X	X	X	X	X	X	X	X	X
	1105	X									
	1246	X	X	X	X	X	X	X			
	1358	X	X	X	X	X	X	X	X	X	X
	1449	X	X	X	X	X	X			X	X
	1548	X	X	X	X	X	X		X	X	X
	1852	X	X	X	X	X	X	X	X		X
	2040	X	X	X	X	X	X	X	X		X
	2132	X	X	X	X	X	X	X	X	X	X
	2257	X	X	X	X	X		X	X	X	X
	2307	X		X		X	X	X	X	X	X
	2384	X	X	X				X	X		X
	2542	X	X	X	X	X	X	X	X	X	X
	2644	X	X	X	X	X	X	X	X	X	X
	2728	X	X	X	X	X	X	X	X	X	X
	2880	X									
	3083	X					X	X	X	X	X
	3194	X	X	X	X	X	X	X	X	X	X
	3229	X	X	X	X	X	X	X	X	X	X
	3257	X		X			X				
	3327	X	X	X	X	X	X	X	X	X	X
	3431	X	X	X	X	X	X	X	X	X	X
	3637	X	X	X						X	
	3858	X				X					
	3944	X	X	X	X	X	X	X	X		
	3968	X	X	X	X	X	X	X	X	X	X
	4004	X	X	X	X	X	X	X	X	X	X
	4167	X	X	X	X	X	X	X	X	X	X
	4231	X	X	X				X	X	X	X
	4268	X		X	X	X	X	X	X	X	X
	4889	X	X	X	X	X	X	X	X	X	X
	4894	X	X	X			X				
	4966	X	X	X	X	X	X	X	X	X	X
	5018	X	X			X			X		
	5138	X	X	X		X	X		X	X	X
	5377	X	X	X	X	X	X	X	X	X	X
	5697	X	X	X	X	X	X		X	X	X
5927	X	X	X				X	X	X	X	
5947	X	X	X	X	X	X	X	X	X	X	
5988	X										
6076	X	X	X	X	X	X	X	X	X	X	
6158	X										
6241	X	X	X	X	X	X	X	X	X	X	
6547	X	X	X	X	X	X	X	X	X	X	
6655	X	X	X	X	X	X	X	X	X	X	
6978	X	X	X	X	X	X	X	X	X	X	
7075	X				X	X	X	X		X	
7081	X	X	X	X	X	X	X	X	X	X	
7113	X	X	X	X	X				X		
7118	X	X	X	X	X	X	X	X	X	X	
7162	X	X	X	X	X	X	X	X	X	X	
7233	X	X		X	X	X	X	X	X	X	
7474	X	X	X	X	X	X	X	X	X	X	
7751	X		X	X	X	X					
7841	X						X	X			
7903	X										

Tabla 14 Bytes con errores detectados de 10 ciclos consecutivos del bloque 2784.

		Número de ciclo									
		44162	44163	44164	44165	44166	44167	44168	44169	44170	44171
Número de byte con error	59	X	X	X	X	X	X	X	X	X	X
	63	X	X	X	X	X	X	X	X	X	X
	262	X									
	399	X	X	X	X	X	X	X	X	X	X
	420	X	X	X	X	X	X	X	X	X	X
	428	X	X	X	X	X	X	X	X	X	X
	516	X	X	X	X	X	X	X	X	X	X
	569	X	X	X	X	X	X	X	X	X	X
	608	X			X					X	X
	764	X	X	X	X	X	X	X	X	X	X
	874	X	X	X	X	X	X	X	X	X	X
	889	X	X	X	X	X	X	X	X	X	X
	1043	X	X	X	X	X	X	X	X	X	X
	1045	X	X	X	X	X	X	X	X	X	X
	1182	X									
	1378	X	X	X	X	X	X	X	X	X	X
	1465	X	X	X	X	X		X	X	X	X
	1603	X				X				X	
	1706	X	X	X	X	X	X	X	X	X	X
	1749	X	X	X	X		X	X	X	X	X
	1815	X	X	X	X	X		X	X	X	X
	1897	X									
	1989	X	X	X	X	X	X	X	X	X	X
	2295	X	X	X			X		X	X	X
	2498	X									
	3201	X									
	3608	X	X	X	X	X	X	X	X	X	X
	3878	X	X	X	X	X	X	X	X	X	X
	3892	X	X	X	X	X	X	X	X	X	X
	3916	X									X
	4053	X	X	X	X	X	X	X	X	X	X
	4256	X	X	X	X	X	X	X	X	X	X
	4388	X	X	X	X	X	X	X	X	X	X
	4418	X	X	X	X	X	X		X	X	X
	4455	X	X	X	X	X	X	X	X	X	X
	4596	X	X	X	X	X	X	X	X	X	X
	4860	X	X	X	X	X	X	X	X	X	X
	4949	X	X	X	X	X	X	X	X	X	X
	4971	X	X	X	X	X		X	X		
	5193	X	X	X	X	X	X	X	X	X	X
	5236	X	X	X	X	X	X	X	X	X	X
	5772	X	X	X	X	X		X		X	X
	5828	X	X	X	X	X	X	X	X	X	X
	6014	X						X			
	6057	X	X					X	X	X	X
	6413	X		X	X		X	X			X
6414	X	X	X	X	X	X	X	X	X	X	
6513	X	X	X	X	X	X	X	X	X	X	
6593	X	X	X	X	X	X	X		X	X	
6669	X	X	X		X	X	X	X	X	X	
6956	X				X		X	X	X	X	
7190	X	X		X	X	X	X	X	X		
7199	X	X	X						X	X	
7206	X	X	X	X	X	X	X	X	X	X	
7234	X	X	X	X	X	X	X	X	X	X	
7367	X	X	X	X	X	X	X	X	X		
7392	X	X	X	X	X	X	X	X	X	X	
7425	X	X	X	X	X	X	X	X	X	X	
7587	X	X	X	X	X	X	X	X	X	X	
7672	X	X	X	X	X	X	X	X	X	X	
7879	X	X									

Tabla 15 Bytes con errores detectados de 10 ciclos consecutivos del bloque 5654.

	Número de ciclo									
	55540	55541	55542	55543	55544	55545	55546	55547	55548	55549
287	X	X	X	X	X	X	X		X	X
565	X	X	X	X	X	X	X	X	X	X
1094	X	X	X	X	X	X	X		X	
1185	X	X						X	X	X
1217	X	X	X		X		X	X		X
1295	X	X				X				
1302	X	X	X	X	X	X	X	X	X	X
1309	X			X	X	X		X		X
1488	X	X	X	X	X	X	X	X	X	X
1873	X	X	X	X	X	X	X	X		X
2086	X	X	X	X	X	X	X	X	X	X
2382	X									
2763	X	X								
2996	X	X	X	X	X	X	X	X	X	X
3361	X	X	X	X	X	X	X	X	X	X
3542	X	X	X	X	X	X	X	X	X	X
3752	X	X	X		X			X	X	
3826	X	X	X	X	X	X	X	X	X	X
3847	X				X	X	X	X	X	
3948	X	X				X			X	
4049	X	X	X	X		X			X	X
4258	X	X	X	X	X	X	X	X	X	X
4340	X	X	X	X	X	X	X		X	X
4455	X							X		
4745	X	X	X	X	X	X	X	X	X	X
5223	X	X								
5409	X						X			
5502	X	X	X	X	X	X	X	X	X	X
5628	X	X	X	X	X	X	X	X	X	X
6230	X								X	X
6331	X									
6470	X	X	X	X	X	X	X	X	X	X
7027	X	X	X	X	X	X	X	X	X	X
7206	X									
7208	X	X		X		X			X	X
7242	X	X	X	X	X	X	X	X	X	X
7526	X	X	X	X	X		X	X	X	X
7563	X							X		X

Tabla 16 Bytes con errores detectados de 10 ciclos consecutivos del bloque 7721.

		Número de ciclo									
		49152	49153	49154	49155	49156	49157	49158	49159	49160	49161
Número de byte con error	68	X	X	X	X	X	X	X	X	X	X
	365	X	X	X	X	X	X	X	X	X	X
	463	X	X	X		X	X	X			
	574	X	X								
	814	X	X		X	X	X	X	X		X
	1318	X	X	X	X	X	X	X	X	X	X
	1756	X	X	X	X	X	X		X		
	1777	X	X	X	X	X	X	X	X	X	X
	1797	X	X	X	X		X	X	X		
	1962	X	X	X	X	X	X	X	X	X	X
	1980	X	X	X	X	X	X	X	X	X	X
	2079	X	X		X	X	X		X		X
	2137	X	X	X	X	X	X	X	X	X	X
	2147	X	X	X	X	X	X		X		
	2159	X	X	X	X	X	X	X	X	X	X
	2264	X	X	X	X	X	X	X	X	X	X
	2270	X	X	X	X	X	X	X	X	X	X
	2647	X	X	X	X	X	X	X	X	X	X
	3436	X	X	X	X	X	X	X	X	X	X
	3459	X	X	X	X				X	X	
	3628	X	X	X	X	X	X	X	X	X	X
	4103	X	X	X	X	X	X		X	X	
	4376	X	X	X		X	X				
	4377	X	X	X	X	X	X	X	X	X	X
	4381	X		X			X		X		
	4508	X			X		X	X	X	X	
	4517	X		X	X		X		X	X	X
	4544	X						X			
	4594	X			X					X	
	4689	X		X							
	5310	X	X	X	X	X	X	X	X	X	X
	5555	X				X					
5934	X	X	X	X	X	X	X	X	X	X	
6018	X	X	X	X	X	X	X	X	X	X	
6097	X	X	X	X	X	X	X	X	X	X	
6198	X	X	X	X	X	X	X	X	X	X	
6254	X	X	X	X	X	X	X	X	X	X	
6581	X	X	X	X	X	X	X	X	X	X	
6785	X	X	X	X	X	X	X	X	X	X	
6988	X					X			X		
7310	X	X	X	X	X	X	X	X	X	X	
7525	X	X	X	X	X	X	X	X	X	X	
7730	X	X	X	X	X	X	X	X	X	X	
7787	X										

Para una segunda prueba de degradación se selecciona una señal de tipo constante con valor de cero y un total de 50 mil ciclos a 5 bloques del dispositivo NAND Flash. En la Tabla 17 se muestran los números de bloque y página seleccionados por la interfaz.

Tabla 17 Número de bloque y página para prueba de degradación con señal constante 0.

Número de bloque	Número de página
93	113
3538	193
6372	25
7209	193
6243	9

En Figura 56 se muestra la cantidad de bytes recibidos que fueron detectados con errores en cada ciclo realizado, así también en la figura 57 se muestra la cantidad de bits recibidos que fueron detectados con mutaciones en los bytes con errores en cada ciclo realizado y en la Figura 58 se muestra el RBER por cada ciclo realizado.

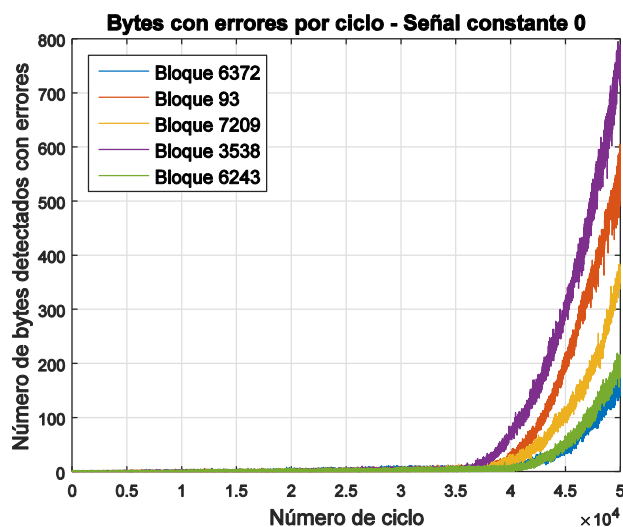


Figura 55 Resultados obtenidos del banco de pruebas empleando una señal 0 constante: número de bytes detectados con errores por ciclo.

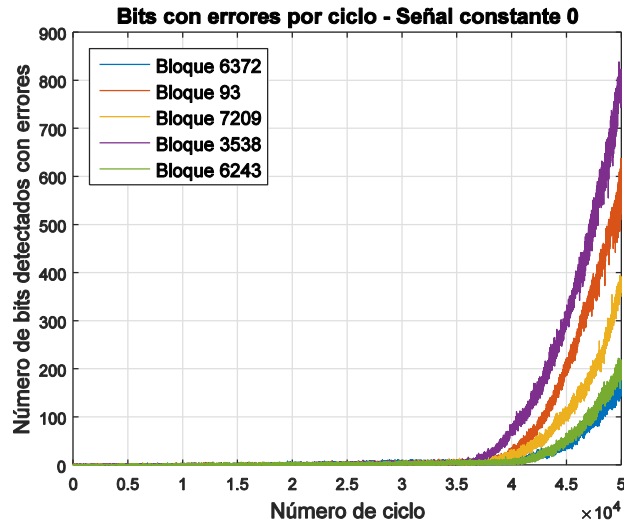


Figura 56 Resultados obtenidos del banco de pruebas empleando una señal 0 constante: número de bits detectados con errores por ciclo.

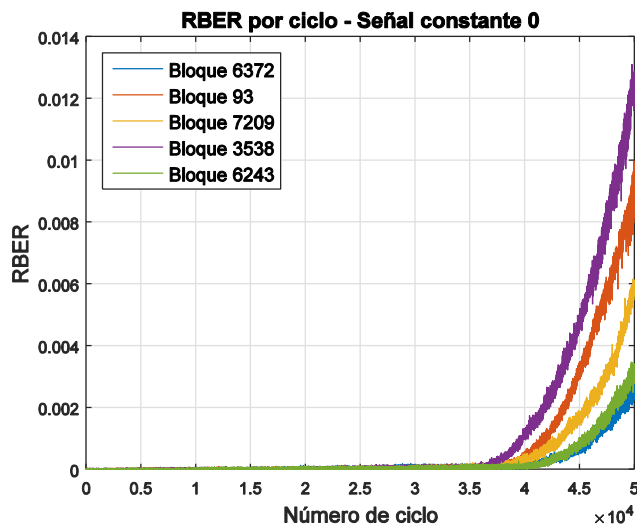


Figura 57 Resultados obtenidos del banco de pruebas empleando una señal 0 constante: RBER por ciclo.

En las Tablas 18, 19, 20, 21 y 22 se muestran los bytes que se detectaron con errores (marcados con una “X”) en 10 iteraciones consecutivas. En las tablas se puede observar que la mayoría de los bytes con errores detectados tienen una tendencia de repetición de error con las iteraciones consecutivas, las celdas verdes muestran las posiciones de los bytes con error que se repitieron a lo largo de las 10 iteraciones analizadas.

Tabla 18 Bytes con errores detectados de 10 ciclos consecutivos del bloque 93.

	Número de ciclo									
	41563	41564	41565	41566	41567	41568	41569	41570	41571	41572
Número de byte con error	132	X								
	241	X	X	X	X	X	X	X	X	X
	253	X	X	X	X	X	X	X	X	X
	541	X			X	X	X			X
	562	X	X	X	X	X	X	X	X	X
	601	X			X	X	X	X		X
	676	X	X	X	X	X	X	X	X	X
	799	X	X	X	X	X	X	X	X	X
	921	X	X	X	X	X	X	X	X	X
	934	X	X	X		X		X		
	1272	X	X	X	X	X	X	X	X	X
	1549	X	X	X	X	X	X	X	X	X
	1801	X	X	X	X	X	X	X	X	X
	1960	X								
	2041	X	X	X		X	X	X	X	
	2431	X	X		X		X		X	X
	2443	X	X	X	X	X	X	X	X	X
	2679	X	X	X	X	X		X	X	X
	2826	X	X	X	X	X	X	X	X	X
	2833	X	X	X	X	X	X	X	X	X
	2834	X	X	X	X	X	X	X	X	
	3155	X	X	X	X	X	X	X	X	X
	3328	X	X	X	X		X	X	X	X
	3362	X	X	X		X	X		X	X
	3608	X								
	3640	X		X	X	X	X	X		
	3979	X	X	X	X	X	X	X	X	X
	4363	X	X	X				X	X	X
	4423	X	X	X	X	X		X	X	X
	4427	X		X	X	X	X			
	4588	X	X	X	X	X	X	X	X	X
	4621	X		X		X				
	4716	X	X	X				X	X	X
	4879	X	X	X				X		X
	4921	X								
	5116	X	X	X	X	X	X		X	X
	5307	X	X	X	X	X	X	X	X	X
	5325	X	X		X				X	
	5560	X	X	X	X	X	X	X	X	X
	5763	X			X					
5767	X	X	X	X	X	X	X	X	X	
5974	X	X	X	X	X	X	X	X	X	
5990	X	X	X					X		
6324	X	X	X	X	X	X	X	X	X	
6387	X	X							X	
6553	X	X	X	X	X	X	X	X	X	
6580	X	X	X	X	X	X	X	X	X	
6752	X	X	X	X	X	X	X	X	X	
7037	X									
7171	X	X	X	X	X	X	X	X	X	
7445	X		X		X		X			
7461	X	X	X	X	X	X	X	X	X	
7462	X	X	X	X	X	X	X	X	X	
7615	X									
7695	X	X	X	X	X	X	X	X	X	
7782	X	X	X	X	X	X	X	X	X	
7934	X	X	X	X	X	X	X	X	X	
7970	X	X	X	X	X	X	X	X	X	

Tabla 19 Bytes con errores detectados de 10 ciclos consecutivos del bloque 3538.

		Número de ciclo									
		39671	39672	39673	39674	39675	39676	39677	39678	39679	39680
Número de byte con error	245	X	X	X	X	X	X	X	X	X	X
	505	X	X				X				
	641	X		X	X	X	X	X	X	X	X
	700	X					X				
	792	X	X	X		X		X	X	X	X
	836	X	X		X	X		X	X	X	X
	1069	X	X			X					X
	1163	X	X	X	X	X	X	X	X	X	X
	1167	X	X	X	X	X	X	X	X	X	X
	1180	X	X	X	X	X	X	X	X	X	X
	1210	X	X	X	X			X	X	X	X
	1216	X	X	X	X	X	X	X	X	X	X
	1354	X	X	X		X	X	X	X	X	X
	1524	X	X	X	X	X	X	X	X	X	X
	1568	X		X	X	X	X	X	X	X	X
	1735	X	X	X	X	X	X	X	X	X	X
	1915	X	X	X	X	X	X	X	X	X	X
	1953	X		X							
	2507	X	X	X	X	X	X	X	X	X	X
	2585	X									
	2602	X	X	X	X			X	X	X	X
	2861	X	X	X	X	X	X	X	X	X	X
	3209	X	X	X	X	X	X	X	X	X	X
	3316	X									
	3329	X		X	X		X	X	X	X	X
	3345	X	X	X	X	X	X	X	X	X	X
	3446	X	X	X	X	X	X	X	X	X	X
	3456	X		X	X	X	X	X			X
	3539	X	X	X	X	X	X	X	X	X	X
	3642	X	X	X		X			X	X	
	3689	X		X	X	X	X				
	3795	X	X	X	X	X	X	X	X	X	X
	4057	X	X	X	X	X	X	X	X	X	X
	4141	X									
	4292	X									
	4293	X									
	4574	X	X	X	X	X	X		X		X
	4715	X	X	X	X	X	X	X	X	X	X
	4820	X	X	X	X	X	X		X	X	X
	4827	X	X	X	X	X		X	X	X	
	4849	X	X	X	X	X	X	X	X	X	X
	4881	X	X	X	X	X	X	X	X	X	X
	5367	X	X								X
	5445	X	X	X	X	X	X	X	X	X	X
	5520	X	X	X	X	X	X	X	X	X	X
5729	X	X	X	X	X	X	X	X	X	X	
5856	X	X	X	X	X	X	X	X	X	X	
5988	X	X	X	X	X	X	X	X	X	X	
6302	X	X	X	X	X	X	X	X	X	X	
6349	X	X	X	X	X	X	X	X	X	X	
6353	X	X	X	X	X	X	X	X	X	X	
6442	X										
6956	X	X	X			X	X	X		X	
7121	X	X	X	X	X	X	X	X	X	X	
7199	X	X	X	X	X		X	X	X	X	
7219	X	X	X	X	X	X	X	X	X	X	
7377	X					X					
7409	X	X	X		X	X	X	X	X	X	
7759	X			X			X				
7895	X	X	X	X	X	X	X	X	X	X	

Tabla 20 Bytes con errores detectados de 10 ciclos consecutivos del bloque 6372.

		Número de ciclo									
		27991	27992	27993	27994	27995	27996	27997	27998	27999	28000
Número de byte con error	5952	X						X			
	7076	X									
	547			X				X	X		X
	655			X			X				
	3737			X							
	4341			X							
	746					X	X				
	1299						X				X
	4636		X				X			X	X
	7229									X	
	1920								X		
	5798								X		

Tabla 21 Bytes con errores detectados de 10 ciclos consecutivos del bloque 7209.

		Número de ciclo									
		43231	43232	43233	43234	43235	43236	43237	43238	43239	43240
Número de byte con error	32	X	X	X	X	X	X	X	X	X	X
	90	X	X		X						X
	297	X	X	X	X	X	X	X	X	X	X
	543	X	X	X	X	X	X	X	X	X	X
	691	X			X						
	750	X					X				
	1296	X	X	X		X	X	X	X	X	X
	1364	X	X	X	X	X	X	X	X	X	X
	1560	X									
	1596	X	X	X	X	X	X	X	X	X	X
	1684	X	X	X	X	X	X	X		X	X
	1697	X	X	X	X	X	X	X	X	X	X
	1874	X	X		X	X	X	X	X	X	X
	2131	X	X	X	X	X	X	X	X	X	X
	2158	X	X	X	X	X	X	X	X	X	X
	2160	X	X	X	X	X	X	X	X	X	X
	2674	X	X	X	X	X	X	X	X	X	X
	3037	X	X	X					X	X	X
	3260	X	X			X				X	
	3342	X	X	X		X	X	X	X	X	X
	3550	X									X
	4199	X	X	X	X	X	X	X	X	X	X
	4345	X									
	4425	X	X	X	X	X	X	X	X	X	X
	4756	X	X				X		X	X	
	4943	X	X	X	X	X	X	X	X	X	X
	5074	X									
	5214	X	X		X	X	X			X	
	5556	X	X	X	X	X	X	X	X	X	X
	5605	X	X	X	X	X	X	X	X	X	X
	5771	X									
	5879	X	X								
	5938	X	X	X	X	X	X	X	X		X
	5995	X						X			
	5999	X	X	X							
	6234	X	X	X	X	X		X	X	X	X
	6430	X	X	X	X	X	X	X		X	X
	6451	X	X	X	X	X	X	X	X	X	X
	6518	X	X	X	X	X	X	X	X	X	X
	6586	X									
6877	X	X	X	X	X	X	X	X			
7134	X	X	X	X			X	X	X	X	
7269	X	X	X	X	X	X	X	X	X	X	
7314	X	X	X	X	X	X	X	X	X	X	
7347	X		X	X	X	X	X	X	X	X	
7490	X										
7505	X	X	X	X	X	X	X	X	X	X	
7731	X		X	X		X	X	X	X	X	
7837	X	X	X	X	X	X	X	X		X	
7852	X	X	X		X	X		X			

Tabla 22 Bytes con errores detectados de 10 ciclos consecutivos del bloque 6243.

		Número de ciclo									
		45374	45375	45376	45377	45378	45379	45380	45381	45382	45383
Número de byte con error	35	X	X	X	X	X	X	X	X	X	X
	76	X	X	X		X	X	X	X	X	X
	176	X	X	X	X	X	X	X	X	X	X
	263	X	X	X	X	X	X	X	X	X	X
	711	X	X	X	X	X	X	X	X	X	X
	771	X	X	X	X		X				
	878	X	X	X	X	X	X	X	X	X	X
	884	X									
	918	X			X						X
	925	X	X	X	X	X	X	X	X	X	X
	1034	X	X	X	X	X	X	X	X	X	X
	1117	X	X	X	X	X	X	X	X	X	X
	1378	X	X		X						
	1571	X									
	1609	X	X	X	X	X	X	X	X	X	X
	2007	X									
	2028	X	X	X	X	X	X	X	X	X	X
	2081	X	X	X	X	X	X	X	X	X	X
	2431	X						X			
	2483	X	X	X	X	X	X	X	X	X	X
	2485	X	X		X	X		X	X		X
	2585	X									
	2605	X	X	X	X	X	X	X	X	X	X
	2636	X						X		X	
	2738	X	X	X	X	X	X	X	X	X	X
	2759	X	X	X	X	X	X	X	X	X	X
	2846	X	X	X	X	X	X	X			X
	3079	X	X	X	X	X	X	X	X	X	X
	3206	X	X	X	X		X	X	X	X	X
	3222	X	X	X	X	X	X	X	X	X	X
	3263	X	X	X	X	X	X	X	X	X	X
	3415	X	X	X	X	X	X	X	X	X	X
	3647	X					X				
	3815	X								X	
	4432	X	X	X		X	X				
	4640	X						X			
	4791	X									
	4850	X	X	X	X	X	X	X	X	X	X
	5182	X	X	X	X	X	X	X	X	X	X
	5502	X	X	X	X	X	X	X	X	X	X
5653	X		X	X		X			X		
5897	X		X	X	X	X	X	X	X	X	
5906	X	X	X	X	X	X	X	X	X	X	
5942	X	X	X	X		X	X	X	X	X	
6235	X									X	
6243	X	X	X				X		X		
6548	X	X	X		X	X	X	X	X	X	
6605	X					X					
6671	X							X			
6876	X	X	X		X	X				X	
6952	X	X	X	X	X	X	X		X	X	
7141	X	X	X	X		X	X	X	X	X	
7255	X	X	X		X		X	X	X	X	
7412	X									X	
7518	X	X	X	X	X	X	X	X	X	X	
7810	X	X	X	X	X	X	X	X	X	X	
7843	X										
7964	X	X	X	X	X	X	X	X	X	X	

Los resultados muestran que la degradación de los bloques aumenta en mayor medida la probabilidad de error cuando se escribió una señal constante 0 en comparación con la señal diente de sierra a partir de la iteración 40 mil aproximadamente (ver Figura 59). La razón por la cual ocurre este fenómeno se puede deber a que la operación de escritura en una página del dispositivo NAND Flash significa cambiar todos los bits de cero a uno, escribir ceros a toda la página previamente borrada significa excitar todas las celdas de la página con la misma intensidad, en comparación de escribir una señal, que aunque sea periódica, los valores de las celdas no siempre son los mismos, es decir, que para la señal constante con valor cero en la operación de escritura se establece el 100% de energía para cada celda y en la operación de borrado se establece el valor de 0% de energía para cada celda.

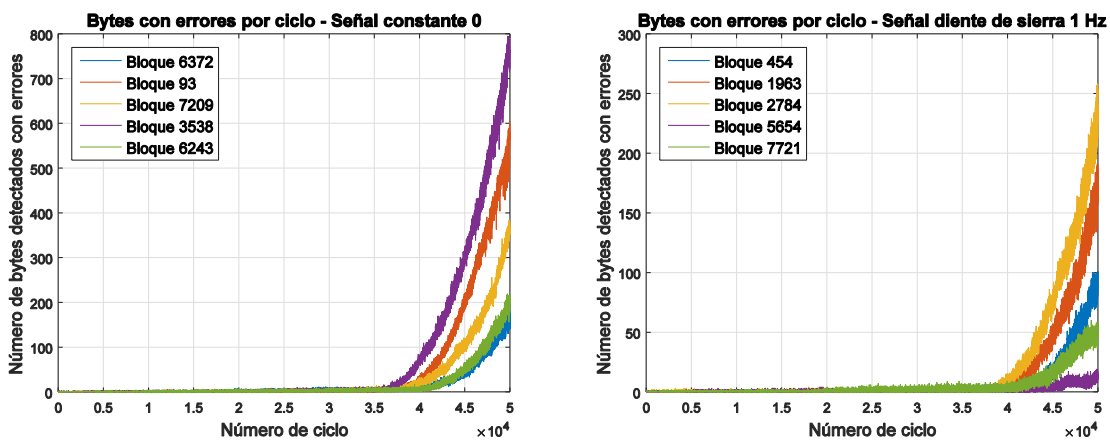


Figura 58 Comparación de resultados obtenidos para una señal tipo diente de sierra y una señal constante 0.

Conclusiones

En ocasiones uno propone un ECC de cierta capacidad de corrección, si se propone un ECC que corrige 4 pos cada 511, al ECC le lleva cierto tiempo para decodificar, tal vez no es necesario ese nivel de corrección. Con los resultados obtenidos podría proponerse un ECC adaptativo acorde con la cuenta de borrados que se lleva, para ello se requiere hacer un análisis estadístico que ayudará a dar con cierto grado de confianza que el dispositivo no fallará con cierta capacidad de corrección. Acorde a los resultados obtenidos tal vez las primeras 1000 corridas no requiera ningún ECC externo, pero entre 1000 y 2000 re requiera una capacidad de 1 por cada 255, luego de 2000 a 3000 2 por cada 511, y así sucesivamente subir el nivel de corrección.

Bibliografía

- Alvarez, A. (2015). *Guia Tkinter*. Recuperado el 1 de Agosto de 2020, de <https://guia-tkinter.readthedocs.io/es/develop/>
- Ávila. (29 de Septiembre de 2014). *Módulo OS*. Recuperado el 1 de Agosto de 2020, de <https://www.youtube.com/watch?v=QhiQ4PArooY>
- BROADCOM. (Enero de 2008). *DIGI-KEY*. Recuperado el 8 de Febrero de 2020, de <https://datasheetspdf.com/pdf-file/696239/HynixSemiconductor/HY27UF164G2B/1>
- CodigoFácil. (12 de Junio de 2012). *Lenguaje de Programación Python*. Recuperado el 1 de Agosto de 2020, de <https://www.youtube.com/watch?v=CjmzDHMHxwU>
- CoreyShafer. (9 de Agosto de 2017). *Módulo CSV*. Recuperado el 1 de Agosto de 2020, de <https://www.youtube.com/watch?v=q5uM4VKywbA>
- CursosAbiertos. (20 de Agosto de 2019). *Funciones matemáticas de Python*. Recuperado el 1 de Agosto de 2020, de <https://www.youtube.com/watch?v=u95pk80M5OU>
- David, A. (24 de Julio de 2019). *INTRODUCCIÓN A PILLOW*. Recuperado el 1 de Agosto de 2020, de https://www.youtube.com/watch?v=YL8_aAh0v5k
- De León Gordillo, D., Medina Rodríguez, C. M., Rodríguez Olivares, N. A., & Soto Cajiga, J. A. (2017). Diseño de un Registrador de Datos Usando Memorias NAND Flash Basado en un Microcontrolador de Ultrabajo Consumo. *Revista de innovación sistemática*, 19-30.
- DonPython. (16 de Diciembre de 2018). *Modulo Matplotlib*. Recuperado el 1 de Agosto de 2020, de <https://www.youtube.com/watch?v=Cj4ZYp6yakg>
- Eitan Yaakobi, J. M. (2010). *Esquemas de codificación y caracterización de errores para memorias flash*. Recuperado el 2 de Agosto de 2020, de https://www.researchgate.net/profile/Laura_Grupp/publication/228912320_Error_Characterization_and_Coding_Schemes_for_Flash_Memories/links/0fcfd50d0a15df2aea000000/Error-Characterization-and-Coding-Schemes-for-Flash-Memories.pdf

EMECATRONICOS. (26 de Diciembre de 2019). *Modulo Numpy en Python*. Recuperado el 1 de Agosto de 2020, de <https://www.youtube.com/watch?v=3Znl6uwF3MA>

Errodriger. (28 de Octubre de 2018). *Threading en Python*. Recuperado el 1 de Agosto de 2020, de <https://www.youtube.com/watch?v=uv6AM0JFhLA>

MasterCodeOnline. (20 de Septiembre de 2017). *PySerial Introduction*. Recuperado el 1 de Agosto de 2020, de <https://www.youtube.com/watch?v=kx4FoOAHG4U>

Micron®. (2020). *Micron Products*. Recuperado el 1 de Agosto de 2020, de <https://www.micron.com/search-results?searchRequest=%7B%22term%22%3A%22MT29F256G08CJABB%22%7D>

MicronTechnologies®. (2006). *Una introducción a las memorias flash NAND y cómo diseñarlo en su proximo producto*. Recuperado el 1 de Agosto de 2020, de <https://user.eng.umd.edu/~blj/CS-590.26/micron-tn2919.pdf>

Piogram. (23 de Marzo de 2020). *Módulo Random*. Recuperado el 1 de Agosto de 2020, de <https://www.youtube.com/watch?v=OkVQARcq86Q>

PythonSoftwareFundation. (2 de Septiembre de 2020). *Module time*. Recuperado el 1 de Agosto de 2020, de <https://docs.python.org/3/library/time.html>

RaspberryPi®. (Agosto de 2020). *Raspberry Web Site Oficial*. Recuperado el 1 de Agosto de 2020, de <https://www.raspberrypi.org/>

Udemy. (2018). *Ciencias de la información*. Recuperado el 7 de Agosto de 2020, de https://www.udemy.com/course/aprende-data-science-y-machine-learning-con-python/?gclid=EAIaIQobChMIre6J2PKv7QIVpeyGCh2kJglYEAAYASAAEgKBefD_BwE&utm_campaign=20181010-Spanish-mexico&utm_content=deal4584&utm_medium=udemyads&utm_source=adwords-intl&utm_term