

**Flash Memory**

# **เสนอ**

**อาจารย์วีรศักดิ์ เจริญรัตน์**

# **จัดทำโดย**

**63102105112 นายอัครพล พิกุลศรี**

# **รายงานนี้เป็นส่วนหนึ่งของรายวิชา****สถาปัตยกรรมคอมพิวเตอร์ (14122501)**

**ภาคเรียนที่ 1 ปีการศึกษา 2564**

**มหาวิทยาลัยราชภัฏสกลนคร**

# **คำนำ**

รายงานเล่มนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของรายวิชา สถาปัตยกรรมคอมพิวเตอร์ (Computer Architecture) รหัสวิชา 14122501 เพื่อให้ได้ศึกษาหาความรู้เกี่ยวกับ Flash Memory

ผู้จัดทำหหวังว่ารายงานเล่มนี้จะเป็นประโยชน์กับผู้อ่าน หรือผู้ที่กำลังหาข้อมูลเรื่องนี้อยู่หากมีข้อแนะนำหรือข้อผิดพลาดประการใด ผู้จัดทำขอน้อมรับไว้และขออภัยมา ณ ที่นี้ด้วย

นายอัครพล พิกุลศรี

# **สารบัญ**

**เรื่อง หน้า**

หน้าปก ก

คำนำ ข

สารบัญ ค

ประวัติความเป็นมา 1

หลักการทำงาน

ข้อจำกัด

การเข้าถึงระดับต่ำ

ความแตกต่างระหว่าง NOR และ NAND Flash

ระบบไฟล์แฟลช

ความจุ

อัตราการโอน

แอปพลิเคชั่น

อุตสาหกรรม

ความสามารถในการปรับขนาดแฟลช

อ้างอิง

# **ประวัติความเป็นมา**

ต้นกำเนิดของหน่วยความจำแฟลช ย้อนไปถึงการพัฒนาของ MOSFET แบบ Floating-gate (FGMOS) หรือเรียกว่าทรานซิสเตอร์แบบ Floating-gate

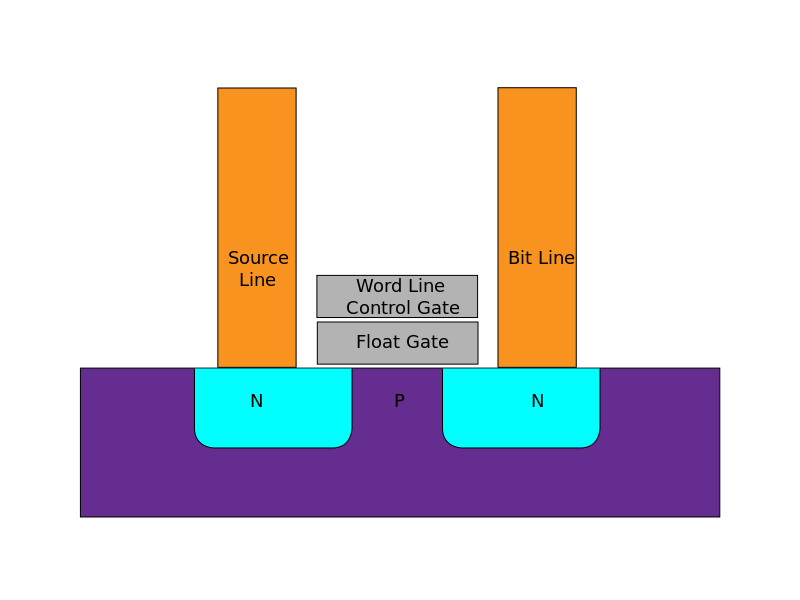
MOSFET ดั้งเดิม ทรานซิสเตอร์แบบ field-effect (ของโลหะออกไซด์-เซมิคอนดักเตอร์) หรือที่รู้จักในชื่อทรานซิสเตอร์ MOS ถูกคิดค้นโดยวิศวกรชาวอียิปต์ Mohammed M. Atalla และวิศกรชาวเกาหลี Dawon Kahng ที่ Bell Labs ในปี 1959

ต่อมาในปี 1967 Kahng กับชาววิศวกรชาวจีน Simon Min Sze ที่ Bell Labs ได้พัฒนารูปแบบอื่นคือ Floating-gate MOSFET ซึ่งได้เสนอว่าสามารถใช้เป็นเซลล์หน่วยความจำแบบ Floating-gate เพื่อจัดเก็บรูปแบบของหน่วยความจำแบบอ่านอย่างเดียวที่ตั้งโปรแกรมได้ (Programmable read-only memory) ซึ่งเป็นทั้งแบบไม่ลบเลือนและตั้งโปรแกรมซ้ำได้

หน่วยความจำแบบ Floating-gate รุ่นแรกๆ ได้แก่ EPROM (PROM ที่ลบได้) และ EEPROM (PROM แบบลบด้วยไฟฟ้า) ในปี 1970 อย่างไรก็ตาม หน่วยความจำแบบ Floating-gate ในยุคแรกๆ ต้องการให้วิศวกรสร้างเซลล์หน่วยความจำสำหรับข้อมูลแต่ละบิต ซึ่งพิสูจน์แล้วว่าช้า และมีราคาแพง โดยจำกัดหน่วยความจำแบบ Floating-gate ให้กับแอปพลิเคชันเฉพาะในช่วงทศวรรษ 1970 เช่นเป็นอุปกรณ์ทางการทหารและโทรศัพท์มือถือรุ่นทดลองรุ่นแรกสุด

# **หลักการทำงาน**

หน่วยความจำแฟลชจัดเก็บข้อมูลในอาร์เรย์ของเซลล์หน่วยความจำที่สร้างจากทรานซิสเตอร์แบบ Floating-gate ในอุปกรณ์เซลล์ระดับเดียว (SLC) แต่ละเซลล์จะเก็บข้อมูลเพียงบิตเดียว อุปกรณ์เซลล์หลายระดับ (MLC) รวมถึงอุปกรณ์เซลล์สามระดับ (TLC) สามารถจัดเก็บได้มากกว่าหนึ่งบิตต่อเซลล์ Floating gate อาจเป็นสื่อกระแสไฟฟ้า (โดยทั่วไปคือโพลีซิลิคอนใน หน่วยความจำแฟลช SONOS)

[](https://upload.wikimedia.org/wikipedia/commons/2/2c/Flash_cell_structure.svg)

## **ประตูลอย MOSFET (Floating-gate MOSFET)**

ในหน่วยความจำแฟลช แต่ละเซลล์หน่วยความจำจะมีลักษณะคล้ายกับทรานซิสเตอร์แบบ field-effect ของโลหะออกไซด์–เซมิคอนดักเตอร์มาตรฐาน (MOSFET) ยกเว้นว่าทรานซิสเตอร์มีเกทสองเกทแทนที่จะเป็นหนึ่งเกท เซลล์สามารถมองได้ว่าเป็นสวิตช์ไฟฟ้าที่กระแสไหลระหว่างขั้วทั้งสอง (แหล่งจ่ายและท่อระบาย) และถูกควบคุมโดย Floating gate (FG) และ control gate (CG)

CG นั้นคล้ายกับเกทในทรานซิสเตอร์ MOS อื่นๆ แต่ด้านล่างมี FG ที่หุ้มด้วยชั้นออกไซด์โดยรอบ FG ถูกสอดแทรกระหว่างช่อง CG และ MOSFET เนื่องจาก FG ถูกแยกด้วยไฟฟ้าโดยชั้นฉนวน อิเล็กตรอนที่วางอยู่บนนั้นจึงติดอยู่ เมื่อ FG ถูกประจุด้วยอิเล็กตรอน ประจุนี้จะคัดกรองสนามไฟฟ้าจาก CG ดังนั้นจึงเป็นการเพิ่มแรงดันธรณีประตู (VT1) ของเซลล์

ซึ่งหมายความว่าขณะนี้ต้องใช้แรงดันไฟฟ้าที่สูงขึ้น (VT2) กับ CG เพื่อให้ช่องนำไฟฟ้า เพื่ออ่านค่าจากทรานซิสเตอร์แรงดันไฟปานกลางระหว่างแรงดันไฟฟ้าเกณฑ์

หากช่องสัญญาณดำเนินการที่แรงดันไฟฟ้าระดับกลางนี้ FG จะต้องไม่ถูกชาร์จ (หากมีการชาร์จ เราจะไม่ได้รับการนำไฟฟ้าเนื่องจากแรงดันไฟฟ้าปานกลางน้อยกว่า VT2) และด้วยเหตุนี้ "1" ทางตรรกะจึงถูกเก็บไว้ในเกต หากช่องสัญญาณไม่ทำงานที่แรงดันไฟฟ้าปานกลาง แสดงว่ามีการชาร์จ FG และด้วยเหตุนี้ จึงเก็บตรรกะ "0" ไว้ในเกต การมีอยู่ของตรรกะ "0" หรือ "1" นั้นรับรู้ได้โดยการพิจารณาว่ามีกระแสไหลผ่านทรานซิสเตอร์หรือไม่เมื่อแรงดันไฟฟ้าระดับกลางถูกยืนยันบน CGในอุปกรณ์เซลล์แบบหลายระดับ ซึ่งเก็บมากกว่าหนึ่งบิตต่อเซลล์ จะตรวจจับปริมาณกระแสไฟ (แทนที่จะเป็นเพียงการมีอยู่หรือไม่มีอยู่) เพื่อกำหนดระดับประจุบน FG ได้แม่นยำยิ่งขึ้น

Floating gate(ประตูลอย) MOSFETs ตั้งชื่ออย่างนั้นเพราะมีชั้นอุโมงค์ออกไซด์ที่เป็นฉนวนไฟฟ้าระหว่าง Floating gate กับซิลิกอน ดังนั้นประตูจึง "Floats(ลอย)" เหนือซิลิกอน ออกไซด์ทำให้อิเล็กตรอนถูกกักขังไว้ที่ประตูลอย การเสื่อมสภาพหรือการสึกหรอ (และความทนทานที่จำกัดของหน่วยความจำแฟลชแบบลอยตัว) เกิดขึ้นเนื่องจากสนามแม่เหล็กที่สูงมาก (10 ล้านโวลต์ต่อเซนติเมตร) ที่เกิดขึ้นจากออกไซด์

ความหนาแน่นของไฟฟ้าแรงสูงดังกล่าวสามารถทำลายพันธะอะตอมเมื่อเวลาผ่านไปในออกไซด์ที่ค่อนข้างบาง ค่อยๆ ลดระดับคุณสมบัติการเป็นฉนวนไฟฟ้าและปล่อยให้อิเล็กตรอนถูกกักขังและผ่านได้อย่างอิสระ (รั่ว) จากประตูลอยสู่ออกไซด์ เพิ่มโอกาสที่ข้อมูลจะสูญหาย เนื่องจากอิเล็กตรอน (ปริมาณที่ใช้เพื่อแสดงระดับประจุที่แตกต่างกัน ซึ่งแต่ละอิเล็กตรอนจะกำหนดให้กับบิตที่แตกต่างกันใน MLC Flash) โดยปกติแล้วจะอยู่ที่ประตูลอย นี่คือเหตุผลที่การเก็บรักษาข้อมูลลดลงและความเสี่ยงของการสูญเสียข้อมูลจะเพิ่มขึ้นตามการลดลงที่เพิ่มขึ้น

## **อุโมงค์ฟาวเลอร์–นอร์ดไฮม์ (Fowler–Nordheim tunneling)**

กระบวนการย้ายอิเล็กตรอนจากประตูควบคุมไปยังประตูลอยเรียกว่าอุโมงค์ Fowler–Nordheim และเปลี่ยนแปลงลักษณะเฉพาะของเซลล์โดยพื้นฐานโดยการเพิ่มแรงดันธรณีประตูของ MOSFET ในทางกลับกันสิ่งนี้จะเปลี่ยนกระแสแหล่งจ่ายที่ไหลผ่านทรานซิสเตอร์สำหรับแรงดันเกตที่กำหนด

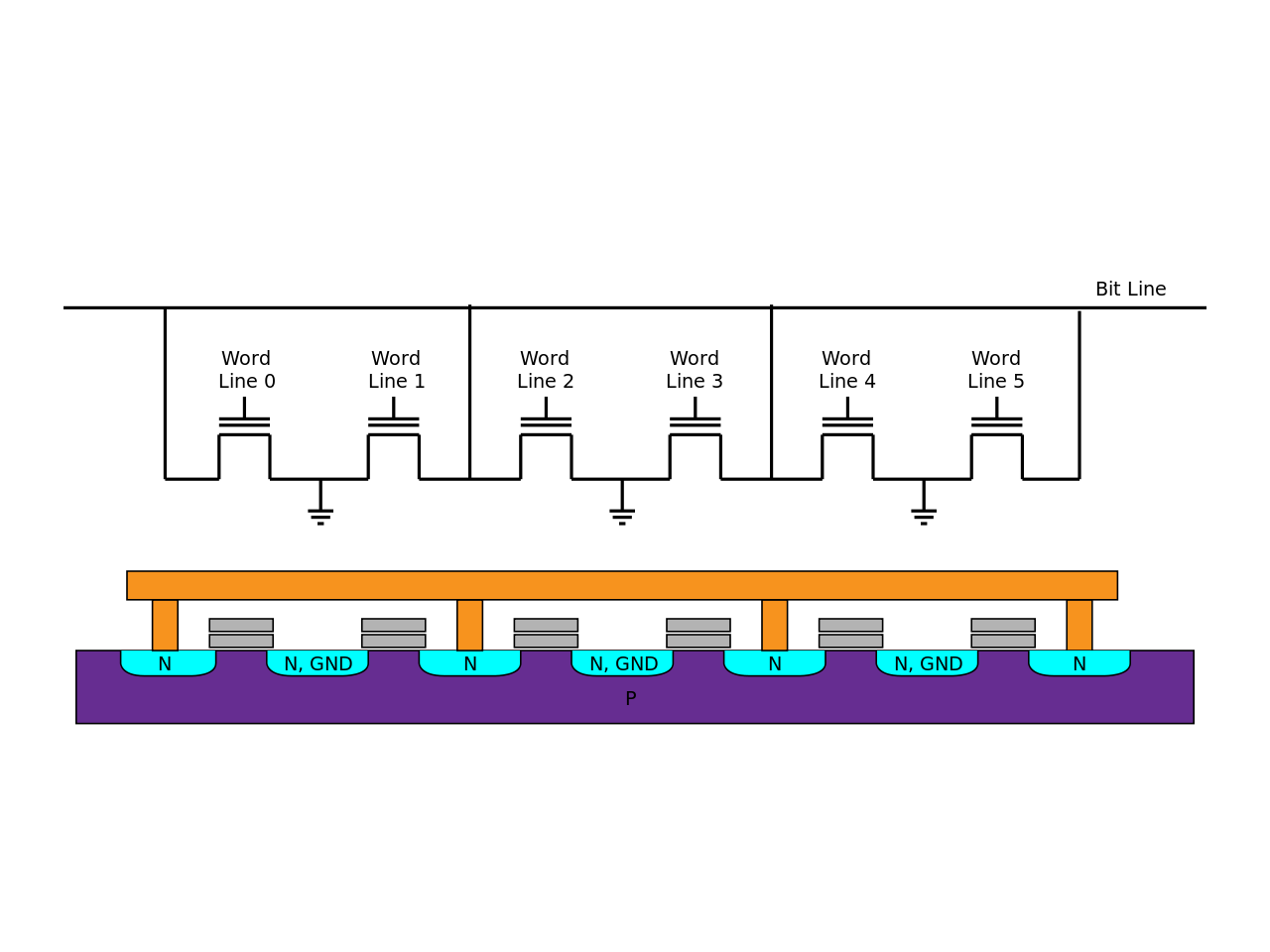
ซึ่งท้ายที่สุดแล้วจะใช้เพื่อเข้ารหัสค่าไบนารี เอฟเฟกต์อุโมงค์ Fowler-Nordheim สามารถย้อนกลับได้ ดังนั้นจึงสามารถเพิ่มหรือนำอิเล็กตรอนออกจากประตูลอยได้ ซึ่งกระบวนการที่เรียกกันทั่วไปว่าการเขียนและการลบ

## **ปั๊มชาร์จภายใน (Internal charge pumps)**

ชิปแฟลชแทบทั้งหมดในปัจจุบันต้องการแรงดันไฟฟ้าเพียงแหล่งเดียวและผลิตไฟฟ้าแรงสูงที่ต้องการโดยใช้ปั๊มประจุบนชิป พลังงานที่ใช้โดยชิปแฟลช NAND ขนาด 1.8 V สูญเสียไปในปั๊มชาร์จเอง เนื่องจากบูสต์คอนเวอร์เตอร์มีประสิทธิภาพมากกว่าปั๊มชาร์จโดยเนื้อแท้ นักวิจัยที่พัฒนา SSD พลังงานต่ำได้เสนอให้กลับไปใช้แรงดันไฟฟ้า Vcc/Vpp คู่ที่ใช้กับชิปแฟลชรุ่นก่อนๆ ทั้งหมด

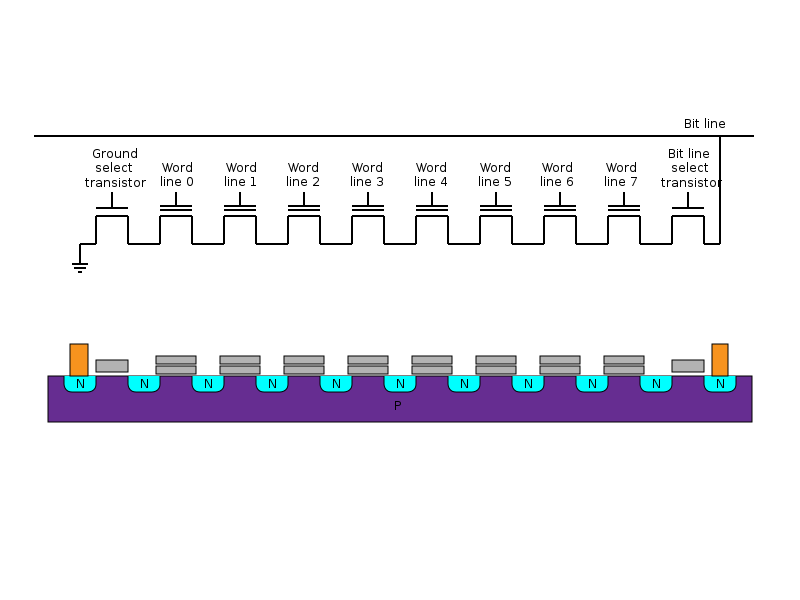
ซึ่งจะทำให้แรงดันไฟฟ้า Vpp สูงสำหรับชิปแฟลชทั้งหมดใน SSD ด้วยตัวเดียว ตัวแปลงบูสต์ภายนอกที่ใช้ร่วมกัน

## **NOR flash**

[](https://upload.wikimedia.org/wikipedia/commons/d/dd/NOR_flash_layout.svg)

ในแฟลช NOR แต่ละเซลล์มีปลายด้านหนึ่งเชื่อมต่อโดยตรงกับกราวด์ และปลายอีกด้านหนึ่งเชื่อมต่อโดยตรงกับเส้นบิต การจัดเรียงนี้เรียกว่า "แฟลช NOR" เพราะมันทำหน้าที่เหมือนเกต NOR: เมื่อบรรทัดคำใดบรรทัดหนึ่ง (เชื่อมต่อกับ CG ของเซลล์) ถูกทำให้สูง ทรานซิสเตอร์หน่วยเก็บข้อมูลที่เกี่ยวข้องจะดึงบรรทัดบิตเอาต์พุตให้ต่ำ แฟลช NOR ยังคงเป็นเทคโนโลยีทางเลือกสำหรับการใช้งานแบบฝังตัวที่ต้องการอุปกรณ์หน่วยความจำแบบไม่ลบเลือนแบบแยกส่วน คุณลักษณะเวลาแฝงในการอ่านต่ำของอุปกรณ์ NOR ช่วยให้สามารถใช้โค้ดได้โดยตรงและการจัดเก็บข้อมูลในผลิตภัณฑ์หน่วยความจำเดียว

## **NAND flash**

[](https://upload.wikimedia.org/wikipedia/commons/f/f5/Nand_flash_structure.svg)

แฟลช NAND ยังใช้ทรานซิสเตอร์แบบ Floating-gate แต่เชื่อมต่อในลักษณะที่คล้ายกับเกท NAND: ทรานซิสเตอร์หลายตัวเชื่อมต่อแบบอนุกรม และเส้นบิตจะถูกดึงให้ต่ำก็ต่อเมื่อบรรทัดคำทั้งหมดถูกดึงขึ้นสูง (เหนือทรานซิสเตอร์ VT) จากนั้นกลุ่มเหล่านี้จะเชื่อมต่อผ่านทรานซิสเตอร์เพิ่มเติมบางตัวกับอาร์เรย์ไลน์บิตแบบ NOR ในลักษณะเดียวกับที่ทรานซิสเตอร์ตัวเดียวเชื่อมโยงในแฟลช NOR

เมื่อเปรียบเทียบกับแฟลช NOR การแทนที่ทรานซิสเตอร์ตัวเดียวด้วยกลุ่มที่เชื่อมโยงแบบอนุกรมจะเพิ่มระดับของการกำหนดแอดเดรสเพิ่มเติม ในขณะที่แฟลช NOR อาจระบุหน่วยความจำทีละหน้า ตามด้วยคำ แฟลช NAND อาจระบุด้วยหน้า คำและบิต การกำหนดแอดเดรสระดับบิตเหมาะกับแอปพลิเคชันบิตซีเรียล (เช่น การจำลองฮาร์ดดิสก์) ซึ่งเข้าถึงได้ครั้งละหนึ่งบิตเท่านั้น ในทางกลับกัน แอปพลิเคชันแบบ Execute-in-place ต้องการการเข้าถึงทุกบิตในหนึ่งคำพร้อมกัน สิ่งนี้ต้องการที่อยู่ระดับคำ ไม่ว่าในกรณีใด โหมดการกำหนดแอดเดรสแบบบิตและแบบคำสามารถทำได้โดยใช้แฟลช NOR หรือ NAND

ในการอ่านข้อมูล ก่อนอื่นให้เลือกกลุ่มที่ต้องการ (ในลักษณะเดียวกับที่เลือกทรานซิสเตอร์ตัวเดียวจากอาร์เรย์ NOR) ถัดไป บรรทัดคำส่วนใหญ่จะถูกดึงขึ้นเหนือ VT ของบิตที่ตั้งโปรแกรมไว้ ในขณะที่บรรทัดหนึ่งถูกดึงขึ้นไปเหนือ VT ของบิตที่ถูกลบ กลุ่มอนุกรมจะดำเนินการ (และดึงบรรทัดบิตต่ำ) หากไม่ได้ตั้งโปรแกรมบิตที่เลือกไว้

แม้จะมีทรานซิสเตอร์เพิ่มเติม แต่การลดสายกราวด์และสายบิตช่วยให้มีเลย์เอาต์ที่หนาแน่นขึ้นและความจุต่อชิปที่มากขึ้น (จริง ๆ แล้วสายกราวด์และสายบิตกว้างกว่าเส้นในไดอะแกรมมาก) นอกจากนี้ แฟลช NAND มักจะได้รับอนุญาตให้มีข้อผิดพลาดจำนวนหนึ่ง (แฟลช NOR เช่นเดียวกับที่ใช้สำหรับ ROM ของ BIOS คาดว่าจะเป็น ปราศจากข้อผิดพลาด) ผู้ผลิตพยายามเพิ่มปริมาณพื้นที่เก็บข้อมูลที่ใช้งานได้สูงสุดโดยลดขนาดของทรานซิสเตอร์ลง เซลล์แฟลช NAND สามารถอ่านได้โดยการวิเคราะห์การตอบสนองต่อแรงดันไฟฟ้าต่างๆ

## **Vertical NAND**

หน่วยความจำ NAND แนวตั้ง (V-NAND) หรือ 3D NAND จะสแต็คเซลล์หน่วยความจำในแนวตั้งและใช้สถาปัตยกรรมแฟลชดักประจุ เลเยอร์แนวตั้งช่วยให้มีความหนาแน่นบิตของพื้นที่ที่ใหญ่ขึ้นโดยไม่ต้องใช้เซลล์แต่ละเซลล์ที่เล็กกว่า

นอกจากนี้ยังจำหน่ายภายใต้เครื่องหมายการค้า BiCS Flash ซึ่งเป็นเครื่องหมายการค้าของ Kioxia Corporation (อดีต Toshiba Memory Corporation) 3D NAND ได้รับการประกาศครั้งแรกโดยโตชิบาในปี 2550 V-NAND ถูกผลิตขึ้นเชิงพาณิชย์ครั้งแรกโดย Samsung Electronics ในปี 2556

**โครงสร้าง**

V-NAND ใช้รูปทรงแฟลชกับดักประจุ (ซึ่งเปิดตัวในเชิงพาณิชย์ในปี 2545 โดย AMD และ Fujitsu) ซึ่งเก็บประจุไว้บนฟิล์มซิลิคอนไนไตรด์ที่ฝังอยู่ ฟิล์มดังกล่าวมีความทนทานต่อจุดบกพร่องและสามารถทำให้หนาขึ้นเพื่อเก็บอิเล็กตรอนจำนวนมากขึ้นได้ V-NAND ห่อหุ้มเซลล์ดักจับประจุระนาบให้อยู่ในรูปทรงกระบอก

ในปี 2020 หน่วยความจำ 3D NAND Flash โดย Micron และ Intel จะใช้เกตแบบลอยแทน อย่างไรก็ตาม หน่วยความจำ 3D NAND ที่เลเยอร์ 128 และสูงกว่านั้นใช้โครงสร้างกับดักประจุแบบเดิม เนื่องจากการล่มสลายของความร่วมมือระหว่าง Micron และ Intel แท่นชาร์จ 3D NAND Flash นั้นบางกว่าประตูลอย 3D NAND ใน Float Gate 3D NAND เซลล์หน่วยความจำจะถูกแยกออกจากกันโดยสิ้นเชิง ในขณะที่ 3D NAND ของกับดักประจุ กลุ่มเซลล์หน่วยความจำในแนวตั้งใช้วัสดุซิลิกอนไนไตรด์เดียวกัน

เซลล์หน่วยความจำแต่ละเซลล์ประกอบด้วยชั้นโพลีซิลิคอนที่มีระนาบหนึ่งชั้นซึ่งมีรูที่เต็มไปด้วยกระบอกสูบแนวตั้งที่มีศูนย์กลางหลายอัน พื้นผิวโพลีซิลิคอนของรูทำหน้าที่เป็นอิเล็กโทรดเกท กระบอกสูบซิลิกอนไดออกไซด์ที่อยู่นอกสุดทำหน้าที่เป็นไดอิเล็กตริกเกตล้อมรอบกระบอกสูบซิลิกอนไนไตรด์ที่เก็บประจุ ในทางกลับกันก็ปิดถังซิลิกอนไดออกไซด์เป็นไดอิเล็กตริกแบบอุโมงค์ที่ล้อมรอบแกนกลางของตัวนำโพลีซิลิกอนซึ่งทำหน้าที่เป็นช่องทางนำไฟฟ้า

เซลล์หน่วยความจำในชั้นแนวตั้งที่ต่างกันไม่รบกวนซึ่งกันและกัน เนื่องจากประจุไม่สามารถเคลื่อนที่ในแนวตั้งผ่านสื่อเก็บซิลิคอนไนไตรด์ได้ และสนามไฟฟ้าที่เกี่ยวข้องกับเกตจะถูกจำกัดอย่างใกล้ชิดภายในแต่ละชั้น คอลเล็กชันแนวตั้งจะเหมือนกันทางไฟฟ้ากับกลุ่มที่ลิงก์อนุกรมซึ่งมีการกำหนดค่าหน่วยความจำแฟลช NAND แบบธรรมดา

# **ข้อจำกัด**

## **บล็อกการลบ**

ข้อจำกัดอย่างหนึ่งของหน่วยความจำแฟลชคือ แม้ว่าจะสามารถอ่านหรือตั้งโปรแกรมทีละไบต์หรือคำในรูปแบบการเข้าถึงโดยสุ่ม แต่ก็สามารถลบได้ครั้งละหนึ่งบล็อกเท่านั้น

โดยทั่วไปจะตั้งค่าบิตทั้งหมดในบล็อกเป็น 1 โดยเริ่มจากบล็อกที่เพิ่งลบใหม่ ตำแหน่งใดๆ ภายในบล็อกนั้นสามารถตั้งโปรแกรมได้ อย่างไรก็ตาม เมื่อบิตถูกตั้งค่าเป็น 0 โดยการลบบล็อกทั้งหมดเท่านั้น สามารถเปลี่ยนกลับเป็น 1 ได้ กล่าวอีกนัยหนึ่ง หน่วยความจำแฟลช (โดยเฉพาะแฟลช NOR) ให้การดำเนินการอ่านและเขียนโปรแกรมโดยสุ่มเข้าถึงแต่ไม่มีการสุ่มตามอำเภอใจ - เข้าถึงการดำเนินการเขียนใหม่หรือลบ

อย่างไรก็ตาม ตำแหน่งสามารถเขียนใหม่ได้ตราบใดที่ค่าใหม่เป็น 0 บิตเป็น superset ของค่าที่เขียนทับ ตัวอย่างเช่น ค่า nibble อาจถูกลบไปที่ 1111 จากนั้นเขียนเป็น 1110 การเขียนที่ต่อเนื่องไปยัง nibble นั้นสามารถเปลี่ยนเป็น 1,010 จากนั้น 0010 และสุดท้ายคือ 0000

โดยพื้นฐานแล้ว การลบจะตั้งค่าบิตทั้งหมดเป็น 1 และการเขียนโปรแกรมสามารถล้างบิตเป็น 0 เท่านั้นระบบไฟล์บางระบบที่ออกแบบมาสำหรับอุปกรณ์แฟลชใช้ประโยชน์จากความสามารถในการเขียนซ้ำนี้ เช่น Yaffs1 เพื่อแสดงข้อมูลเมตาของเซกเตอร์ ระบบไฟล์แฟลชอื่นๆ เช่น YAFFS2 ไม่เคยใช้ความสามารถ "เขียนซ้ำ" นี้เลย เนื่องจากระบบทำงานพิเศษหลายอย่างเพื่อให้เป็นไปตามกฎ "เขียนครั้งเดียว"

แม้ว่าโครงสร้างข้อมูลในหน่วยความจำแฟลชจะไม่สามารถอัปเดตในลักษณะทั่วไปได้อย่างสมบูรณ์ แต่สิ่งนี้ทำให้สมาชิกสามารถ "ลบ" โดยทำเครื่องหมายว่าไม่ถูกต้อง เทคนิคนี้อาจจำเป็นต้องแก้ไขสำหรับอุปกรณ์เซลล์หลายระดับ โดยที่เซลล์หน่วยความจำหนึ่งเซลล์มีมากกว่าหนึ่งบิต

อุปกรณ์แฟลชทั่วไป เช่น แฟลชไดรฟ์ USB และการ์ดหน่วยความจำมีเฉพาะอินเทอร์เฟซระดับบล็อก หรือเลเยอร์การแปลแฟลช (FTL) ซึ่งเขียนไปยังเซลล์อื่นในแต่ละครั้งเพื่อให้ระดับการสึกหรอของอุปกรณ์ สิ่งนี้จะป้องกันการเขียนที่เพิ่มขึ้นภายในบล็อก อย่างไรก็ตาม มันช่วยให้อุปกรณ์ไม่เสื่อมสภาพก่อนเวลาอันควรจากรูปแบบการเขียนที่เข้มข้น

## **หน่วยความจำสวมใส่ (Memory wear)**

ข้อจำกัดอีกประการหนึ่งคือหน่วยความจำแฟลชมีจำนวนโปรแกรมที่จำกัด – รอบการลบ (โดยทั่วไปเขียนเป็นรอบ P/E) ผลิตภัณฑ์แฟลชที่มีจำหน่ายทั่วไปส่วนใหญ่รับประกันว่าสามารถทนต่อรอบ P/E ได้ประมาณ 100,000 รอบ ก่อนที่การสึกหรอจะทำให้ความสมบูรณ์ของการจัดเก็บลดลง[80] Micron Technology และ Sun Microsystems ได้ประกาศชิปหน่วยความจำแฟลช SLC NAND ที่มีอัตรา 1,000,000 P/E รอบในวันที่ 17 ธันวาคม 2551

จำนวนรอบที่รับประกันอาจใช้เฉพาะกับบล็อกศูนย์ (เช่นเดียวกับอุปกรณ์ TSOP NAND) หรือกับบล็อกทั้งหมด (เช่นใน NOR) ผลกระทบนี้จะลดลงในเฟิร์มแวร์ชิปบางตัวหรือไดรเวอร์ระบบไฟล์โดยการนับบล็อกการเขียนและการแมปใหม่แบบไดนามิกเพื่อกระจายการดำเนินการเขียนระหว่างเซกเตอร์ เทคนิคนี้เรียกว่าการปรับระดับการสึกหรอ อีกวิธีหนึ่งคือดำเนินการตรวจสอบการเขียนและทำการแมปใหม่ไปยังเซกเตอร์สำรองในกรณีที่เกิดข้อผิดพลาดในการเขียน ซึ่งเป็นเทคนิคที่เรียกว่าการจัดการบล็อกที่ไม่ดี (BBM)

สำหรับอุปกรณ์พกพาสำหรับผู้บริโภค เทคนิคการจัดการเหล่านี้มักจะช่วยยืดอายุหน่วยความจำแฟลชให้ยาวนานกว่าอายุของอุปกรณ์เอง และข้อมูลบางส่วนอาจสูญหายได้ในแอปพลิเคชันเหล่านี้

อย่างไรก็ตาม สำหรับการจัดเก็บข้อมูลที่มีความน่าเชื่อถือสูง ไม่แนะนำให้ใช้หน่วยความจำแฟลชที่จะต้องผ่านรอบการเขียนโปรแกรมจำนวนมาก ข้อจำกัดนี้ไม่มีความหมายสำหรับแอปพลิเคชันที่ 'อ่านอย่างเดียว' เช่น ธินไคลเอ็นต์และเราเตอร์ ซึ่งถูกตั้งโปรแกรมเพียงครั้งเดียวหรือมากที่สุดสองสามครั้งในช่วงอายุการใช้งาน

## **การรบกวนการอ่าน (Read disturb)**

วิธีที่ใช้ในการอ่านหน่วยความจำแฟลช NAND อาจทำให้เซลล์ใกล้เคียงในบล็อกหน่วยความจำเดียวกันเปลี่ยนแปลงเมื่อเวลาผ่านไป (กลายเป็นโปรแกรม) สิ่งนี้เรียกว่าการรบกวนการอ่าน จำนวนเกณฑ์การอ่านโดยทั่วไปอยู่ในการอ่านหลายแสนครั้งระหว่างการดำเนินการลบที่แทรกแซง หากอ่านอย่างต่อเนื่องจากเซลล์หนึ่ง เซลล์นั้นจะไม่ล้มเหลว แต่จะอ่านเซลล์รอบข้างหนึ่งเซลล์ในการอ่านครั้งถัดไป

เพื่อหลีกเลี่ยงปัญหาการรบกวนการอ่าน โดยทั่วไปตัวควบคุมแฟลชจะนับจำนวนการอ่านทั้งหมดไปยังบล็อกหนึ่งๆ นับตั้งแต่การลบครั้งล่าสุด เมื่อการนับเกินขีดจำกัดเป้าหมาย บล็อกที่ได้รับผลกระทบจะถูกคัดลอกไปยังบล็อกใหม่ ลบออก แล้วปล่อยไปยังบล็อกพูล บล็อกเดิมดีเหมือนใหม่หลังจากลบ อย่างไรก็ตาม หากตัวควบคุมแฟลชไม่เข้าไปแทรกแซงตามเวลา จะเกิดข้อผิดพลาดในการอ่านข้อมูลโดยที่ข้อมูลอาจสูญหายได้ หากข้อผิดพลาดมีมากเกินไปที่จะแก้ไขด้วยรหัสการแก้ไขข้อผิดพลาด

**เอฟเฟกต์เอ็กซ์เรย์ (X-ray effects)**

แฟลชไอซีส่วนใหญ่มาในแพ็คเกจ ball grid array (BGA) และแม้แต่วงจรที่ไม่ได้ติดตั้งบน PCB ถัดจากแพ็คเกจ BGA อื่นๆ หลังจากการประกอบ PCB บอร์ดที่มีแพ็คเกจ BGA มักจะถูกเอ็กซ์เรย์เพื่อดูว่าลูกบอลกำลังเชื่อมต่อกับแพดที่เหมาะสมหรือไม่ หรือหาก BGA จำเป็นต้องทำใหม่ รังสีเอกซ์เหล่านี้สามารถลบบิตที่ตั้งโปรแกรมไว้ในชิปแฟลช (แปลงบิต "0" ที่ตั้งโปรแกรมไว้เป็นบิต "1" ที่ถูกลบ) บิตที่ถูกลบ (บิต "1") จะไม่ได้รับผลกระทบจากรังสีเอกซ์

ผู้ผลิตบางรายกำลังผลิตอุปกรณ์หน่วยความจำ USB แบบ SD[90] ที่กันรังสีเอ็กซ์

# **การเข้าถึงระดับต่ำ**

อินเทอร์เฟซระดับต่ำกับชิปหน่วยความจำแฟลชแตกต่างจากหน่วยความจำประเภทอื่นๆ เช่น DRAM, ROM และ EEPROM ซึ่งรองรับการเปลี่ยนแปลงบิต (ทั้งศูนย์ถึงหนึ่งและหนึ่งถึงศูนย์) และการเข้าถึงแบบสุ่มผ่านแอดเดรสบัสที่เข้าถึงได้จากภายนอก

หน่วยความจำ NOR มีบัสที่อยู่ภายนอกสำหรับการอ่านและการเขียนโปรแกรม สำหรับหน่วยความจำ NOR การอ่านและการตั้งโปรแกรมเป็นการเข้าถึงแบบสุ่ม และการปลดล็อกและการลบจะเป็นแบบบล็อก สำหรับหน่วยความจำ NAND การอ่านและการตั้งโปรแกรมเป็นแบบเพจ และการปลดล็อกและการลบจะเป็นแบบบล็อก

## **NOR memories**

การอ่านจากแฟลช NOR นั้นคล้ายกับการอ่านจากหน่วยความจำเข้าถึงโดยสุ่ม โดยมีการระบุที่อยู่และบัสข้อมูลไว้อย่างถูกต้อง ด้วยเหตุนี้ ไมโครโปรเซสเซอร์ส่วนใหญ่จึงสามารถใช้หน่วยความจำแฟลช NOR เป็นหน่วยความจำแบบดำเนินการในสถานที่ (XIP) ได้ ซึ่งหมายความว่าโปรแกรมที่จัดเก็บไว้ในแฟลช NOR สามารถทำงานได้โดยตรงจากแฟลช NOR โดยไม่จำเป็นต้องคัดลอกลงใน RAM ก่อน แฟลช NOR อาจถูกตั้งโปรแกรมในลักษณะการเข้าถึงแบบสุ่มคล้ายกับการอ่าน การเขียนโปรแกรมเปลี่ยนบิตจากตรรกะหนึ่งเป็นศูนย์ บิตที่เป็นศูนย์อยู่แล้วจะไม่เปลี่ยนแปลง การลบจะต้องเกิดขึ้นทีละบล็อก และรีเซ็ตบิตทั้งหมดในบล็อกที่ถูกลบกลับไปเป็นบล็อกเดียว ขนาดบล็อกทั่วไปคือ 64, 128 หรือ 256 KiB

การจัดการบล็อกที่ไม่ดีเป็นคุณสมบัติที่ค่อนข้างใหม่ในชิป NOR ในอุปกรณ์ NOR รุ่นเก่าที่ไม่รองรับการจัดการบล็อกที่ไม่ดี ซอฟต์แวร์หรือไดรเวอร์อุปกรณ์ที่ควบคุมชิปหน่วยความจำจะต้องแก้ไขสำหรับบล็อกที่เสื่อมสภาพ ไม่เช่นนั้นอุปกรณ์จะหยุดทำงานอย่างน่าเชื่อถือ

คำสั่งเฉพาะที่ใช้ในการล็อค ปลดล็อค โปรแกรม หรือลบหน่วยความจำ NOR นั้นแตกต่างกันไปสำหรับผู้ผลิตแต่ละราย เพื่อหลีกเลี่ยงความต้องการซอฟต์แวร์ไดรเวอร์เฉพาะสำหรับอุปกรณ์ทุกชิ้นที่ผลิต คำสั่ง Common Flash Memory Interface (CFI) พิเศษช่วยให้อุปกรณ์สามารถระบุตัวเองและพารามิเตอร์การทำงานที่สำคัญได้

นอกจากการใช้เป็น ROM เข้าถึงโดยสุ่มแล้ว แฟลช NOR ยังสามารถใช้เป็นอุปกรณ์จัดเก็บข้อมูลได้ด้วยการใช้ประโยชน์จากการเขียนโปรแกรมเข้าถึงโดยสุ่ม อุปกรณ์บางอย่างมีฟังก์ชันอ่านขณะเขียนเพื่อให้โค้ดทำงานต่อไปได้แม้ในขณะที่โปรแกรมหรือการดำเนินการลบเกิดขึ้นในเบื้องหลัง สำหรับการเขียนข้อมูลตามลำดับ โดยปกติชิป NOR แฟลชจะมีความเร็วในการเขียนที่ช้า เมื่อเทียบกับแฟลช NAND

แฟลช NOR ทั่วไปไม่ต้องการรหัสแก้ไขข้อผิดพลาด

## **NAND memories**

สถาปัตยกรรมแฟลช NAND เปิดตัวโดยโตชิบาในปี 1989[93] ความทรงจำเหล่านี้เข้าถึงได้เหมือนกับอุปกรณ์บล็อก เช่น ฮาร์ดดิสก์ แต่ละบล็อกประกอบด้วยหลายหน้า หน้าโดยทั่วไปจะมีขนาด 512 [94] 2,048 หรือ 4,096 ไบต์ ที่เกี่ยวข้องกับแต่ละหน้าคือสองสามไบต์ (โดยทั่วไปคือ 1/32 ของขนาดข้อมูล) ที่สามารถใช้สำหรับการจัดเก็บการตรวจสอบรหัสแก้ไขข้อผิดพลาด (ECC)

ขนาดบล็อกทั่วไป ได้แก่ :

32 หน้า 512+16 ไบต์สำหรับขนาดบล็อก (มีผล) 16 KiB

64 หน้า 2,048+64 ไบต์ต่อบล็อกขนาด 128 KiB[95]

64 หน้า 4,096+128 ไบต์ต่อบล็อกขนาด 256 KiB[96]

128 หน้า 4,096+128 ไบต์ต่อบล็อกขนาด 512 KiB

ในขณะที่อ่านและเขียนโปรแกรมบนพื้นฐานหน้า การลบสามารถทำได้บนพื้นฐานบล็อกเท่านั้น

อุปกรณ์ NAND ยังต้องการการจัดการบล็อกที่ไม่ดีโดยซอฟต์แวร์ไดรเวอร์อุปกรณ์หรือโดยชิปควบคุมแยกต่างหาก ตัวอย่างเช่น การ์ด SD มีวงจรควบคุมเพื่อดำเนินการจัดการบล็อกที่ไม่ดีและปรับระดับการสึกหรอ เมื่อซอฟต์แวร์ระดับสูงเข้าถึงบล็อกเชิงตรรกะ

บล็อกนั้นจะถูกจับคู่กับบล็อกจริงโดยไดรเวอร์อุปกรณ์หรือตัวควบคุม อาจมีการแบ่งบล็อกจำนวนหนึ่งบนชิปแฟลชเพื่อจัดเก็บตารางการแมปเพื่อจัดการกับบล็อกที่เสียหาย หรือระบบอาจตรวจสอบแต่ละบล็อกเมื่อเปิดเครื่องเพื่อสร้างแผนที่บล็อกที่ไม่ดีใน RAM ความจุของหน่วยความจำโดยรวมจะค่อยๆ ลดขนาดลงเมื่อมีบล็อกจำนวนมากขึ้นถูกทำเครื่องหมายว่าไม่ดี

NAND อาศัย ECC เพื่อชดเชยบิตที่อาจล้มเหลวโดยธรรมชาติระหว่างการทำงานของอุปกรณ์ปกติ ECC ทั่วไปจะแก้ไขข้อผิดพลาดหนึ่งบิตในแต่ละ 2048 บิต (256 ไบต์) โดยใช้ ECC 22 บิต หรือข้อผิดพลาดหนึ่งบิตในแต่ละ 4096 บิต (512 ไบต์) โดยใช้ ECC 24 บิต[98] หาก ECC ไม่สามารถแก้ไขข้อผิดพลาดระหว่างการอ่านได้ ระบบอาจยังตรวจพบข้อผิดพลาด เมื่อทำการลบหรือการทำงานของโปรแกรม อุปกรณ์สามารถตรวจจับบล็อคที่ไม่สามารถตั้งโปรแกรมหรือลบและทำเครื่องหมายว่าบล็อกนั้นไม่ถูกต้อง ข้อมูลจะถูกเขียนไปยังบล็อกอื่นที่ดีและมีการอัปเดตแผนที่บล็อกที่ไม่ดี

รหัส Hamming เป็น ECC ที่ใช้บ่อยที่สุดสำหรับแฟลช SLC NAND รหัส Reed-Solomon และรหัส BCH (รหัส Bose-Chaudhuri-Hocquenghem) มักใช้ ECC สำหรับแฟลช MLC NAND ชิปแฟลช MLC NAND บางตัวสร้างรหัสแก้ไขข้อผิดพลาด BCH ที่เหมาะสมภายใน

อุปกรณ์ NAND ส่วนใหญ่จัดส่งมาจากโรงงานโดยมีบล็อกที่ไม่ดี โดยทั่วไปแล้วสิ่งเหล่านี้จะถูกทำเครื่องหมายตามกลยุทธ์การทำเครื่องหมายบล็อกที่ไม่ถูกต้องที่ระบุ โดยการอนุญาตบล็อกที่ไม่ดี ผู้ผลิตจะได้ผลตอบแทนที่สูงกว่าที่จะเป็นไปได้หากบล็อกทั้งหมดต้องได้รับการตรวจสอบว่าดี ซึ่งช่วยลดต้นทุนแฟลช NAND ได้อย่างมากและลดความจุในการจัดเก็บของชิ้นส่วนลงเพียงเล็กน้อยเท่านั้น

เมื่อเรียกใช้ซอฟต์แวร์จากหน่วยความจำ NAND มักใช้กลยุทธ์หน่วยความจำเสมือน: เนื้อหาหน่วยความจำจะต้องถูกเพจหรือคัดลอกไปยัง RAM ที่แมปหน่วยความจำก่อนและดำเนินการที่นั่น (นำไปสู่การรวมกันระหว่าง NAND + RAM) หน่วยการจัดการหน่วยความจำ (MMU) ในระบบมีประโยชน์ แต่ก็สามารถทำได้ด้วยการซ้อนทับ ด้วยเหตุนี้ บางระบบจะใช้หน่วยความจำ NOR และ NAND ร่วมกัน โดยที่หน่วยความจำ NOR ที่เล็กกว่าจะถูกใช้เป็น ROM ของซอฟต์แวร์ และหน่วยความจำ NAND ที่ใหญ่กว่าจะถูกแบ่งพาร์ติชันด้วยระบบไฟล์เพื่อใช้เป็นพื้นที่จัดเก็บข้อมูลแบบไม่ลบเลือน

NAND สละข้อได้เปรียบในการเข้าถึงแบบสุ่มและดำเนินการแทน NOR NAND เหมาะที่สุดสำหรับระบบที่ต้องการการจัดเก็บข้อมูลที่มีความจุสูง มีความหนาแน่นสูงขึ้น ความจุที่มากขึ้น และต้นทุนที่ต่ำกว่า มีการลบที่เร็วขึ้น การเขียนตามลำดับ และการอ่านตามลำดับ

## **มาตรฐาน (Standardization)**

กลุ่มที่เรียกว่า Open NAND Flash Interface Working Group (ONFI) ได้พัฒนาอินเทอร์เฟซระดับต่ำที่ได้มาตรฐานสำหรับชิปแฟลช NAND ซึ่งช่วยให้สามารถทำงานร่วมกันได้ระหว่างอุปกรณ์ NAND ที่สอดคล้องจากผู้ขายรายต่างๆ ข้อมูลจำเพาะ ONFI เวอร์ชัน 1.0 เผยแพร่เมื่อวันที่ 28 ธันวาคม พ.ศ. 2549 โดยระบุ:

* อินเทอร์เฟซทางกายภาพมาตรฐาน (pinout) สำหรับแฟลช NAND ในแพ็คเกจ TSOP-48, WSOP-48, LGA-52 และ BGA-63
* ชุดคำสั่งมาตรฐานสำหรับการอ่าน เขียน และลบชิปแฟลช NAND
* กลไกสำหรับการระบุตัวเอง (เทียบได้กับคุณสมบัติการตรวจจับการมีอยู่ของซีเรียลของโมดูลหน่วยความจำ SDRAM)

กลุ่ม ONFI ได้รับการสนับสนุนจากผู้ผลิตแฟลช NAND รายใหญ่ ซึ่งรวมถึง Hynix, Intel, Micron Technology และ Numonyx ตลอดจนผู้ผลิตรายใหญ่ของอุปกรณ์ที่ใช้ชิปแฟลช NAND

ผู้ผลิตอุปกรณ์แฟลชรายใหญ่สองรายคือ Toshiba และ Samsung เลือกใช้อินเทอร์เฟซของการออกแบบของตนเองที่เรียกว่า Toggle Mode (และปัจจุบันคือ Toggle V2.0) อินเทอร์เฟซนี้ไม่สามารถใช้งานร่วมกับข้อกำหนดของ ONFI ได้ ผลที่ได้คือผลิตภัณฑ์ที่ออกแบบมาสำหรับอุปกรณ์ของผู้ขายรายหนึ่งอาจไม่สามารถใช้อุปกรณ์ของผู้ขายรายอื่นได้

กลุ่มผู้จำหน่าย รวมทั้ง Intel, Dell และ Microsoft ได้ก่อตั้งคณะทำงาน Non-Volatile Memory Host Controller Interface (NVMHCI) เป้าหมายของกลุ่มนี้คือการจัดหาอินเทอร์เฟซการเขียนโปรแกรมซอฟต์แวร์และฮาร์ดแวร์มาตรฐานสำหรับระบบย่อยหน่วยความจำแบบไม่ลบเลือน ซึ่งรวมถึงอุปกรณ์ "แฟลชแคช" ที่เชื่อมต่อกับบัส PCI Express

# **ความแตกต่างระหว่าง NOR และ NAND Flash**

แฟลช NOR และ NAND แตกต่างกันในสองวิธีที่สำคัญ:

* การเชื่อมต่อของเซลล์หน่วยความจำแต่ละเซลล์นั้นแตกต่างกั
* อินเทอร์เฟซสำหรับอ่านและเขียนหน่วยความจำแตกต่างกัน NOR อนุญาตการเข้าถึงแบบสุ่ม ในขณะที่ NAND อนุญาตเฉพาะการเข้าถึงหน้าเท่านั้น

แฟลช NOR และ NAND ได้ชื่อมาจากโครงสร้างของการเชื่อมต่อระหว่างเซลล์หน่วยความจำ[ต้องการอ้างอิง] ในแฟลช NOR เซลล์จะเชื่อมต่อแบบขนานกับบรรทัดบิต ทำให้เซลล์สามารถอ่านและตั้งโปรแกรมแยกกันได้

การเชื่อมต่อแบบขนานของเซลล์คล้ายกับการเชื่อมต่อแบบขนานของทรานซิสเตอร์ในเกท CMOS NOR ในแฟลช NAND เซลล์จะเชื่อมต่อแบบอนุกรม คล้ายกับเกท CMOS NAND การเชื่อมต่อแบบอนุกรมใช้พื้นที่น้อยกว่าการเชื่อมต่อแบบขนาน ซึ่งช่วยลดต้นทุนของแฟลช NAND โดยตัวมันเองไม่ได้ป้องกันไม่ให้เซลล์ NAND ถูกอ่านและตั้งโปรแกรมทีละรายการ

เซลล์แฟลช NOR แต่ละเซลล์มีขนาดใหญ่กว่าเซลล์แฟลช NAND – 10 F2 เทียบกับ 4 F2 – แม้ว่าจะใช้การผลิตอุปกรณ์เซมิคอนดักเตอร์เหมือนกันทุกประการ ดังนั้นทรานซิสเตอร์ หน้าสัมผัส ฯลฯ แต่ละตัวจึงมีขนาดเท่ากันทุกประการ เนื่องจากเซลล์แฟลช NOR ต้องใช้โลหะแยกต่างหาก ติดต่อแต่ละเซลล์

เนื่องจากการเชื่อมต่อแบบอนุกรมและการลบผู้ติดต่อ wordline กริดขนาดใหญ่ของเซลล์หน่วยความจำแฟลช NAND อาจใช้พื้นที่เพียง 60% ของพื้นที่เซลล์ NOR ที่เทียบเท่ากัน[105] (สมมติว่ามีความละเอียดของกระบวนการ CMOS เท่ากัน เช่น 130 นาโนเมตร, 90 นาโนเมตร หรือ 65 นาโนเมตร) นักออกแบบของแฟลช NAND ตระหนักดีว่าพื้นที่ของชิป NAND และด้วยเหตุนี้ต้นทุนจึงสามารถลดลงได้อีกโดยการถอดที่อยู่ภายนอกและวงจรบัสข้อมูลออก

แต่อุปกรณ์ภายนอกสามารถสื่อสารกับแฟลช NAND ผ่านคำสั่งที่เข้าถึงตามลำดับและการลงทะเบียนข้อมูล ซึ่งจะดึงและส่งออกข้อมูลที่จำเป็นภายใน ตัวเลือกการออกแบบนี้ทำให้การเข้าถึงหน่วยความจำแฟลช NAND แบบสุ่มเป็นไปไม่ได้ แต่เป้าหมายของแฟลช NAND คือการเปลี่ยนฮาร์ดดิสก์แบบกลไก ไม่ใช่เพื่อแทนที่ ROM

|  |  |  |
| --- | --- | --- |
| **Attribute** | **NAND** | **NOR** |
| Main application | File storage | Code execution |
| Storage capacity | High | Low |
| Cost per bit | Low |  |
| Active power | Low |  |
| Standby power |  | Low |
| Write speed | Fast |  |
| Read speed |  | Fast |
| Execute in place (XIP) | No | Yes |
| Reliability |  | High |

# **ระบบไฟล์แฟลช**

# **ความจุ**

# **อัตราการโอน**

# **แอปพลิเคชั่น**

# **อุตสาหกรรม**

# **ความสามารถในการปรับขนาดแฟลช**

# **อ้างอิง**

https://en.wikipedia.org/wiki/Flash\_memory

https://www.slideshare.net/ennael/dwmw2-kr201209

https://hypnocube.com/2014/11/flash-endurance-testing/

https://adreca.net/NAND-Flash-Data-Recovery-Cookbook.pdf

https://openwrt.org/docs/techref/flash.layout#types\_of\_flash\_memory