МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ

На правах рукопису

ГРУДАНОВ ОЛЕКСАНДР МИКОЛАЙОВИЧ

УДК 621.3.(043.5)

МЕТОДИ ПРОЄКТУВАННЯ ЕНЕРГОЕФЕКТИВНОЇ ДВОПОРТОВОЇ RFSRAM-ПАМ'ЯТІ ДЛЯ ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ

Спеціальність 172 «Телекомунікації та радіотехніка»
Галузь знань 17 «Електроніка та телекомунікації»

Подається на здобуття наукового ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____О.М. Груданов

Науковий керівник
доктор технічних наук, професор
Шутко Володимир Миколайович

Київ – 2026

АНОТАЦІЯ

Груданов О. М. Методи проєктування енергоефективної двопортової RFSRAM-пам'яті для телекомунікаційних систем. — Кваліфікаційна наукова праця на правах рукопису.

Дисертаційна робота на здобуття наукового ступеня доктора філософії за спеціальністю 172 — електронні комунікації та радіотехніка. — КАІ, Київ, 2026.

Дисертаційна робота присвячена розробленню методів проєктування енергоефективної двопортової пам'яті регістрового типу RFSRAM, призначеної для використання у складі телекомунікаційних систем, сигнальних процесорів, мережевих процесорів, модемних, радарних та інших спеціалізованих систем на кристалі. Актуальність роботи зумовлена тим, що в сучасних інтегральних схемах класу System-on-Chip значна частина площі та енергоспоживання припадає на вбудовані блоки пам'яті, які забезпечують швидкий доступ до операндів, проміжних результатів обробки сигналів, буферів і службових даних. У роботі розглянуто задачу проєктування двопортової RFSRAM-пам'яті з пріоритетом низького енергоспоживання за умови збереження статичної стійкості комірки, прийнятного часу доступу, працездатності в різних PVT-умовах та можливості урахування паразитних RC-параметрів після топологічної реалізації. Обґрунтовано, що просте зниження напруги живлення не є самодостатнім методом енергооптимізації, оскільки воно супроводжується деградацією SNM/WNM, зростанням впливу mismatch параметрів MOS-транзисторів та збільшенням чутливості до паразитних параметрів топології.

Об'єктом дослідження є процеси схемотехнічного проєктування, топологічної реалізації та післятопологічної перевірки двопортової RFSRAM-пам'яті. Предметом дослідження є методи, моделі та програмні засоби проєктування RFSRAM на основі 8T-комірки статичної оперативної пам'яті SRAM, орієнтовані

на зниження енергоспоживання, забезпечення запасу статичної стійкості та урахування статистичних і паразитних факторів.

У дисертаційній роботі запропоновано SNM/WNM-орієнтований підхід, використано статистичну модель розподілу SNM з урахуванням розкиду порогових напруг MOS-транзисторів, розроблено архітектурні рішення для двопортової RFSRAM-пам'яті та показано застосування post-layout SPICE-моделювання з урахуванням екстрагованих паразитних RC-параметрів. Для оцінювання характеристик пам'яті використано схемотехнічне моделювання, аналіз butterfly-кривих, статистичне оцінювання SNM-розподілу, розрахунок середнього динамічного струму, timing/power-характеризацію, формування LIB-описів, вбудоване самотестування BIST та експериментальну перевірку тестових зразків у кремнії. Схемотехнічний аналіз охоплює передавальні характеристики інверторів зі зворотним зв'язком та еквівалентні схеми пам'яті з паразитними опорами і ємностями.

Наукова новизна роботи полягає у поєднанні методів статистичного оцінювання стабільності 8T-комірки, вибору геометричних розмірів транзисторів для зниження енергоспоживання, post-layout характеризації та експериментальної перевірки у єдиному маршруті проєктування енергоефективної двопортової RFSRAM-пам'яті. Практичне значення результатів полягає у можливості їх застосування при створенні IP-блоків вбудованої пам'яті для телекомунікаційних, радіотехнічних, сенсорних, автономних та оборонних систем, де критичними є низьке енергоспоживання, гарантований час доступу та надійна робота при знижених напругах живлення.

Ключові слова: телекомунікаційні системи, радіотехніка, радіоелектронне обладнання, радари, частота, електромагнітна хвиля, смуга пропускання, цифрова обробка сигналів, експлуатація, система на кристалі, схемотехнічне

моделювання, моделювання електронних схем, еквівалентна схема, зворотний зв’язок, інтегральна схема, системи експлуатації, SoC, напівпровідники, метали, діелектрики, вбудована пам’ять, RFSRAM, регістровий файл, 8T-комірка, низьке енергоспоживання, SNM, WNM, статистичне моделювання, надійність, розкид порогової напруги, mismatch, PVT-умови, SPICE-моделювання, рівняння, post-layout моделювання, паразитні RC-параметри, BIST, верифікація у кремнії.

ABSTRACT

Grudanov O. M. Design Methods for Energy-Efficient Dual-Port RFSRAM Memory for Telecommunication Systems. — Qualifying scientific work as a manuscript.

Dissertation for the degree of Doctor of Philosophy in specialty 172 — Electronic Communications and Radio Engineering. — Kyiv Aviation Institute, Kyiv, 2026.

The dissertation is devoted to the development of design methods for energy-efficient dual-port register-file memory RFSRAM (Register File Static Random Access Memory) intended for use in telecommunication systems, signal processors, network processors, modems, radar systems and other specialized systems-on-chip. The relevance of the research is determined by the fact that in modern System-on-Chip integrated circuits a significant part of chip area and power consumption is associated with embedded memory blocks that provide fast access to operands, intermediate signal-processing data, buffers and service information.

The work addresses the problem of designing dual-port RFSRAM memory with low-power priority while preserving cell stability, acceptable access time, reliable operation under different PVT conditions and the possibility of accounting for parasitic RC parameters after layout implementation. It is shown that supply-voltage reduction alone is not sufficient for memory power optimization, since it causes SNM/WNM

degradation, increases sensitivity to MOS-transistor mismatch and makes parasitic layout effects more significant.

The object of the research is the set of processes related to circuit design, layout implementation and post-layout verification of dual-port RFSRAM memory. The subject of the research is the set of methods, models and software tools for designing RFSRAM memory based on an 8T SRAM cell, aimed at reducing power consumption, ensuring static stability and accounting for statistical and parasitic effects.

The dissertation proposes an SNM/WNM-oriented approach to 8T cell sizing, applies a statistical SNM-distribution model considering MOS-transistor threshold-voltage variation, develops architectural solutions for dual-port RFSRAM memory and demonstrates post-layout SPICE simulation with extracted RC parameters. The research uses SPICE-based electronic circuit simulation, butterfly-curve analysis, statistical SNM evaluation, average dynamic-current calculation, timing/power characterization, LIB description generation, BIST testing and experimental silicon verification. The circuit analysis covers the transfer characteristics of feedback-connected inverters and equivalent memory circuits incorporating parasitic resistances and capacitances.

The scientific novelty of the work consists in combining statistical stability evaluation of an 8T cell, low-power sizing, post-layout RC characterization and experimental verification into a unified design flow for energy-efficient dual-port RFSRAM memory. The practical significance of the obtained results lies in their applicability to embedded memory IP blocks for telecommunication, radio-engineering, sensor, autonomous and defense systems where low power consumption, guaranteed access time and reliable low-voltage operation are critical.

Keywords: electronic communications, telecommunication systems, radar systems, exploitation, radio engineering, radio-electronic equipment, electromagnetic wave,

frequency, bandwidth, System-on-Chip, SoC, circuit simulation, electronic circuit modeling, equivalent circuit, feedback, integrated circuit, Digital Signal Processing, operating conditions, embedded memory, RFSRAM, register file, dual-port memory, 8T SRAM cell, low power consumption, reliability, metals, semiconductor characteristics, dielectrics, SNM, WNM, statistical modeling, threshold-voltage variation, mismatch, PVT conditions, SPICE simulation, equations, post-layout simulation, parasitic RC parameters, BIST, silicon verification.

Список публікацій здобувача за темою дисертації

1. Груданов О. М. 8Т комірка пам’яті регістрового файлу для технологій 180 нм із низькою напругою живлення // Вісті вищих учбових закладів. Радіоелектроніка. 2023. Т. 66, № 12. С. 736–745. https://doi.org/10.20535/S0021347023100059. ISSN 2307-6011 (Фахове видання; Scopus, Q4).
2. Grudanov O. M. Stability Parameters of Register File Bit Cell with Low Power Consumption Priority // Electronics and Control Systems. 2023. Vol. 3, No. 77. P. 40–46. https://doi.org/10.18372/1990-5548.77.17963. ISSN 1990-5548 (Фахове видання).
3. Grudanov O. M., Grudanov M. B. Smart-CX — Method of Extraction of Parasitic Capacitances in ICs // Solid-State Electronics. 2024. Vol. 221–222. Article 109007. https://doi.org/10.1016/j.sse.2024.109007. ISSN 1879-2405 (Scopus, Q2).
4. Grudanov O. M., Grudanov M. B., Shutko V. M. Rule-Based Layout-Driven Parasitic RC Extraction for Post-Layout SPICE Simulation of CMOS ICs // Chips. 2026. Vol. 5, No. 2. Article 13. https://doi.org/10.3390/chips5020013. ISSN 2674-0729 (Scopus, Q2).

5. Grudanov O. M., Grudanov M. B., Shutko V. M. Impact of Hardware Test Equipment on Frequency Characterization of 130-nm Register File // 2026 IEEE 43rd International Conference on Electronics and Nanotechnology (ELNANO), Kyiv, Ukraine, 2026. P. 364–368. https://doi.org/10.1109/ELNANO63396.2026.11601742. ISSN 3065-5633 (Scopus).

Матеріали доповідей на конференціях, які засвідчують апробацію результатів дисертації

1. Grudanov O. M. Built-in Self Test for Register File Memory Compiler // Flight. Modern Problems of Science: abstracts of the XXI International Scientific-Practical Conference of Applicants for Higher Education and Young Scientists. National Aviation University. Kyiv, 2024. P. 244–245. UDC 621.315-022.538(045).
2. Grudanov O. M. Power Consumption Calculation of Register File Memory // Інтегровані інтелектуальні робототехнічні комплекси (ІІРТК-2024): матеріали XVII Міжнародної науково-практичної конференції, Київ, Україна, 21–22 травня 2024 р. Київ, 2024. С. 444–446. УДК 621.315-022.538(045).
3. Grudanov O. M., Shutko V. M. Minimizing Leakage Current in 64K SRAM Using High-Threshold Voltage Transistors // ABIA-2025: матеріали XVII Міжнародної науково-технічної конференції, Державний університет «Київський авіаційний інститут», Київ, Україна, 22–24 квітня 2025 р. Київ, 2025. 5 с. UDC 621.382.323.002.2.

ЗМІСТ

ВСТУП 16
РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМИ ПРОЄКТУВАННЯ ЕНЕРГОЕФЕКТИВНОЇ RFSRAM-ПАМ'ЯТІ. 28
1.1. Роль вбудованої пам'яті та регістрових файлів у сучасних SoC 28
1.2. Компілятори пам'яті як засіб автоматизованого проєктування RFSRAM 31
1.3. Методи зниження енергоспоживання SRAM/RFSRAM 33
1.3.1. Зниження напруги живлення як базовий метод енергооптимізації 34
1.3.2. Оптимізація розмірів транзисторів і співвідношень CR/PR 35
1.3.3. Перехід від 6T до 8T, 9T, 10T та інших SRAM-комірок 36
1.3.4. Read-decoupled архітектури та 8T-комірки 37
1.3.5. Зменшення амплітуди перемикання бітових ліній і оптимізація precharge 38
1.3.6. Зниження leakage за рахунок transistor stacking 39
1.3.7. Використання assist circuits для read/write stability 40
1.3.8. Статистичні методи оцінки стабільності та мінімальної напруги 41
1.4. Вплив технологічного масштабування та статистичних варіацій на стабільність комірки пам'яті 41
1.4.1 Варіації роботи виходу Фермі затвора (Work Function Variation) 43
1.4.2 Випадкові флуктуації домішок: 44
1.4.3 Варіації довжини каналу та ефект DIBL 45
1.4.4 Варіації ширини каналу (Width Variation) 46
1.5. Урахування паразитних RC-параметрів у post-layout моделюванні пам'яті. 47
РОЗДІЛ 2. Моделі та методи оцінювання стабільності 8T-комірки RFSRAM при знижених напругах живлення 50
2.1. Проблематика забезпечення стабільності RFSRAM при знижених напругах живлення 50
2.2. Архітектура 8T-комірки двопортової RFSRAM 52
2.3. Методика вибору розмірів транзисторів 8T-комірки 54
2.4. Метод оцінювання SNM/WNM на основі butterfly-кривих 55
2.5. Метод оцінювання SNM на основі butterfly-кривих 61
2.6. SPICE-моделювання та визначення критичної напруги живлення 63
2.7. Визначення WNM за результатами моделювання 70
2.8 Оптимізація параметрів комірки пам'яті для компілятора регістрових файлів за 130-нм технологією 73
2.9. Експериментальний статистичний аналіз SNM-розподілу 8T dual-port SRAM-комірок 78
2.9.1 Передумови статистичного аналізу SNM 79

2.9.2 Методика експериментального дослідження 79
2.9.3 Розрахунок струмів і потужності споживання блока пам'яті 81
2.9.4 Експериментальні результати 84
2.9.5 Теоретичний розрахунок стандартного відхилення 88
РОЗДІЛ 3. Архітектура та алгоритмічне забезпечення компілятора двопортової RFSRAM-пам’яті з урахуванням post-layout SPICE-характеризації 92
3.1. Конфігурація та функціональні можливості компілятора RFSRAM 92
3.2. Архітектура макросу пам’яті, згенерованої компілятором RFSRAM 94
3.3. Формування топологічної структури макросу пам’яті 97
3.3.1 Збірка матриці 97
3.3.2. Збірка вихідних блоків даних порту читання 97
3.3.3. Збірка вхідних блоків даних порту запису 98
3.3.4. Збірка дешифратора рядків 98
3.3.5. Збірка адресних формувачів портів читання та запису 99
3.3.6. Підключення блоку керування портом читання 99
3.3.7. Підключення блоку керування портом запису 99
3.4. SPICE-модель макросу пам’яті та стан моделювання SRAM_MODEL 100
3.4.1 Параметри моделі 100
3.4.2 Контрольний список моделювання 102
3.4.3 Результати моделювання часу доступу 103
3.4.4. Розведення шин живлення головних блоків верхнього рівня 106
3.4.5 Узагальнення 109
3.5. Врахування паразитних RC-параметрів у post-layout характеризації 110
3.6. Оцінювання середнього динамічного струму та енергоспоживання згенерованих макросів RFSRAM 116
3.7. Програмна реалізація та генерація вихідних файлів компілятора… 121
3.7.1. Модульна структура програмної реалізації 122
3.7.2. Алгоритм формування матриці комірок пам’яті 124
3.7.3. Алгоритм збірки верхнього рівня макросу пам’яті 126
3.7.4. Генерація схемного LVS-нетліста 128
3.7.5. Генерація timing/power LIB-файлів на основі SPICE-характеризації 130
РОЗДІЛ 4. Експериментальна та післятопологічна верифікація макросів RFSRAM 135
4.1 Частотна характеризація регістрових файлів, виготовлених у кремнії, з урахуванням впливу тестового оточення. 135
4.2. Вбудовані засоби самотестування BIST у компіляторі регістрових файлів 144
4.2.1 Архітектура компілятора з підтримкою BIST 145

4.2.2. Інтеграція BIST у структуру пам’яті 146
4.2.3 Алгоритм тестування 147
4.2.4 Оцінка ефективності BIST 148
4.3. Post-layout SPICE-верифікація згенерованих макросів RFSRAM з урахуванням паразитних RC-параметрів 151
4.3.1. Мета та об’єкт post-layout верифікації 151
4.3.2. Формування netT, netR, netC та netRC 152
4.3.3. Порівняння T/R/C/RC-моделей 154
4.3.4 Аналіз впливу scaling та process-parameter variations 156
4.3.5. Порівняння Tacc без/з урахуванням паразитних параметрів у PVT-умовах 159
ВИСНОВКИ 162
Список використаних джерел 166
ДОДАТОК А. Фрагмент програмного коду Lua-модуля генерації timing/power LIB-файлів 174
ДОДАТОК Б. АКТ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЙНОГО ДОСЛІДЖЕННЯ 181

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ADC — Analog-to-Digital Converter, аналого-цифровий перетворювач.

ADRR — Address Read, адресна шина порту читання.

ADRW — Address Write, адресна шина порту запису.

ASIC — Application-Specific Integrated Circuit, інтегральна схема спеціального призначення.

BFAIL — BIST Fail, вихідний сигнал індикації помилки під час вбудованого самотестування.

BIST — Built-In Self-Test, вбудоване самотестування.

BSTOP — BIST Stop, вихідний сигнал завершення процедури вбудованого самотестування.

CDL — Circuit Description Language, текстовий схемний нетліст інтегральної схеми.

CLK — Clock, тактовий сигнал.

CLKR — Clock Read, тактовий сигнал порту читання.

CLKW — Clock Write, тактовий сигнал порту запису.

CMOS — Complementary Metal-Oxide-Semiconductor, комплементарна MOS-технологія.

CR — Cell Ratio, співвідношення провідностей pull-down та access транзисторів SRAM-комірки.

DIBL — Drain-Induced Barrier Lowering, зниження потенціального бар’єра, індуковане напругою стоку.

DIN — Data Input, шина вхідних даних.

DOUT — Data Output, шина вихідних даних.

DRM — Design Rules Manual, керівництво з правил проєктування для технологічного процесу.

DRV — Data Retention Voltage, мінімальна напруга збереження даних.

DSP — Digital Signal Processor, цифровий сигнальний процесор.

EDA — Electronic Design Automation, автоматизоване проєктування електронних систем та інтегральних схем.

FF — Fast-Fast, технологічний кут із швидкими n- та p-канальними MOS-транзисторами.

FS — Fast-Slow, технологічний кут із швидким n-канальним та повільним p-канальним MOS-транзистором.

GDSII — Graphic Data System II, формат топологічного опису інтегральної схеми.

GND — Ground, спільний провід / низький рівень напруги живлення

HBM — High Bandwidth Memory, пам’ять із високою пропускною здатністю.

IC — Integrated Circuit, інтегральна схема.

IP — Intellectual Property, функціональний блок інтегральної схеми.

IoT — Internet of Things, Інтернет речей.

LEF — Library Exchange Format, формат опису топологічних та технологічних обмежень бібліотечного блока.

LIB — Liberty library format, формат бібліотечного опису timing/power-характеристик.

LVS — Layout Versus Schematic, перевірка відповідності топології схемному опису.

MOS — Metal-Oxide-Semiconductor, структура метал–оксид–напівпровідник.
MOSFET — Metal-Oxide-Semiconductor Field-Effect Transistor, польовий транзистор зі структурою метал–оксид–напівпровідник.

MPW — Multi-Project Wafer, багатопроєктна кремнієва пластина.

NMOS — n-channel MOS transistor, n-канальний MOS-транзистор.

NoC — Network-on-Chip, мережа на кристалі.

PDK — Process Design Kit, набір технологічних даних та моделей для проєктування інтегральних схем.

PMOS — p-channel MOS transistor, p-канальний MOS-транзистор.

PR — Pull-up Ratio, співвідношення провідностей pull-up та access транзисторів SRAM-комірки.

PVT — Process, Voltage, Temperature, сукупність технологічних, електричних і температурних умов моделювання.

RC — Resistance-Capacitance, резистивно-ємнісні параметри електричного кола.
RCX — Resistance-Capacitance Extraction, екстракція паразитних резистивно-ємнісних параметрів.

RDF — Random Dopant Fluctuation, випадкові флуктуації домішок у каналі транзистора.

REN — Read Enable, сигнал дозволу читання.

RF — Register File, регістровий файл.

RFSRAM — Register File Static Random Access Memory, статична пам’ять регістрового типу.

RNM — Read Noise Margin, запас завадостійкості при читанні.

RSNM — Read Static Noise Margin, запас статичної стійкості при читанні.

SC — Single Clock, режим роботи з єдиним тактовим сигналом.

SF — Slow-Fast, технологічний кут із повільним n-канальним та швидким p-канальним MOS-транзистором.

SNM — Static Noise Margin, запас статичної завадостійкості комірки пам’яті.
SoC — System-on-Chip, система на кристалі.

SPICE — Simulation Program with Integrated Circuit Emphasis, програмний засіб схемотехнічного моделювання інтегральних схем.

SRAM — Static Random Access Memory, статична оперативна пам’ять.

SS — Slow-Slow, технологічний кут із повільними n- та p-канальними MOS-транзисторами.

STA — Static Timing Analysis, статичний аналіз часових параметрів цифрової схеми.

Tacc — Access Time, час доступу до даних пам’яті.

TT — Typical-Typical, типовий технологічний кут.

ULV — Ultra-Low Voltage, режим наднизької напруги живлення.

VDD — напруга живлення.

Vt — Threshold Voltage, порогова напруга MOS-транзистора.

W/L — співвідношення ширини та довжини каналу MOS-транзистора.

WEN — Write Enable, сигнал дозволу запису.

WFV — Work Function Variation, варіація роботи виходу матеріалу затвора.

WNM — Write Noise Margin, запас завадостійкості при записі.

АЦП — аналого-цифровий перетворювач.

ВАХ — вольт-амперна характеристика.

ІС — інтегральна схема.

РЕБ — радіоелектронна боротьба.

СК — система на кристалі.

ВСТУП

Актуальність теми.

Напівпровідникові технології є фундаментом сучасної цифрової трансформації, забезпечуючи функціонування ключових галузей економіки та безпеки, зокрема телекомунікацій, радіоелектроніки, штучного інтелекту, Інтернету речей, автономних систем та енергетики. Стрімке зростання глобального ринку напівпровідників, який вже перевищив сотні мільярдів доларів і продовжує динамічно розвиватися, зумовлене, передусім, експоненційним попитом на обчислювальні та комунікаційні рішення нового покоління.

Особливе значення напівпровідники мають для телекомунікаційних систем, які становлять основу сучасної інформаційної інфраструктури. Системи мобільного зв'язку 5G/6G, супутникові комунікації, радіолокаційні комплекси, системи радіоелектронної боротьби (РЕБ), безпілотні платформи та мережеві обчислювальні вузли базуються на високопродуктивних інтегральних мікросхемах, здатних обробляти великі обсяги даних у режимі реального часу. У таких системах критично важливими є не лише обчислювальні блоки, але й підсистеми пам'яті, що забезпечують швидкий доступ до даних та визначають ефективність виконання алгоритмів обробки сигналів.

З точки зору телекомунікаційних і радіотехнічних застосувань такі інтегральні схеми працюють у складі радіоелектронного обладнання, що приймає, формує та обробляє сигнали, пов'язані з поширенням електромагнітних хвиль, тому характеристики вбудованої пам'яті впливають на частоту обробки, смугу пропускання, надійність та стабільність роботи системи в умовах експлуатації.

Сучасні інтегральні мікросхеми класу **System-on-Chip (SoC)** характеризуються високим ступенем інтеграції, при якому значна частина площі кристала припадає на вбудовану пам’ять. Зокрема, пам’ять регістрового типу **RFSRAM (Register File Static Random Access Memory**) використовується як банки операндів, буфери даних, кеш-пам’ять першого рівня та внутрішні структури зберігання у цифрових сигнальних процесорах, мережевих процесорах, модемах, радарних системах та засобах РЕБ. Частка таких блоків у площі кристала може перевищувати 50 %, а їхній внесок у загальне енергоспоживання є визначальним, особливо у системах із обмеженим енергоресурсом.

Паралельно з цим, світова напівпровідникова індустрія перебуває у стані глибокої трансформації, пов’язаної з геополітичними факторами, зміною ланцюгів постачання та переходом до нових архітектур (chiplet, 3D-інтеграція, Beyond Moore). Це створює нові виклики, але водночас відкриває можливості для країн, які здатні сформувати власну науково-технологічну базу.

Для України питання розвитку напівпровідникових технологій має стратегічне значення. В умовах воєнного стану та необхідності зміцнення обороноздатності держави особливої ваги набуває створення власних високотехнологічних рішень у сфері мікроелектроніки. Формування національної екосистеми проєктування інтегральних схем дозволяє забезпечити технологічний суверенітет, зменшити залежність від зовнішніх постачальників та створити основу для розвитку сучасних систем зв’язку, радіолокації, РЕБ та інших критично важливих напрямів.

Україна має значний потенціал для інтеграції у глобальні ланцюги створення напівпровідникової продукції завдяки високому рівню інженерної освіти, наявності кваліфікованих кадрів та наукових шкіл. Водночас перспективними є нішеві напрямки, зокрема розробка енергоефективних спеціалізованих

інтегральних схем (**ICs**), радіаційно-стійких компонентів, телекомунікаційної електроніки та пристроїв Інтернету речей (**IoT**).

У цьому контексті розробка двопортових регістрових файлів із підвищеною енергоефективністю та гарантованою стабільністю є важливою складовою створення сучасних телекомунікаційних систем. Особливо актуальним є створення методології, яка дозволяє враховувати фізичні, статистичні та технологічні обмеження на ранніх етапах проєктування та інтегрувати ці підходи у автоматизовані інструменти.

Отже, актуальність роботи зумовлена необхідністю створення методів проєктування енергоефективних двопортових макросів RFSRAM, здатних працювати при знижених напругах живлення із забезпеченням стабільності збереження даних, прийнятного часу доступу та автоматизованої генерації у складі memory compiler. Практичне застосування таких рішень є важливим для SoC телекомунікаційного обладнання, IoT-пристроїв, edge-computing вузлів, радарних і автономних систем, де потрібні низьке енергоспоживання та гарантовані часові характеристики.

Зв'язок роботи з науковими програмами, планами, темами.

Дисертаційна робота виконана в межах пріоритетних напрямів розвитку науки і техніки України відповідно до Закону України «Про пріоритетні напрями розвитку науки і техніки» №2623-III, де серед ключових визначено розвиток інформаційно-комунікаційних технологій, електроніки та мікро- і наноелектроніки. Тематика роботи також узгоджується із Законом України «Про наукову і науково-технічну діяльність» №848-VIII та спрямована на вирішення задач створення енергоефективних компонентів інтегральних схем.

Тематика роботи узгоджується з напрямами формування державної політики у сфері стратегічних технологій, зокрема Стратегією цифрового розвитку інновацій України WINWIN до 2030 року, що здійснюється за координації

Міністерства цифрової трансформації України, а також іншими ініціативами щодо розвитку напівпровідникової індустрії України та інтеграції до європейських механізмів EU Chips Act.

На міжнародному рівні тематика дисертації відповідає положенням Регламенту (ЄС) 2023/1781 Європейського Парламенту та Ради («EU Chips Act»), який визначає розвиток проєктування інтегральних схем, створення бібліотек IP-блоків та забезпечення енергоефективності як ключові складові технологічного суверенітету. Результати роботи корелюють із напрямами діяльності Chips Joint Undertaking та Європейської ради з напівпровідників (European Semiconductor Board), зокрема у частині розвитку дизайн-екосистеми та підвищення надійності мікроелектронних компонентів.

Автор дисертаційної роботи брав участь у роботі експертної групи при Міністерстві цифрової трансформації України, що здійснює підготовку пропозицій щодо інтеграції України до ініціатив EU Chips Act та формування національної напівпровідникової стратегії. У межах цієї діяльності автором обґрунтовано доцільність розвитку напрямів проєктування інтегральних схем, зокрема створення енергоефективних блоків пам’яті та відповідних інструментів автоматизованого проєктування.

Тематика дисертації має безпосередню прикладну спрямованість і відповідає задачам створення сучасних мікроелектронних компонентів для телекомунікаційних систем, засобів радіоелектронної боротьби, радарних комплексів та безпілотних платформ, де критичними є вимоги до енергоспоживання, надійності та автономності.

Мета і задачі дослідження.

Мета дисертаційної роботи полягає у розробленні та верифікації методів автоматизованого проєктування двопортової пам’яті регістрового типу RFSRAM

на основі 8T SRAM-комірки з пріоритетом низького енергоспоживання, забезпеченням статичної стійкості при знижених напругах живлення та урахуванням паразитних параметрів топології під час застосування післятопологічного (post-layout) SPICE-моделювання (**SPICE — Simulation Program with Integrated Circuit Emphasis**).

У дисертаційній роботі використовуються методи та засоби схемотехнічного моделювання, топологічного моделювання, аналізу та синтезу інтегральних мікросхем. Застосовуються методи та засоби дискретної математики, теорії Булевих операцій, теорії ймовірності, а також комплементарної MOS-схемотехніки CMOS (Complementary Metal-Oxide-Semiconductor) та комп'ютерного програмування. Розрахункова частина роботи базується на використанні рівнянь, що описують динамічну потужність, середній динамічний струм, розкид порогової напруги, запас статичної стійкості та зміну часу доступу з урахуванням паразитних параметрів.

Для досягнення поставленої мети у роботі поєднано схемотехнічне моделювання, топологічну реалізацію, статистичний аналіз стабільності комірки пам'яті, post-layout SPICE-моделювання та експериментальну перевірку тестових зразків у кремнії з метою отримання даних про технічні характеристики та відсоток виходу придатних виробів (yield).

Об'єктом дослідження є процеси схемотехнічного проєктування, топологічної реалізації, автоматизованої генерації та post-layout верифікації двопортової пам'яті регістрового типу RFSRAM.

Для досягнення поставленої мети необхідно було вирішити наступні науково-технічні задачі:

1. Проаналізувати сучасні архітектури SRAM/RFSRAM пам'яті та методи зниження енергоспоживання вбудованих блоків пам'яті, визначити обмеження класичних 6T SRAM-комірок у режимах знижених напруг живлення та

обґрунтувати доцільність використання 8T-комірки з розділеними портами читання і запису.

2. Розробити та верифікувати підхід, орієнтований на аналіз запасу статичної завадостійкості **SNM (Static Noise Margin)**, до вибору розмірів транзисторів 8T-комірки регістрового файлу на основі аналізу **butterfly curves**, критичної напруги живлення та статистичних варіацій технологічних параметрів.

3. Дослідити вплив розкиду порогових напруг MOS-транзисторів (Metal-Oxide-Semiconductor) на запас статичної завадостійкості пам’яті та обґрунтувати використання статистичного підходу для оцінювання стабільності 8T dual-port SRAM/RFSRAM комірок.

4. Створити параметризовану структуру та програмне забезпечення двопортового компілятора пам’яті **RFSRAM 1R1W**, що забезпечує автоматизовану генерацію макросів регістрової пам’яті з пріоритетом низького енергоспоживання.

5. Розробити методику урахування паразитних резистивно-ємнісних параметрів **RC (Resistance-Capacitance)** топології згенерованих реалізацій пам’яті на основі rule-based екстракції та формування SPICE-compatible netlist для post-layout (післятопологічного) моделювання.

6. Розробити маршрут post-layout SPICE-характеризації реалізацій RFSRAM для оцінювання часу доступу, затримок, динамічного струму та працездатності в різних технологічних, електричних і температурних умовах **PVT (Process, Voltage, Temperature)**.

7. Провести SPICE-моделювання, post-layout simulation та експериментальну перевірку тестових зразків пам’яті, виготовлених у кремнії, з метою підтвердження працездатності запропонованих рішень, оцінювання часу доступу, струму споживання та стабільності пам’яті.

8. Оцінити практичну ефективність запропонованих методів для проєктування енергоефективних мікросхем пам’яті, придатних для використання у телекомунікаційних, радіотехнічних, сенсорних, автономних та оборонних системах.

Предмет дослідження − методи, моделі та програмні засоби автоматизованого проєктування двопортової RFSRAM-пам’яті на основі 8T SRAM-комірки, орієнтовані на зниження енергоспоживання, забезпечення запасу статичної стійкості SNM, урахування статистичних варіацій параметрів MOS-транзисторів та паразитних RC-параметрів топології при post-layout SPICE-моделюванні.

Методи дослідження.

Для вирішення поставлених в дисертації задач використовувалися наступні методи аналізу та дослідження:

- ***Аналітичний (теоретичний)*** – з метою узагальнення останніх наукових розробок та досягнень провідних закордонних вчених у сфері автоматизованого проєктування та фізичної верифікації топології мікросхем пам’яті типу SRAM (Static Random Access Memory) та регістрового файлу (розділ 1);
- ***Аналізу та синтезу (комплексний)*** – для уточнення і конкретизації понятійно-категоріального апарату у сфері автоматизованого проєктування та фізичної верифікації топології інтегральних мікросхем (розділ 1);
- ***Емпіричний*** - вимірювання та експеримент застосовуються на етапі збору необхідних даних щодо вимірювання вольт-амперних характеристик та проходження функціонального контролю зразків мікросхем, виготовлених на фабриці (Розділ 2).

- ***Систематизації (теоретичний)*** – для розкриття основних принципів проєктування інтегральних мікросхем тенденцій розвитку новітніх технологій у даній галузі на світових ринках.
- ***Абстрактно-логічний (комплексний)*****,** що дозволив запропонувати новітні методи/підходи до зниження енергоспоживання мікросхем пам'яті регістрового типу. (Розділ 3);

- ***Комп'ютерне моделювання (спеціальний) та симуляція*****:** використання засобів автоматизованого проєктування (CAD) для симуляції та моделювання поведінки комірок і схем SRAM у різних робочих умовах. Це дозволяє створювати віртуальні прототипи та оптимізувати перед фізичним виготовленням (розділ 3, 4).

  Зокрема:
- ***Характеризація*** – моделювання характеристик тестових структур SRAM із використанням інформації щодо процесу виробництва напівпровідників, наприклад CMOS, і визначення їхніх електричних властивостей за допомогою електричних випробувань і методів визначення характеристик, таких як вимірювання ВАХ, аналіз перехідних процесів і вимірювання частотної характеристики.
- ***Графічний метод (спеціальний)*** - аналіз SNM (Static Noise Margin) по так званих кривих butterfly, що є критично важливим аспектом конструкції та характеристики SRAM. Він визначає стійкість комірки SRAM до співвідношення шуму та напруги корисного сигналу під час операцій читання та запису.

Вищезазначені спеціальні наукові методи постійно вдосконалюються та адаптуються дослідниками та інженерами, щоб розширити межі технології

SRAM, що призводить до покращення щільності, швидкості, енергоефективності та загальної продуктивності.

Наукова новизна одержаних результатів.

До основних нових наукових результатів, які одержані в дисертаційній роботі, можна віднести наступні:

1. **Вперше запропоновано** SNM-орієнтований підхід до проєктування 8T-комірки двопортової пам'яті RFSRAM для режимів зниженої напруги живлення, який враховує критичну мінімальну напругу живлення, розкид порогових напруг MOS-транзисторів та умови стійкості, визначені за butterfly curves.
2. **Вперше для досліджуваного класу 8T dual-port SRAM/RFSRAM-комірок експериментально обґрунтовано** застосування статистичної моделі розподілу SNM (Static Noise Margin), пов'язаної з Gaussian-distributed threshold voltage variations MOS-транзисторів, та показано узгодження експериментальних даних із розрахунками на основі A_{vto} - area mismatch coefficient.
3. **Удосконалено** метод оцінювання мінімальної робочої напруги живлення RFSRAM, який, на відміну від використання лише номінального значення SNM, враховує статистичний розкид параметрів транзисторів і дозволяє прогнозувати стабільність пам'яті при зниженні V_{DD}.
4. **Розроблено** параметризовану архітектуру двопортового RFSRAM memory compiler, у якій 8T-комірка, схеми керування, адресні формувачі, дешифратори рядків, вхідні та вихідні буфери інтегровані в автоматизований маршрут генерації реалізацій пам'яті з пріоритетом низького енергоспоживання.
5. **Набули подальшого розвитку** методи post-layout верифікації реалізацій RFSRAM шляхом інтеграції rule-based екстракції паразитних RC-параметрів, формування компактного RC-netlist та використання post-layout SPICE simulation для уточнення часу доступу, затримок і струму споживання. Запропонований

підхід дозволяє формувати simulation-ready RC-нетлісти для великих схем пам’яті безпосередньо придатні для post-layout SPICE-моделювання.

Практичне значення одержаних результатів.

Практичне значення одержаних в дисертації результатів полягає в наступному:

1. Запропоновані методи можуть бути використані для інженерного проєктування енергоефективних макросів двопортової пам’яті RFSRAM для CMOS-технологій 180 нм, масштабованої 152 нм, 130 нм та інших mature / advanced planar process nodes.
2. Розроблений підхід до проєктування 8T-комірки дозволяє знижувати напругу живлення пам’яті без втрати стабільності збереження даних, що підтверджено моделюванням та тестуванням зразків пам’яті у кремнії.
3. Результати статистичного аналізу SNM можуть застосовуватися для вибору допустимих значень V_{DD}, розмірів транзисторів та технологічних допусків при проєктуванні low-power RFSRAM.
4. Запропонована структура RFSRAM memory compiler скорочує час розробки окремих екземплярів пам’яті та забезпечує автоматизовану генерацію макросів з різною глибиною, розрядністю та режимами роботи.
5. Розроблений маршрут post-layout SPICE-характеризації з урахуванням паразитних RC-параметрів дозволяє підвищити точність оцінювання access time, затримок і динамічного струму до виготовлення мікросхеми. На прикладах досліджених макросів показано, що неврахування паразитних параметрів топології може призводити до суттєвої похибки оцінювання часу доступу, максимальної швидкодії та динамічного струму; для окремих досліджених режимів така похибка перевищує 30 %.

Отримані результати можуть бути застосовані при створенні та експлуатації радіоелектронного обладнання і систем на кристалі для телекомунікаційного обладнання, цифрової обробки сигналів, радарних комплексів, систем радіоелектронної боротьби, безпілотних платформ та IoT-пристроїв, де критичними є низьке енергоспоживання, надійність, час доступу, частотні обмеження та стабільна робота при зниженій напрузі живлення.

У промисловому контексті результати роботи можуть бути використані як основа для створення бібліотек пам’яті та IP-блоків для напівпровідникових фабрик і fabless-компаній.

Особистий внесок здобувача.

Основні положення та результати дисертаційної роботи отримані автором самостійно. Автором виконано аналіз сучасних архітектур SRAM/RFSRAM, розроблено методику SNM-орієнтованого вибору параметрів 8T-комірки, проведено SPICE-моделювання та статистичний аналіз стабільності комірок пам’яті, розроблено архітектуру двопортового компілятора RFSRAM, виконано аналіз post-layout SPICE-характеристик з урахуванням паразитних RC-параметрів та проведено інтерпретацію результатів експериментального тестування зразків у кремнії. У працях, опублікованих у співавторстві, автору належать постановка задачі, розроблення методики дослідження, виконання моделювання, аналіз отриманих результатів та підготовка основних висновків.

Апробація результатів дисертації.

Наукові й практичні результати дисертаційної роботи доповідалися та обговорювалися на наступних міжнародних і всеукраїнських науково-

практичних та науково-технічних конференціях: − XXIV Міжнародна науково-практична конференція здобувачів вищої освіти і молодих учених "ПОЛІТ. Сучасні проблеми науки" (2–5 квітня 2024 року). Тема виступу: «Built-In Self Test For Register File Memory Compiler». - Інтегровані Інтелектуальні Робототехнічні Комплекси (ІІРТК-2024) Сімнадцята Міжнародна Науково-Практична Конференція 21-22 травня 2024 р. Київ, Україна. Тема виступу: “Power Consumption Calculation Of Register File Memory”. - XVII Міжнародна науково-технічна конференція «ABIA-2025», Державний університет «Київський авіаційний інститут», 22-24 квітня 2025 року. Тема виступу: “Minimizing leakage current in 64K SRAM using High-Threshold Voltage transistors”. -- *2026 IEEE 43rd International Conference on Electronics and Nanotechnology (ELNANO-2026)*, Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine. April 27-30, 2026. Тема виступу: “Impact of Hardware Test Equipment on Frequency Characterization of 130-nm Register File”.

Публікації.

Основні результати дисертації опубліковано у 8 наукових працях: 4 статтях у наукових журналах, 1 статті в матеріалах міжнародної конференції IEEE ELNANO, проіндексованих у Scopus, та 3 публікаціях матеріалів доповідей на наукових конференціях. Чотири праці опубліковано одноосібно.

Структура та об'єм дисертації.

Робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел та додатків.

У першому розділі виконано аналіз проблеми проєктування енергоефективної RFSRAM-пам’яті, розглянуто методи зниження енергоспоживання SRAM/RFSRAM, вплив технологічного масштабування та статистичних варіацій

на стабільність комірки пам’яті, а також обґрунтовано необхідність урахування паразитних RC-параметрів у post-layout моделюванні.

У другому розділі розроблено моделі та методи оцінювання стабільності 8T-комірки RFSRAM при знижених напругах живлення, включаючи аналіз SNM та запасу завадостійкості запису **WNM (Write Noise Margin)**, статистичних варіацій порогових напруг та експериментальний аналіз SNM-розподілу.

У третьому розділі розглянуто архітектуру та алгоритмічне забезпечення двопортового компілятора RFSRAM, формування макросу пам’яті, SPICE-моделі, post-layout characterization та генерацію вихідних файлів компілятора.

У четвертому розділі наведено результати експериментальної та післятопологічної верифікації макросів RFSRAM, включаючи частотну характеризацію, BIST-тестування та post-layout SPICE-аналіз із урахуванням паразитних RC-параметрів.

У висновках повідомляються основні результати проведених досліджень.

РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМИ ПРОЄКТУВАННЯ ЕНЕРГОЕФЕКТИВНОЇ RFSRAM-ПАМ’ЯТІ.

1.1. Роль вбудованої пам’яті та регістрових файлів у сучасних SoC

Більшість сучасних інтегральних мікросхем (IC) являють собою системи на кристалі (СК), що включають такі стандартні елементи, як процесори, пам’ять, перетворювачі, логіка. Зокрема, СК містять готові топологічно розведені IP блоки, які представляють різні види пам'яті (SRAM, RFSRAM, FLASH, OTP…) вхідні/вихідні та різні аналогові блоки.

Сучасні телекомунікаційні прилади містять кілька рівнів організації пам’яті безпосередньо на кристалі своїх мікроконтролерів, а також додаткові зовнішні

рівні, включаючи оперативну пам’ять, пам’ять для зберігання даних користувача, пам’ять одноразового чи багаторазового запису/стирання для зберігання налаштувань режимів роботи обладнання – «прошивки».

У практиці проєктування SoC кількість екземплярів пам’яті може сягати сотень реалізацій пам'яті різного об’єму, форми (aspect ratio) та довжини слова. Досягти такого результату можна тільки за допомогою компіляторів пам'яті [1, 2]. Наприклад, у комерційному варіанті компілятора двопортового ОЗП кількість можливих варіантів генераторів перевищує 600 000.

Упродовж останніх років найбільші технологічні компанії посилили розробку власних спеціалізованих мікросхем, зокрема блоків пам’яті та Semiconductor IP. Це підвищує значення автоматизованих засобів проєктування, здатних швидко генерувати енергоефективні макроси пам’яті для різних технологічних процесів і конфігурацій.

Поряд із цим, світовий ринок Semiconductor IP перевищив $2.5 млрд у 2021 р., а загальна кількість проданих мікрочипів перевищила 1.2 трлн шт. Більшу частину від усіх проданих ліцензій займають IP блоки пам’яті різного типу.

Розробка та подальша фабрикація (виготовлення у кремнії) чипів пам’яті є ключовим етапом становлення кожної нової технології на напівпровідниковій фабриці, що передує масовому старту прийому замовлень на виготовлення чипів. Найбільш технологічною, швидкісною та коштовною є пам’ять регістрового типу RFSRAM, що є кеш-буфером першого рівня для зберігання інструкцій та даних процесору, яка розміщується безпосередньо на одному кристалі із ним.

Для телекомунікаційних і радіотехнічних застосувань характерною є необхідність одночасного забезпечення високої продуктивності, надійності та енергоефективності. Це особливо актуально для оборонних систем, де

енергетична автономність, стабільність роботи в екстремальних умовах і гарантована працездатність є критичними параметрами. У таких системах пам’ять працює в умовах підвищених температурних, електромагнітних та радіаційних впливів, що додатково ускладнює її проєктування.

Одним із ключових напрямів підвищення енергоефективності є зниження напруги живлення інтегральних мікросхем. Відомо, що динамічна потужність визначається залежністю:

$$P_{dyn} = \alpha C V_{DD}^2 f \tag{1.1}$$

що означає квадратичну залежність потужності від напруги живлення. Таким чином, зменшення V_{DD} є найбільш ефективним способом зниження енергоспоживання, особливо для систем, що працюють у режимах постійної обробки сигналів. Однак зниження напруги живлення супроводжується фундаментальними фізичними обмеженнями, серед яких:

- деградація запасу статичної стійкості (Static Noise Margin, SNM);
- зростання впливу статистичних варіацій параметрів транзисторів;
- ускладнення процесів запису та читання;
- зниження максимальної робочої частоти;
- підвищена чутливість до паразитних ємностей та опорів.

Ці ефекти є особливо критичними для двопортових регістрових файлів, де одночасна робота портів читання та запису створює додаткові обмеження на стабільність комірки пам’яті [3-10].

Незважаючи на значний розвиток засобів автоматизованого проєктування електронних систем **EDA (Electronic Design Automation)**, існуючі memory compilers орієнтовані переважно на мінімізацію площі або досягнення максимальної швидкодії. Вони не забезпечують формалізованого підходу до

проєктування пам’яті з урахуванням статистичних варіацій параметрів та не оптимізовані для роботи в ultra-low voltage режимах. Це обмежує їх застосування у критичних телекомунікаційних і оборонних системах, де необхідна гарантована працездатність у широкому діапазоні умов експлуатації.

При проєктуванні пам’яті традиційно розглядаються три основні критерії оптимізації: площа (Area), енергоспоживання (Power) та швидкодія (Speed). Проте, як правило, одночасно можуть бути оптимізовані лише два з трьох параметрів, тоді як третій виступає обмеженням.

Зменшення середнього динамічного струму в пам’яті типу RFSRAM є критично важливим для підвищення енергоефективності, збереження цілісності даних, забезпечення надійності та стабільності роботи системи в різних режимах експлуатації [11-13].

З метою скорочення часу проєктування кожного окремого екземпляра пам’яті використовується спеціалізований інструмент — компілятор пам’яті.

1.2. Компілятори пам’яті як засіб автоматизованого проєктування RFSRAM

Компілятор пам'яті (memory compiler) – це комплекс спеціалізованого програмного забезпечення, що складається із основної керуючої програми, бібліотеки базових елементів (leaf cell library) та безпосередньо допоміжного рівня – компілятора з підтримкою додаткових вихідних форматів даних [1, 2].

Компілятор RFSRAM є спеціалізованим програмним засобом автоматизованого проєктування (EDA), який включає бібліотеки елементів, скрипти та службові файли/каталоги, що містять необхідну інформацію для генерації екземплярів пам’яті. У складі компілятора постачаються шаблонні конфігураційні файли для формування як front-end, так і back-end представлень. Також передбачені файли,

доступні користувачу для налаштування параметрів, і файли, недоступні користувачу, що містять пропрієтарні дані фабрики-виробника.

Зазвичай компілятори пам’яті оптимізуються за трьома ключовими параметрами: площа, швидкодія та енергоспоживання [1, 2, 12]. Саме ці критерії лежать в основі підходів більшості розробників при створенні RFSRAM та SRAM компіляторів.

Запропонований у роботі компілятор регістрового файлу RFSRAM є двопортовою пам’яттю (1 порт читання, 1 порт запису), реалізованою за технологією 0.18 мкм. Це синхронний компілятор пам’яті, оптимізований для досягнення високої швидкодії та низького енергоспоживання. Він здатний працювати при стандартній напрузі живлення 1.8 В, а також підтримує режими знижених напруг до 0.8 В із відповідним зниженням продуктивності.

Системний підхід у дослідженні: об’єкт дослідження розглядається як складна багатокомпонентна динамічна система із набором характеристик, зміна яких залежить від часу.

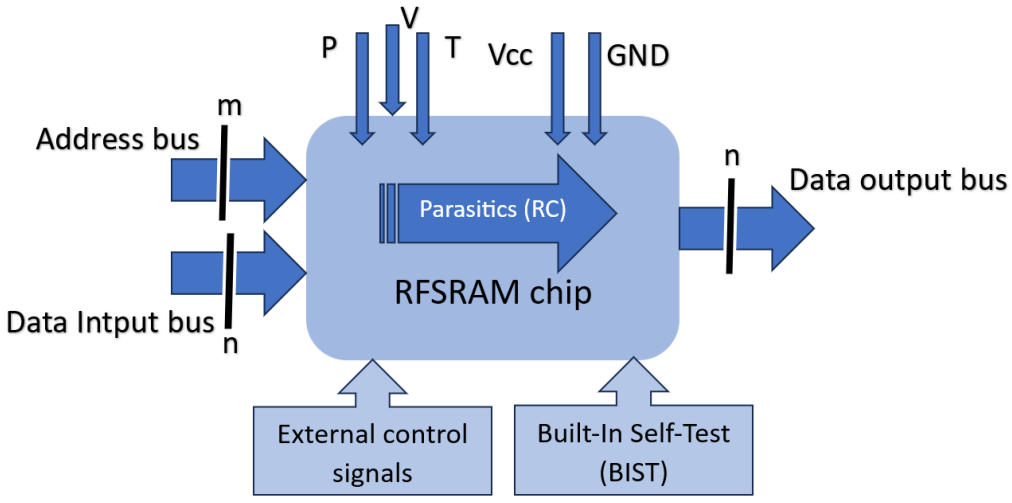


Рис. 1.1 – Базова блок-схема побудови системи об’єкта дослідження RFSRAM

Особливістю є те, що фактор випадковості у даній галузі має дуже велике значення при моделюванні характеристик пам'яті при визначені параметрів надійності комірки. Наприклад: вплив розкиду значень порогової напруги на стабільність збереження інформації при критичних значеннях напруги живлення підтверджується результатами моделювання у режимі читання [4, 5, 10].

Крім того, у даній галузі комп'ютерні / інформаційні технології є головним інструментом пізнання, розробки, моделювання і верифікації. Вони дають змогу побудувати та застосовувати спеціальні моделі пам'яті для визначення її статичних та динамічних характеристик.

Ключовими аспектами є: порогова напруга, критична напруга, SNM (Static Noise Margin) – мінімальне значення рівня шуму, що викликає зміну значення або стану інформації у комірці пам'яті [3].

З точки зору вартості/витрат технологічної реалізації: (1) пріоритет на максимальну швидкодію та продуктивність (Performance), (2) Пріоритет на мінімальну площу топології (Area), (3) Пріоритет на мінімальне енергоспоживання (Low-Power).

Для даної роботи як головний критерій оптимізації обрано енергоспоживання, тоді як площа та швидкодія розглядаються як обмеження.

1.3. Методи зниження енергоспоживання SRAM/RFSRAM

Зниження енергоспоживання вбудованих блоків пам'яті є однією з ключових задач при проєктуванні сучасних інтегральних мікросхем класу SoC, особливо для телекомунікаційних, мобільних, сенсорних, автономних та оборонних систем. SRAM та регістрові файли займають значну частину площі кристала і формують суттєву частку як динамічного, так і статичного енергоспоживання. У низці робіт підкреслюється, що масштабування технологій та зменшення напруги

живлення дозволяють зменшити площу й енергоспоживання, але одночасно погіршують стабільність комірок пам'яті, зокрема у режимах читання та зберігання [4, 6, 10, 14–16]. Загальне енергоспоживання пам'яті можна подати як суму динамічної та статичної складових:

$$P_{total} = P_{dyn} + P_{leak} \tag{1.2}$$

де P_{dyn}— динамічна потужність, пов'язана з перемиканням вузлів схеми, а P_{leak}— статична потужність, обумовлена струмами витоку.

Динамічна складова визначається співвідношенням:

$$P_{dyn} = \alpha C_{load} V_{DD}^2 f \tag{1.3}$$

де α— коефіцієнт активності перемикань, C_{load}— ефективна ємність навантаження, V_{DD}— напруга живлення, f— робоча частота.

З цієї залежності випливає, що зменшення напруги живлення є найбільш ефективним способом зниження динамічної потужності, оскільки потужність зменшується пропорційно квадрату V_{DD}. Саме тому voltage scaling є базовим методом побудови ultra-low-power SRAM та RFSRAM. Водночас у роботах з аналізу SRAM підкреслюється, що зменшення V_{DD} погіршує Static Noise Margin (SNM), Read Noise Margin (RNM/RSNM), Write Margin (WM) та Data Retention Voltage (DRV), а при надмірному зниженні напруги може призвести до втрати даних [3, 6, 7, 10, 17].

1.3.1. Зниження напруги живлення як базовий метод енергооптимізації

Найбільш прямим способом зниження енергоспоживання пам'яті є зменшення напруги живлення. Для SRAM цей метод особливо привабливий, оскільки значна частина енергії витрачається на заряд і розряд бітових ліній, word-line та внутрішніх вузлів комірки. У near-threshold та sub-threshold режимах

теоретично можна досягти суттєвого зменшення енергоспоживання, що є важливим для автономних пристроїв, сенсорних мереж, біомедичних систем та інших застосувань із жорсткими обмеженнями по потужності [6, 10, 14, 15].

Однак саме цей метод має найбільш критичні обмеження. При зниженні V_{DD} зменшується різниця між рівнями логічного «0» і «1», знижується запас статичної стійкості, а вплив варіацій порогових напруг транзисторів стає визначальним. У випадку класичної 6T SRAM комірки одночасне покращення read stability та write-ability є складною задачею, оскільки вимоги до сили транзисторів у режимі читання і запису є конфліктними. Це обмежує можливість застосування традиційної 6T SRAM у глибоких режимах наднизької напруги живлення ULV (Ultra-Low Voltage). Далі — ULV-режими [3, 7, 17].

Для проєктування чипів пам'яті просте зменшення напруги живлення не може розглядатися як самодостатній метод. Воно має бути поєднане з аналізом SNM/WNM, статистичних варіацій транзисторів та архітектурних особливостей комірки пам’яті [3–10, 17].

1.3.2. Оптимізація розмірів транзисторів і співвідношень CR/PR

Іншим поширеним методом покращення енергоефективності та стабільності SRAM є оптимізація розмірів транзисторів комірки. У класичній SRAM комірці важливу роль відіграють співвідношення Cell Ratio (CR) та Pull-up Ratio (PR), які визначають баланс між транзисторами доступу, pull-down та pull-up транзисторами. У роботах з аналізу 6T SRAM показано, що саме CR, PR та V_{DD} є ключовими параметрами для оцінки SNM, Read Margin, Write Margin і DRV [3, 17–19].

Загально цей підхід можна записати через співвідношення провідностей:

$$CR = \frac{(W/L)_{pull-down}}{(W/L)_{access}} \quad (1.4)$$

$$PR = \frac{(W/L)_{pull-up}}{(W/L)_{access}} \quad (1.5)$$

Збільшення CR покращує стійкість при читанні, оскільки pull-down транзистор сильніше утримує внутрішній вузол комірки. Проте надмірне збільшення pull-down транзистора погіршує запис, збільшує площу та паразитні ємності. Аналогічно, зміна PR впливає на здатність комірки до запису, але може погіршувати режим зберігання або збільшувати енергоспоживання.

У роботі, де порівнюються різні SRAM-топології прямо зазначається, що покращення SNM може бути досягнуте за рахунок правильного sizing транзисторів, але при цьому виникає компроміс між стабільністю, затримкою, площею та потужністю [18, 19].

Для даної дисертації цей підхід є принципово важливим, оскільки він безпосередньо пов'язаний з ідеєю SNM/WNM-орієнтованого вибору розмірів транзисторів у компіляторі пам'яті. На відміну від емпіричного підбору W/L, у роботі вибір геометричних параметрів здійснюється з урахуванням статистичного розкиду порогових напруг до формалізованого sizing з урахуванням статистичних розкидів V_t.

1.3.3. Перехід від 6T до 8T, 9T, 10T та інших SRAM-комірок

Класична 6T SRAM комірка є найбільш компактною, однак її головний недолік полягає в конфлікті між read stability і write-ability [3, 7, 17, 19]. При зниженні напруги живлення цей конфлікт стає особливо гострим. Тому значна кількість робіт присвячена розробці альтернативних SRAM-комірок із більшою кількістю транзисторів: 7T, 8T, 9T, 10T, 11T, 12T [20-26].

У роботі з порівняння 6T і 9T SRAM зазначено, що в sub-threshold режимі класична 6T SRAM має слабку здатність до читання і запису [23], а також зменшений SNM через масштабування, зниження V_{DD} та флуктуації порогових напруг. Запропонована 9T-комірка покращує read, hold і write SNM за рахунок додаткових транзисторів, які ізолюють вузли зберігання від бітових ліній.

Подібний висновок наведено у порівняльних роботах по 6T, 7T, 8T, 9T і 10T SRAM: 10T-комірки можуть демонструвати вищий RSNM і менше повне енергоспоживання, але це досягається ціною збільшення площі та ускладнення схеми [18, 20, 22, 24-26].

Загальний компроміс для таких комірок можна подати так:

$$Stability \uparrow \Rightarrow Area \uparrow,\ C_{parasitic} \uparrow$$
$$C_{parasitic} \uparrow \Rightarrow P_{dyn} \uparrow,\ t_{delay} \uparrow$$

Збільшення кількості транзисторів не завжди є оптимальним рішенням при розробці компілятора пам’яті. Для RFSRAM особливо у вигляді параметричного компілятора, важливо знайти баланс між стабільністю, площею, частотою (швидкодією) та енергоспоживанням.

1.3.4. Read-decoupled архітектури та 8T-комірки

Одним із найбільш поширених підходів до підвищення стабільності SRAM при низькій напрузі є відокремлення порту читання від вузлів зберігання [20-22]. У read-decoupled 8T-комірках для читання використовується окремий read-port, завдяки чому внутрішній вузол комірки не піддається прямому впливу бітової лінії під час читання. Це дозволяє значно покращити RSNM порівняно з 6T SRAM.

Такий підхід особливо важливий для регістрових файлів, де природною є двопортова архітектура з окремими портами читання та запису. Саме тому 8T-комірка є доцільною базою для RFSRAM із пріоритетом низького енергоспоживання [20–22, 27, 28].

Однак read-decoupled архітектури також мають недоліки. У bit-interleaved масивах можуть виникати так звані “half-select”-проблеми, коли частина неактивних комірок у вибраному рядку піддається небажаному впливу під час операції запису або читання. Для вирішення цієї проблеми в літературі застосовуються write-back, cross-point комірки, додаткові word-line, а також write-assist схеми. Проте ці рішення часто збільшують площу, ускладнюють timing control або створюють додаткові витрати потужності.

У роботі Huang та Chiou запропоновано single bitline 8T SRAM із асинхронним dual word-line control. Такий підхід спрямований на одночасне покращення read stability, half-select stability та write-ability без додаткових периферійних схем. Тестовий 4 кБ макрос у 90 нм CMOS продемонстрував роботу при напрузі 0.36 В і споживанні 2.68 мкВт [21].

1.3.5. Зменшення амплітуди перемикання бітових ліній і оптимізація precharge

Значна частина динамічної енергії SRAM витрачається на заряд і розряд бітових ліній [7, 11, 13]. Енергія одного перемикання бітової лінії може бути записана як:

$$E_{BL} = C_{BL} V_{swing}^2 \tag{1.6}$$

де C_{BL}— ємність бітової лінії, V_{swing}— амплітуда її перемикання.

Отже, зменшення повного розмаху напруги на бітових лініях є ефективним методом зниження енергоспоживання. У роботах автора для RFSRAM також використовувався підхід обмеженого попереднього заряду шин даних до рівня

приблизно $V_{DD} - V_{th}$, що дозволяє зменшити споживаний струм, оскільки одна з бітових ліній у режимі читання розряджається [29].

Цей метод має природне обмеження: при зростанні кількості рядків збільшується ємність стовпців C_{BL}, що може погіршити швидкодію та зменшити запас для коректного зчитування. Тому обмежений precharge доцільний для відносно невеликих масивів або для регістрових файлів, але потребує ретельної перевірки працездатності на рівні SPICE-моделювання з урахуванням додавання у нетліст екстрагованих паразитних ємностей.

1.3.6. Зниження leakage за рахунок transistor stacking

Для зменшення статичної потужності одним із поширених методів є transistor stacking [11, 13]. Його суть полягає у тому, що два або більше послідовно з'єднаних транзисторів у вимкненому стані суттєво зменшують підпороговий струм. Це часто називають stacking effect або self-reverse bias.

Статичну потужність можна подати як:

$$P_{leak} = I_{leak} V_{DD} \tag{1.7}$$

А підпороговий струм має експоненційну залежність від напруг:

$$I_{sub} \sim e^{\frac{V_{GS} - V_t}{nV_T}} \tag{1.8}$$

Тому навіть невелика зміна ефективної V_{GS}, V_{SB} або V_t може суттєво впливати на leakage.

У роботі з low-power 8T SRAM додаткові buffer transistors були використані для зменшення leakage за рахунок stacking effect, а запропонована комірка показала суттєве зменшення leakage power порівняно зі стандартною 6T SRAM [22].

Однак цей метод також має обмеження. Додаткові транзистори збільшують площу, можуть впливати на затримку та ускладнюють баланс між режимами

читання і запису. Для RFSRAM-компілятора такий підхід може бути корисним не як універсальне рішення, а як локальна архітектурна оптимізація для окремих блоків або режимів standby.

1.3.7. Використання assist circuits для read/write stability

Для покращення роботи SRAM при низьких напругах часто використовуються assist circuits [5, 19, 21]. До них належать read assist, write assist, word-line boosting, negative bit-line, V_{DD} lowering під час запису, body bias та інші схеми.

Загальна ідея assist-підходів полягає у тимчасовій зміні умов роботи комірки під час конкретної операції. Наприклад, для полегшення запису можна тимчасово зменшити ефективну силу pull-up транзистора або збільшити силу access transistor. Для покращення читання можна зменшити disturb на внутрішній вузол або змінити умови попереднього заряду.

У роботі Huang та Chiou зазначається, що cross-point комірки часто потребують write-assist підходів, оскільки stacked access transistors погіршують write-ability і write time у низьковольтних режимах [21]. Водночас ефективність assist circuitry залежить від PVT-варіацій, а самі assist-схеми можуть вимагати складного керування, площі та енергії.

Цей клас методів доцільно розглядати як важливий, але не є основним напрямом для задачі створення параметричного RFSRAM-компілятора, оскільки надмірна кількість assist-схем ускладнює алгоритмізацію генерації реалізацій, верифікацію та timing closure для великої кількості конфігурацій, яку треба перекрити компілятором.

1.3.8. Статистичні методи оцінки стабільності та мінімальної напруги

Окремий напрям досліджень пов’язаний не стільки з безпосереднім зменшенням потужності, скільки з визначенням межі, до якої можна безпечно знижувати напругу живлення [4-10]. Це критично важливо для ULV пам’яті.

У роботах з large-scale variability characterization показано, що зростання process variability є одним із головних обмежень подальшого масштабування SRAM, особливо через використання мінімальних транзисторів у великих масивах [4, 6, 10]. Для SRAM важливо не лише моделювати окрему комірку, а й збирати великі обсяги silicon data для оцінки read stability, writeability та мінімальної робочої напруги V_{MIN}. У даній дисертаційній роботі це важливо, оскільки обґрунтовує необхідність статистичного аналізу SNM. У межах дисертаційного дослідження показано, що розподіл SNM для 8T dual-port SRAM cells може бути описаний нормальним законом, а експериментальні значення стандартного відхилення добре узгоджуються з розрахунками на основі area mismatch coefficient A_{vto}.

Цей підхід дозволяє не просто “зменшувати напругу”, а визначати допустиму нижню межу V_{DD}, при якій зберігається надійність зберігання інформації у комірці пам’яті.

1.4. Вплив технологічного масштабування та статистичних варіацій на стабільність комірки пам’яті

У сучасних CMOS-технологіях при переході до режимів знижених напруг живлення (ULV) визначальним фактором стабільності комірок пам’яті є варіації порогової напруги транзисторів, що визначає поведінку комірки пам’яті, та має статистичний характер розподілу. Ці варіації мають багатофакторну природу та включають як статистичні, так і систематичні складові.

Загалом порогова напруга транзистора може бути представлена як:

$$V_t = V_{t0} + \Delta V_{RDF} + \Delta V_{WF} + \Delta V_L + \Delta V_W \quad (1.9)$$

де:

- ΔV_{RDF}— варіації через випадкові домішки,
- ΔV_{WF}— варіації роботи виходу затвора,
- ΔV_L— варіації довжини каналу,
- ΔV_W— варіації ширини каналу.

Нижче детально розглянуто ключові фактори.

Однією з ключових тенденцій розвитку мікроелектроніки є масштабування технологічних норм, що супроводжується зменшенням геометричних розмірів транзисторів. Незважаючи на очевидні переваги у вигляді підвищення щільності інтеграції та швидкодії, масштабування призводить до суттєвого зростання варіацій електричних параметрів приладів. Для схем статичної пам'яті (SRAM, RFSRAM), що працюють у режимах знижених напруг живлення (Ultra-Low Voltage, ULV), ці варіації стають критичними, оскільки безпосередньо впливають на стабільність збереження інформації.

Відомо, що варіації порогової напруги MOS-транзисторів можуть бути описані нормальним (гаусівським) розподілом:

$$\sigma(V_t) = \frac{A_{Vt}}{\sqrt{W \cdot L}} \quad (1.10)$$

де:

- $\sigma(V_t)$— середньоквадратичне відхилення порогової напруги;
- A_{Vt}— технологічний параметр, що характеризує рівень варіацій;
- W, L— ширина та довжина каналу транзистора.

Таким чином, при зменшенні геометричних розмірів транзистора (що є характерним для масштабування технології) відбувається зростання відносного розкиду порогової напруги [4, 5, 8-10].

Для ULV режимів це має принципове значення, оскільки навіть незначні варіації V_t можуть призводити до:

- втрати стійкості комірки пам’яті;
- деградації SNM;
- появи помилок при читанні.

1.4.1 Варіації роботи виходу Фермі затвора (Work Function Variation)

У свою чергу, робота виходу затвора Φ_m є одним із ключових параметрів, що визначає порогову напругу транзистора:

$$V_t = \Phi_m - \Phi_s - \frac{Q_{ox}}{C_{ox}} + 2\phi_F + \frac{\sqrt{2q\varepsilon_s N_A 2\phi_F}}{C_{ox}} \quad (1.11)$$

де:

- Φ_m— робота виходу металу затвора,
- Φ_s— робота виходу напівпровідника,
- C_{ox}— ємність оксиду,
- N_A— концентрація домішок.

Фізичною причиною варіацій у сучасних технологіях є фактор того, що металевий затвор має зернисту структуру. Кожне зерно може мати різну кристалічну орієнтацію, що призводить до локальних змін роботи виходу:

$$\Phi_m = \Phi_{m0} + \Delta\Phi_m \quad (1.12)$$

Це має безпосередній вплив на порогову напругу:

$$\Delta V_t^{WF} \approx \Delta \Phi_m \tag{1.13}$$

Отже:

$$\sigma_{V_t,WF} \propto \frac{1}{\sqrt{W \cdot L}} \tag{1.14}$$

1.4.2 Випадкові флуктуації домішок:

Даний ефект обумовлений випадковим розташуванням атомів домішок у каналі транзистора. При зменшенні об’єму каналу кількість атомів домішки стає малою, і статистичні флуктуації їх концентрації призводять до значних змін порогової напруги. Даний ефект також зростає при масштабуванні параметрів транзистора, призводить до асиметрії інверторів у SRAM-комірці, що має критичний вплив на SNM.

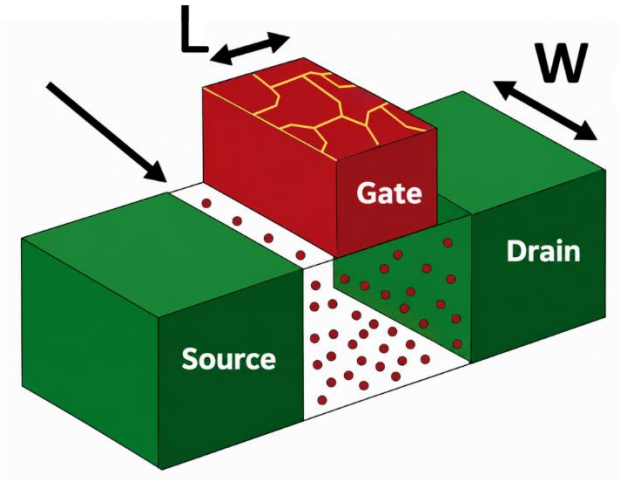


Рис. 1.2 3D-структура MOS транзистора

1.4.3 Варіації довжини каналу та ефект DIBL

Реальна довжина каналу відрізняється від номінальної:

$$L_{eff} = L_g - \Delta L \tag{1.15}$$

де ΔL визначається нерівністю країв (Line Edge Roughness). Це викликає зміну струмів провідності та крутизни передавальної характеристики, що призводить до порушення симетрії інверторів у комірці.

Порогова напруга залежить від напруги стоку:

$$V_t(V_{DS}) = V_{t0} - \eta \cdot V_{DS} \tag{1.16}$$

де:

- η— коефіцієнт DIBL (Drain-Induced Barrier Lowering).

Залежність DIBL від довжини каналу

$$\eta \propto \frac{1}{L_g} \tag{1.17}$$

З урахуванням DIBL:

$$\sigma_{V_t,L} \propto \frac{\sigma_L}{L_g^2} \tag{1.18}$$

Ключовий висновок: при масштабуванні та зменшенні L_g відносна величина варіацій σ_L/L_g зростає, що підсилює DIBL-ефект і збільшує нестабільність порогової напруги.

1.4.4 Варіації ширини каналу (Width Variation)

Зміни ширини каналу виникають через: технологічні обмеження літографії, скруглення кутів та нерівність країв. Цей фактор впливає на струм насичення транзистора та баланс між інверторами комірки, а отже — на стабільність комірки під час операції читання.

Реальна ширина каналу:

$$W_{eff} = W - \Delta W \tag{1.19}$$

де ΔW викликана вищеназваними причинами.

Це має наступну статистичну залежність:

$$\sigma_{V_t,W} \propto \frac{1}{W} \tag{1.20}$$

Таким чином, сукупний ефект варіацій характеризується загальною дисперсією:

$$\sigma_{V_t}^2 = \sigma_{RDF}^2 + \sigma_{WF}^2 + \sigma_L^2 + \sigma_W^2 \tag{1.21}$$

Для ULV-режимів ключовою є залежність:

$$SNM \propto \frac{V_{DD} - V_t}{\sigma_{V_t}} \tag{1.22}$$

Ці фактори є дуже критичними для ULV-режимів, оскільки при малих V_{DD} навіть незначна варіація Vt може зірвати операцію читання або викликати flipping-ефект (перемикання стану) комірки пам’яті.

Таким чином, одночасний вплив зниження напруги живлення V_{DD} та збільшення σ_{V_t} призводить до значного погіршення характеристик SNM, що в свою чергу

призводить до різкого зменшення надійності комірки пам’яті [4, 5, 8-10].

1.5. Урахування паразитних RC-параметрів у post-layout моделюванні пам’яті.

Під час схемотехнічного моделювання макросів пам’яті міжз’єднання часто розглядаються як ідеальні провідники, що не мають власного опору та паразитних ємностей. Такий підхід є прийнятним на ранніх етапах проєктування, однак він не дозволяє повною мірою оцінити характеристики реальної топологічної реалізації пам’яті. Після формування layout-представлення фізичні з’єднання реалізуються через шари полікремнію, металізації, контакти та переходи між шарами, які створюють додаткові паразитні опори та ємності [30-32].

Для макросів RFSRAM вплив паразитних RC-параметрів є особливо суттєвим через регулярну структуру масиву комірок, значну кількість повторюваних елементів, а також наявність протяжних bit-line, word-line, шин живлення, адресних ліній і внутрішніх керуючих сигналів. Паразитні ємності цих провідників впливають на динамічне енергоспоживання, оскільки значна частина енергії пам’яті витрачається на заряд і розряд бітових ліній, word-line та внутрішніх вузлів схеми. Паразитні опори металевих ліній, контактів і via-структур, у свою чергу, формують додаткові RC-затримки, які впливають на форму сигналів, час доступу, максимальну робочу частоту та коректність операцій читання і запису [30-34].

Особливо критичним цей вплив стає у low-voltage та ultra-low-voltage режимах. При зниженні напруги живлення зменшується запас стійкості комірки пам’яті, збільшується чутливість до технологічних варіацій, а додаткові паразитні затримки можуть порушувати часові співвідношення між сигналами керування,

bit-line, word-line та вихідними буферами. Тому оцінювання працездатності згенерованого макросу пам’яті лише на рівні schematic SPICE є недостатнім.

Для отримання більш достовірних характеристик необхідно виконувати післятопологічне SPICE-моделювання з урахуванням extracted RC-параметрів. У такому маршруті після перевірки відповідності схеми та топології формується SPICE-compatible netlist, який містить не лише активні пристрої, але й паразитні опори та ємності міжз’єднань [31, 33-35]. Це дозволяє оцінити реальний вплив топології на access time, затримки внутрішніх сигналів, динамічний струм і максимальну робочу частоту.

Таким чином, урахування паразитних RC-параметрів є необхідною складовою верифікації компілятора пам’яті. Воно забезпечує перехід від ідеалізованої схемотехнічної моделі до реалістичної post-layout характеристики згенерованого макросу RFSRAM і створює основу для подальшого аналізу його працездатності, енергоефективності та часових параметрів.

Постановка наукової проблеми

На основі проведеного аналізу встановлено, що існуючі підходи до проєктування SRAM/RFSRAM не забезпечують повною мірою формалізованого маршруту створення двопортової регістрової пам’яті з пріоритетом низького енергоспоживання, який одночасно враховує SNM/WNM, статистичні варіації параметрів транзисторів, післятопологічні RC-параметри та потреби автоматизованої генерації макросів пам’яті.

Отже, наукова задача полягає у розробленні методів і засобів автоматизованого проєктування двопортової RFSRAM-пам’яті на основі 8T-комірки, що забезпечують стабільну роботу при зниженій напрузі живлення та можливість

post-layout SPICE-верифікації з урахуванням паразитних параметрів топології [3–10, 21, 27, 28, 30, 31].

Висновки за першим розділом.

1. Вбудована SRAM/RFSRAM є критичним IP-блоком сучасних SoC, а її енергоспоживання та час доступу суттєво впливають на характеристики телекомунікаційних, радіотехнічних та автономних систем.

2. Зниження V_{DD} є найефективнішим способом зменшення динамічної потужності, але воно призводить до деградації SNM/WNM, зниження швидкодії та підвищення чутливості до технологічних варіацій.

3. Класична 6T SRAM має фундаментальний конфлікт між read stability і write-ability, тому для low-voltage RFSRAM доцільним є використання 8T read-decoupled комірки з окремими портами читання і запису.

4. Існуючі методи енергооптимізації, зокрема voltage scaling, transistor sizing, assist circuits, bit-line swing reduction і stacking, мають суттєві компроміси між площею, швидкодією, стабільністю та енергоспоживанням.

5. Для проєктування RFSRAM із гарантованою роботою при знижених напругах необхідно враховувати статистичні варіації параметрів MOS-транзисторів і виконувати SNM/WNM-орієнтований вибір розмірів транзисторів.

6. Для отримання достовірних характеристик access time, затримок і струму споживання необхідна post-layout SPICE-верифікація з урахуванням паразитних RC-параметрів топології.

7. Проведений аналіз обґрунтовує необхідність розроблення параметричного RFSRAM-компілятора, який поєднує SNM/WNM-aware проєктування,

статистичне оцінювання стабільності та parasitic-aware post-layout characterization.

РОЗДІЛ 2. Моделі та методи оцінювання стабільності 8T-комірки RFSRAM при знижених напругах живлення

2.1. Проблематика забезпечення стабільності RFSRAM при знижених напругах живлення

Сучасні регістрові файли, що застосовуються у складі мікропроцесорних та телекомунікаційних систем, характеризуються високими вимогами до швидкодії та енергоспоживання. Одним із ключових напрямів підвищення енергоефективності є зниження напруги живлення, що дозволяє суттєво зменшити динамічну складову споживаної потужності.

Разом із тим, зменшення напруги живлення призводить до деградації стабільності комірок пам’яті, що проявляється у зменшенні запасу статичної завадостійкості. Це обумовлено як фізичними обмеженнями MOS-транзисторів, так і статистичними варіаціями їх параметрів, зокрема порогових напруг [3-10].

У відомих роботах [3, 4, 6, 24–26] стабільність комірок SRAM аналізується на основі так званих «батерфляй-кривих», які дозволяють оцінити допустимі рівні шуму, що не призводять до втрати інформації. Проте традиційні підходи орієнтовані переважно на симетричні 6T-комірки та не враховують особливості двопортових структур [17-21].

У межах даного дисертаційного дослідження було запропоновано підхід до визначення параметрів 8T-комірки RFSRAM, орієнтований на забезпечення стабільності при знижених напругах живлення [27, 28]. На відміну від існуючих методів, запропонований підхід базується на аналізі не лише геометрії батерфляй-кривих, але й статистичного розкиду порогових напруг транзисторів.

Як правило, фабрики у специфікаціях на свої технології не дають таких параметрів, як середньоквадратичне відхилення для розподілу Гауса, тому для оцінювання статистичного розкиду порогової напруги використано mismatch-моделі MOS-транзисторів [5, 8, 9]:

$$\sigma(\mathrm{Vt})=\mathrm{A_{Vt}}\sqrt{(2\mathrm{W}\times\mathrm{L})} \qquad (2.1)$$

де σ (Vt) - середньоквадратичне відхилення порогової напруги транзистора з довільними розмірами, $\mathrm{A_{Vt}}$ - середньоквадратичне відхилення порогової напруги транзистора з розмірами W = L = 1 мкм.

Подальший розгляд проведемо для 180-нм CMOS-технології фабрики **TSMC** з коефіцієнтом масштабування **S = 0.84**, що відповідає масштабуванню технологічного процесу до рівня **152 нм**. Для цього введемо до рівняння (2.1) коефіцієнт масштабування:

$$\sigma(\mathrm{Vt})=\mathrm{A_{Vt}}\sqrt{(2\mathrm{W}\times\mathrm{L}\times\ \mathrm{S}^2)} \qquad (2.2)$$

Використання аналітичних виразів для визначення величини **SNM** є недоцільним, оскільки моделі MOS-транзисторів, що надаються фабриками, мають складну структуру і не дозволяють отримати достатньо точні результати при їх спрощенні для аналітичного розв'язання. Тому оцінювання запасу

статичної завадостійкості виконувалося шляхом схемотехнічного моделювання із застосуванням промислових SPICE-моделей.

Аналіз butterfly-кривих для різних технологічних та температурних умов показав, що найгірший випадок (worst-case corner) спостерігається за максимальної температури та в технологічному куті FS, який характеризується мінімальною пороговою напругою n-канальних транзисторів і максимальною пороговою напругою p-канальних транзисторів. Максимальну температуру прийнято рівною **125 °C**.

2.2. Архітектура 8T-комірки двопортової RFSRAM

На рис. 2.1 і 2.2 наведено схему електричну принципову та топологію 8T комірки пам'яті двопортового регістрового файла. Комірка реалізована на восьми транзисторах (**8T**) і містить два незалежні порти: порт запису та порт зчитування [20–22, 27, 28].

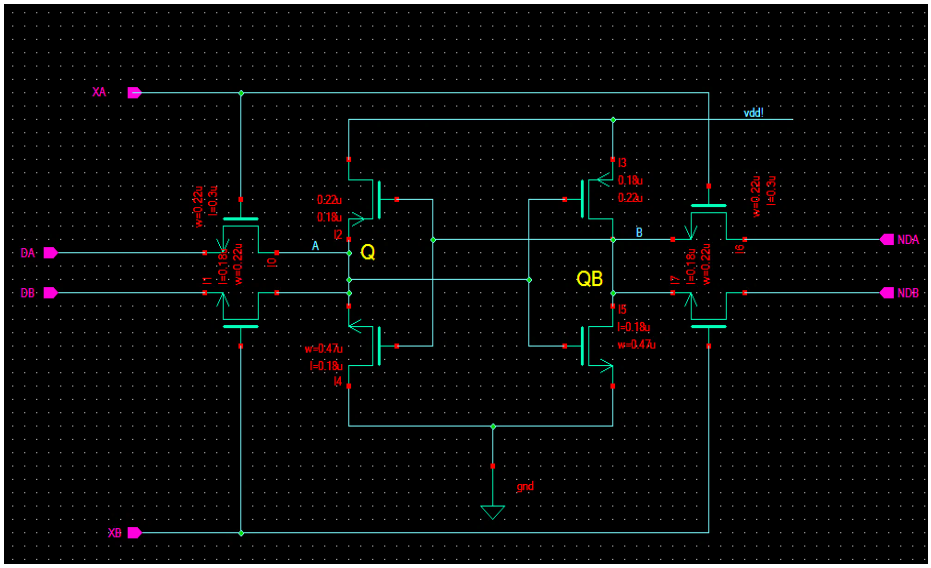


Рис. 2.1 Схема електрична принципова 8T комірки RFSRAM пам’яті

Опис портів:

DA, NDA – шини даних порту читання

DB, NDB – шини даних порту запису

XA – бітова лінія (рядок) порту читання

XB – бітова лінія (рядок) порту запису.

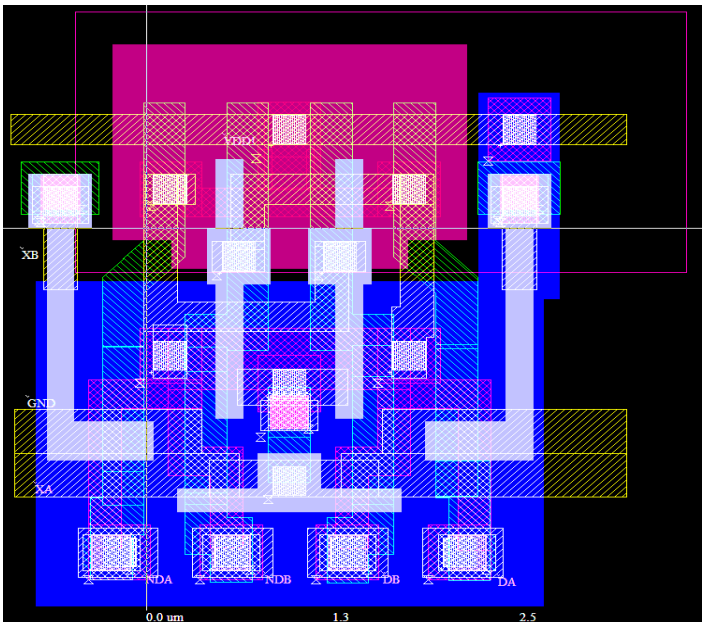


Рис. 2.2 Топологія комірки RFSRAM пам’яті

Кожен із портів має власні дешифратори рядків і стовпців, а також окремі схеми керування.

Під час аналізу необхідно враховувати, що величина SNM у режимі зчитування визначається варіаціями порогових напруг усіх без винятку транзисторів комірки пам'яті. Тому для моделювання статистичного розкиду параметрів транзисторів використано індивідуальні джерела напруги, підключені до затворів транзисторів, значення яких визначаються їх геометричними розмірами відповідно до виразу (2.3) [4, 5, 8, 9].

$$| \Delta V_{t,i} | = 6\sigma(V_{t,i}) = \frac{6A_{Vt,i}}{\sqrt{2W_i L_i S^2}} \qquad (2.3)$$

На рис. 2.3 показано одне плече інвертора з джерелами напруги, що моделюють розкид порогових напруг транзисторів. Друге плече інвертора є повністю аналогічним, за винятком того, що джерела напруги підключені до затворів з протилежною полярністю.

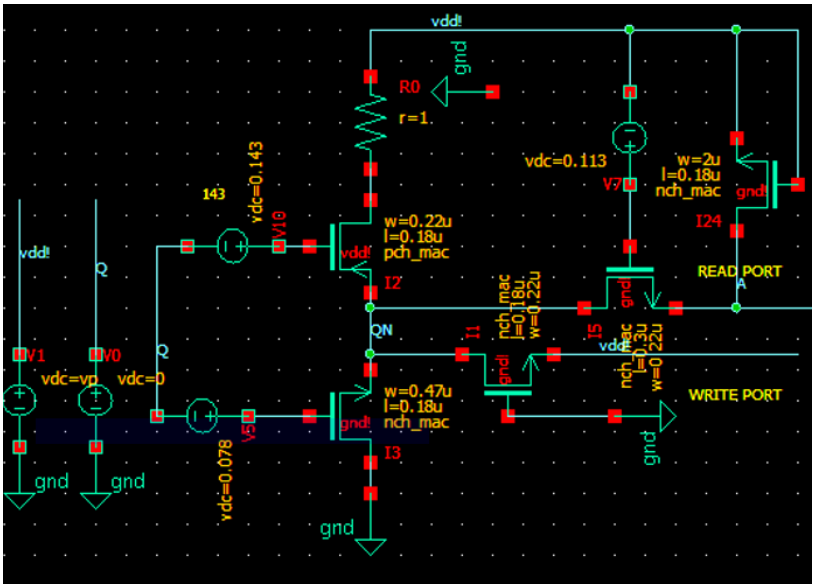


Рис. 2.3. Тестбенч для моделювання розкиду порогових напруг транзисторів комірки

2.3. Методика вибору розмірів транзисторів 8T-комірки

У досліджуваному режимі зчитування адресний транзистор порту запису перебуває у закритому стані, а його затвор підключений до шини землі. Крім транзисторів комірки пам'яті, до шин зчитування підключені n-канальні транзистори попереднього заряджання, оскільки в роботі передбачено обмеження рівня попереднього зарядження шин зчитування з метою зменшення енергоспоживання [7, 11, 13, 29].

Таблиця 2.1

Параметри розміру транзисторів 8T-комірки

Тип MOS	NMOS, inv	NMOS, read	PMOS, inv	NMOS, write
W, мкм	0.25	0.22	0.22	0.22
L, мкм	0.18	0.18	0.18	0.18

Результати моделювання, наведені нижче, показують, що найбільший вплив на запас статичної завадостійкості (**SNM**) мають n-канальні транзистори інверторів комірки пам'яті та адресні транзистори порту читання [3–7, 27, 28].

Крім того, аналіз топології комірки показує, що зміна геометричних параметрів p-канальних навантажувальних транзисторів практично можлива лише за рахунок збільшення довжини каналу (***L***), що призведе до небажаного локального збільшення площі комірки пам'яті. З огляду на це параметри p-канальних транзисторів залишимо незмінними.

Також без змін залишимо параметри адресних транзисторів порту запису, оскільки в режимі зчитування вони не впливають на стабільність комірки.

2.4. Метод оцінювання SNM/WNM на основі butterfly-кривих

Для оцінки стабільності комірки пам’яті використовується графічний метод аналізу батерфляй-кривих, який дозволяє визначити умови існування стійких логічних станів [3, 24–26].

Батерфляй-криві будуються як взаємозалежності:

$$V_{out1} = f(V_{in1}), V_{out2} = f(V_{in2}) \tag{2.4}$$

для двох взаємно інвертованих інверторів, що утворюють бістабільну структуру комірки пам’яті. На схемотехнічному рівні комірка є нелінійною електронною схемою зі зворотним зв’язком між двома інверторами; її статичну стійкість оцінюють за їхніми передавальними характеристиками.

На відміну від класичного підходу, в даній роботі запропоновано використовувати критерій, заснований на кількості точок перетину батерфляй-

кривих з урахуванням статистичного розкиду порогових напруг транзисторів [27, 28].

Показано, що:

- наявність **трьох точок перетину** відповідає стійкому режиму збереження даних [3, 27, 28];
- наявність **однієї точки перетину та однієї точки дотику** відповідає граничному (критичному) режиму;
- відсутність трьох точок перетину призводить до втрати інформації.

Таким чином, вводиться поняття **критичної напруги живлення** $V_{DD,crit}$, яка визначається як мінімальне значення напруги, при якому виконується умова переходу від нестабільного до стабільного режиму. Запропонований підхід дозволяє перейти від емпіричного підбору параметрів до формалізованої процедури синтезу комірки пам’яті.

Визначення SNM

Запас статичної стійкості визначається за допомогою так званої “метеликової” діаграми (butterfly curve), яка утворюється перетином передавальних характеристик двох інверторів комірки. Графічним способом SNM визначається як сторона найбільшого квадрата, який може бути вписаний між цими характеристиками:

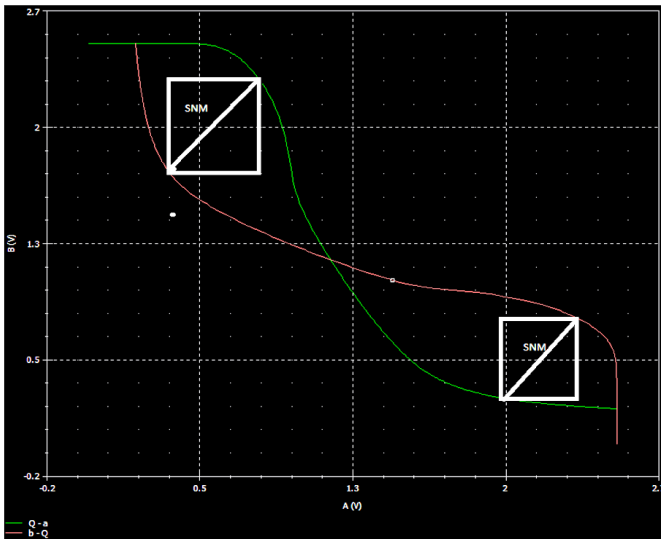


Рис. 2.4. Приклад графічного визначення SNM

Цей підхід відповідає класичному визначенню SNM і використовується для оцінки стійкості SRAM-комірок [3].

Під час аналізу butterfly-діаграм за цільове значення мінімальної напруги живлення приймемо $V_{DD\,\min} \geq 0.6$ В, а номінальну напругу живлення — $V_{DD} = 0.7$ В ± 10 %.

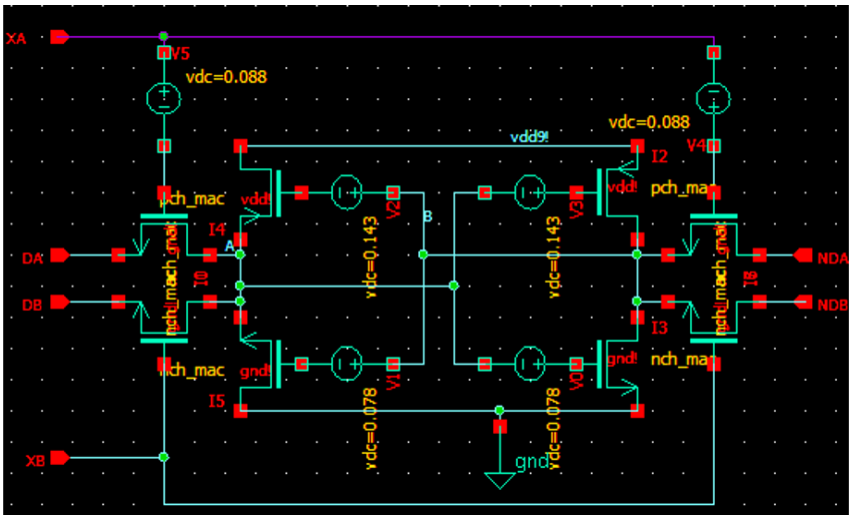


Рис. 2.5. Схема електрична принципова тестбенчу комірки пам’яті з джерелами напруги, що моделюють розкид значень порогових напруг

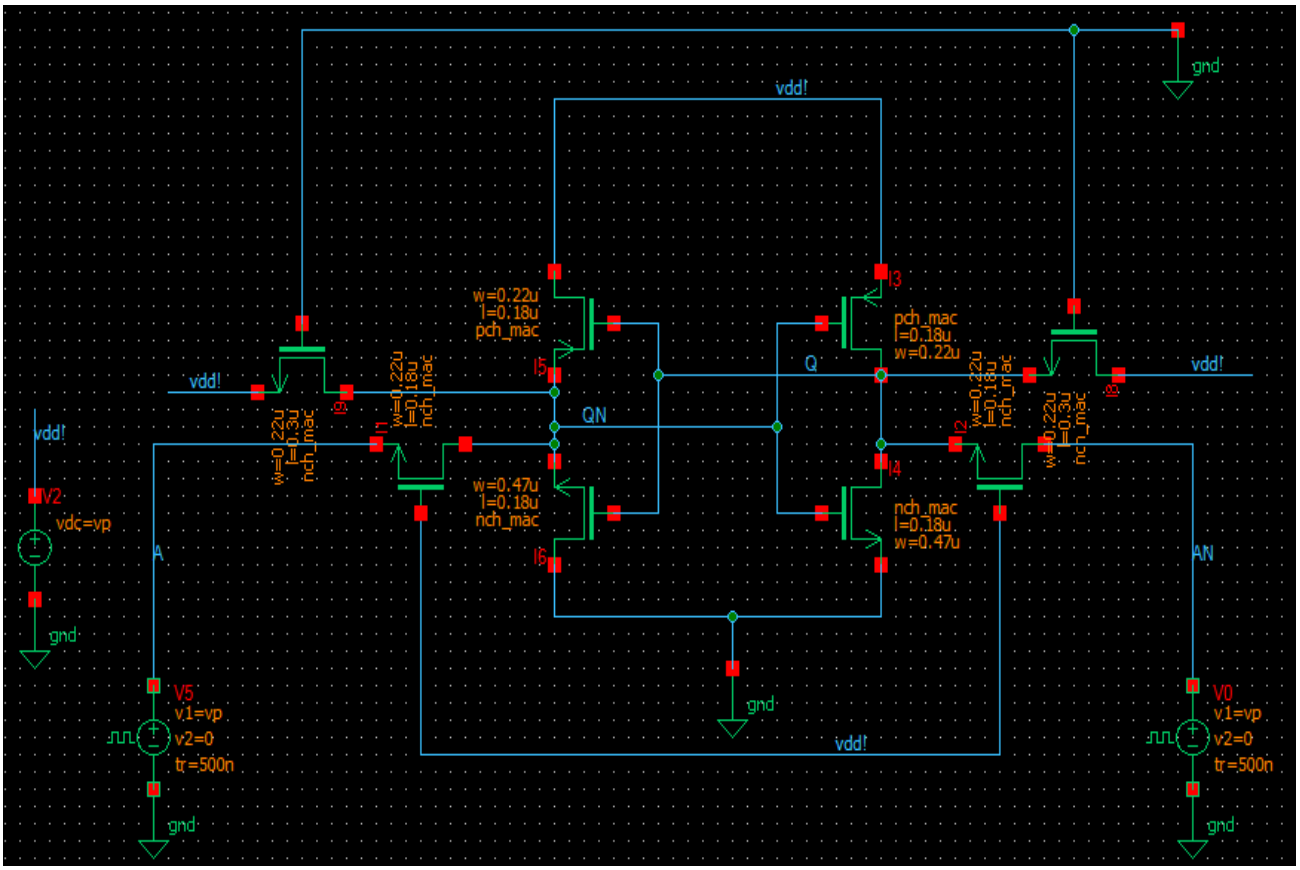


Рис. 2.6. Схема електрична принципова тестбенчу комірки пам’яті для моделювання WNM

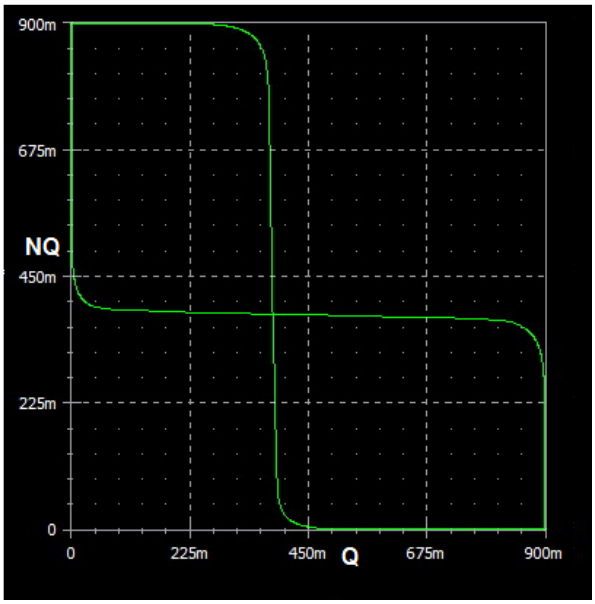


Рис. 2.7. Butterfly-діаграма неоптимізованої комірки в режимі зберігання даних при V_{DD} = 0.90 В, P = SS, T = 85°C, S = 0.84.

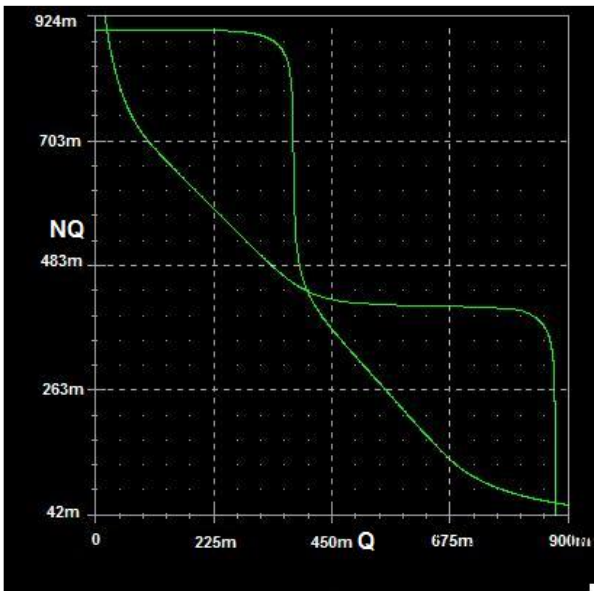


Рис. 2.8. Butterfly-діаграма неоптимізованої комірки в режимі читання даних при V_{DD} = 0.90 В, P = SS, T =85°C, S = 0.84

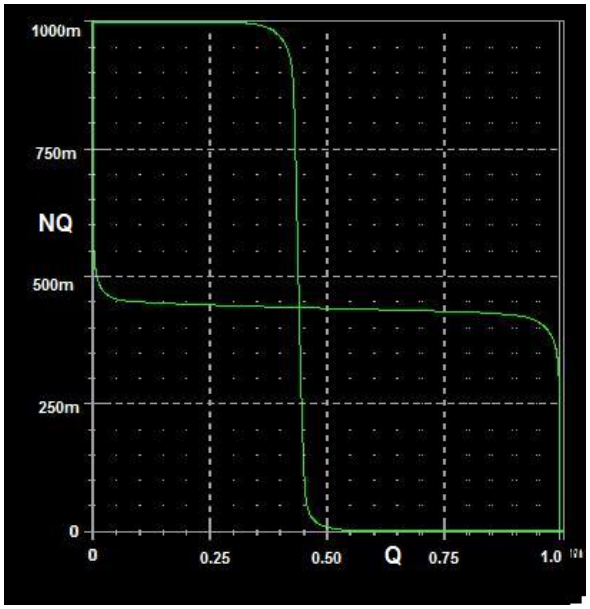


Рис. 2.9. Butterfly-діаграма неоптимізованої комірки в режимі зберігання даних при V_{DD} = 1.00 В, P = TT, T =25°C, S = 0.84

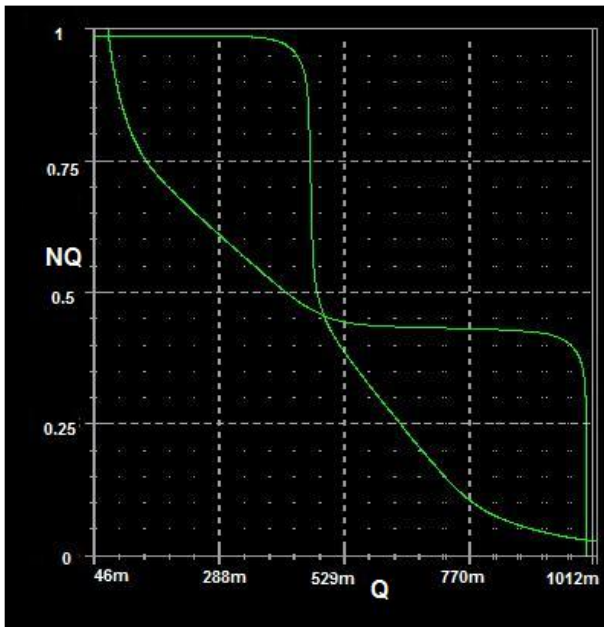


Рис. 2.10. Butterfly-діаграма неоптимізованої комірки в режимі читання даних при V_{DD} = 1.00 В, P = TT, T =25°C, S = 0.84

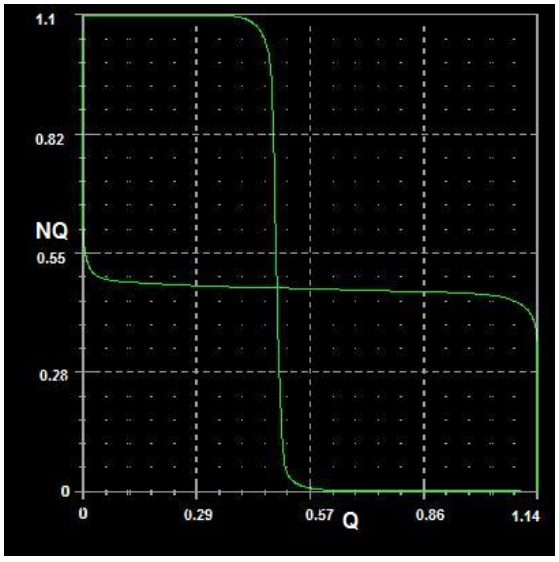


Рис. 2.11. Butterfly-діаграма неоптимізованої комірки в режимі зберігання даних при V_{DD} = 1.1 В, P = FF, T =-40°C, S = 0.84

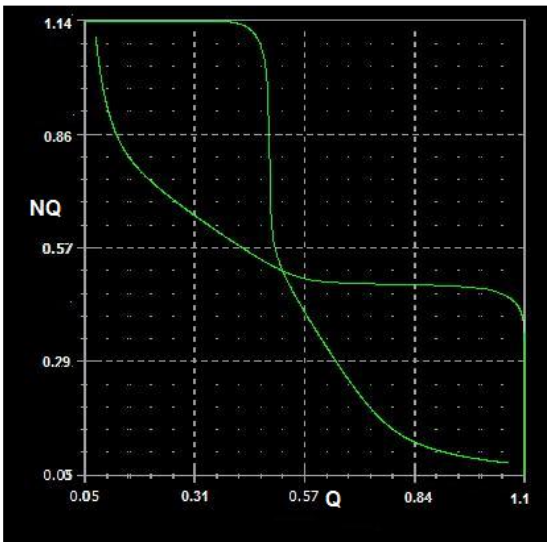


Рис. 2.12. Butterfly-діаграма неоптимізованої комірки в режимі читання даних при V_{DD} = 1.1 В, P = FF, T =-40°C, S = 0.84

У роботах, присвячених моделюванню mismatch параметрів MOS-транзисторів, коефіцієнт A_{Vt} використовується для оцінювання стандартного відхилення порогової напруги залежно від геометричних розмірів транзистора [5, 8, 9]. Для досліджуваної 180-нм CMOS-технології при напрузі живлення ядра 1.8 В товщина підзатворного оксиду для найгіршого технологічного кута становить 4.2 нм. За цих умов коефіцієнт A_{Vt} дорівнює 4.51 мВ·мкм для n-канальних транзисторів та 5.63 мВ·мкм для p-канальних транзисторів.

2.5. Метод оцінювання SNM на основі butterfly-кривих

Метою подальшої оптимізації є визначення таких геометричних параметрів W/L n-канального транзистора інвертора та адресного транзистора порту зчитування, які забезпечують мінімальну напругу живлення $V_{DD\,\min} \geq 0.6$ В.

Під мінімальною напругою живлення $V_{DD\,\min}$ будемо розуміти таке значення напруги V_{DD}, за якого butterfly-діаграма характеризується однією точкою

перетину передавальних характеристик та однією точкою їх дотику [3, 27, 28]. Саме ця умова відповідає граничному стану статичної стійкості комірки пам'яті. У табл. 2.2 наведено залежність величини 6σ для n-канального транзистора інвертора комірки пам'яті від ширини каналу за фіксованої довжини каналу L = 0.18 мкм [4, 5, 8, 9].

Таблиця 2.2

Залежність 6σ для транзистора $NMOS_{inv}$ комірки пам'яті від ширини каналу

$NMOS_{inv}$, (L=0.18 мкм) W мкм,	0.25	0.35	0.47
6σ мВ, S=0.84	107	91	78
6σ мВ, S=1	90	76	66

У табл. 2.3 наведено геометричні параметри адресних NMOS-транзисторів порту зчитування та PMOS-транзисторів інверторів комірки пам'яті, а також відповідні значення величини 6σ розкиду їх порогових напруг для коефіцієнтів масштабування S = 1 та S = 0.84.

Таблиця 2.3.

Геометричні параметри…

Тип MOS,	$NMOS_{addr}$		$NMOS_{addr}$		$PMOS_{inv}$	
W/L, мкм	0.22 / 0.3		0.22 / 0.18		0.22 / 0.18	
S	S=1	S=0.84	S=1	S=0.84	S=1	S=0.84
6σ, мВ	74.3	89	96	114	120	143

2.6. SPICE-моделювання та визначення критичної напруги живлення

На рис. 2.13 – 2.23 наведено результати схемотехнічного моделювання у вигляді butterfly-діаграм та часових діаграм, отримані на різних етапах оптимізації геометричних параметрів транзисторів комірки пам'яті.

Аналіз наведених залежностей дозволяє оцінити вплив зміни співвідношення W/L окремих транзисторів на величину запасу статичної завадостійкості (SNM) та визначити оптимальні розміри транзисторів, що забезпечують необхідний рівень стійкості комірки при зниженій напрузі живлення [3–7, 18, 19, 27, 28].

Аналіз часових діаграм (рис. 2.13–2.14) показує, що при зменшенні напруги живлення до 0.75 В комірка втрачає стійкість під час операції зчитування, що проявляється зміною її внутрішнього стану (рис. 2.15).

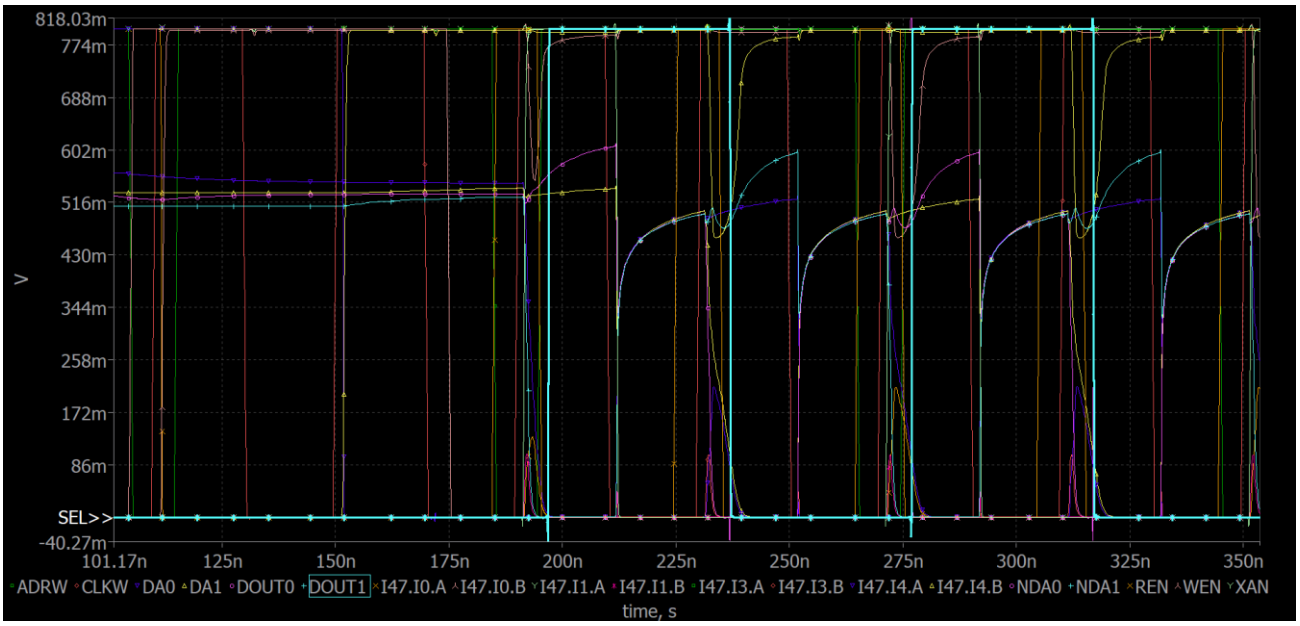


Рис. 2.13. Часова діаграма комірки пам'яті при W=0.25 мкм, L=0.18 мкм, S=0.84, V_{DD}=0.80 В.

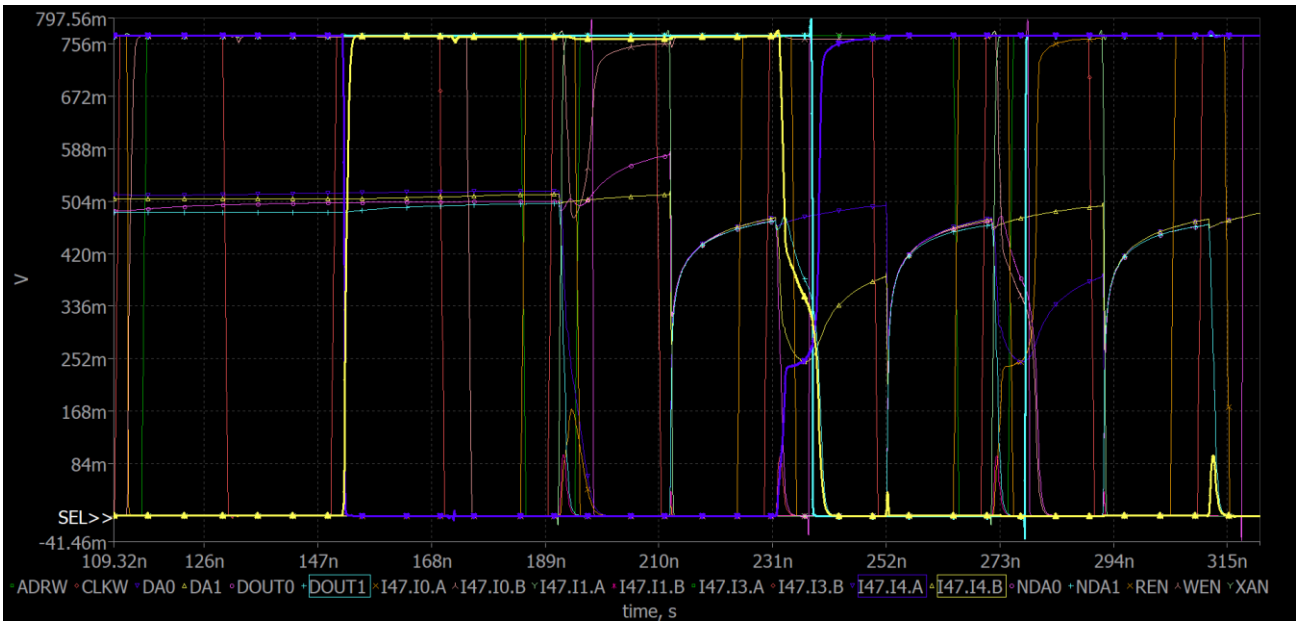


Рис. 2.14. Часова діаграма комірки пам'яті при W=0.25 мкм, L=0.18 мкм, S=0.84, V_{DD}=0.75 В.

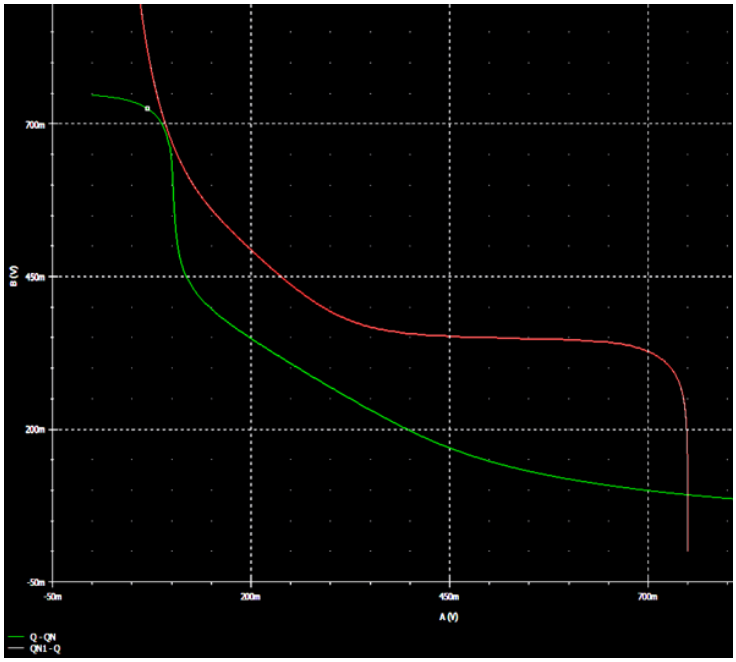


Рис. 2.15. Butterfly-діаграма комірки пам'яті при W=0.25 мкм, L=0.18 мкм, S=0.84, V_{DD}= 0.75 В.

Після збільшення довжини каналу адресного транзистора порту зчитування до L = 0.30 мкм робота комірки при V_{DD} = 0.75 В повністю відновлюється, що

підтверджується результатами часового моделювання (рис. 2.16) та аналізом butterfly-діаграми (рис. 2.17), яка чітко визначає три точки перетину кривих.

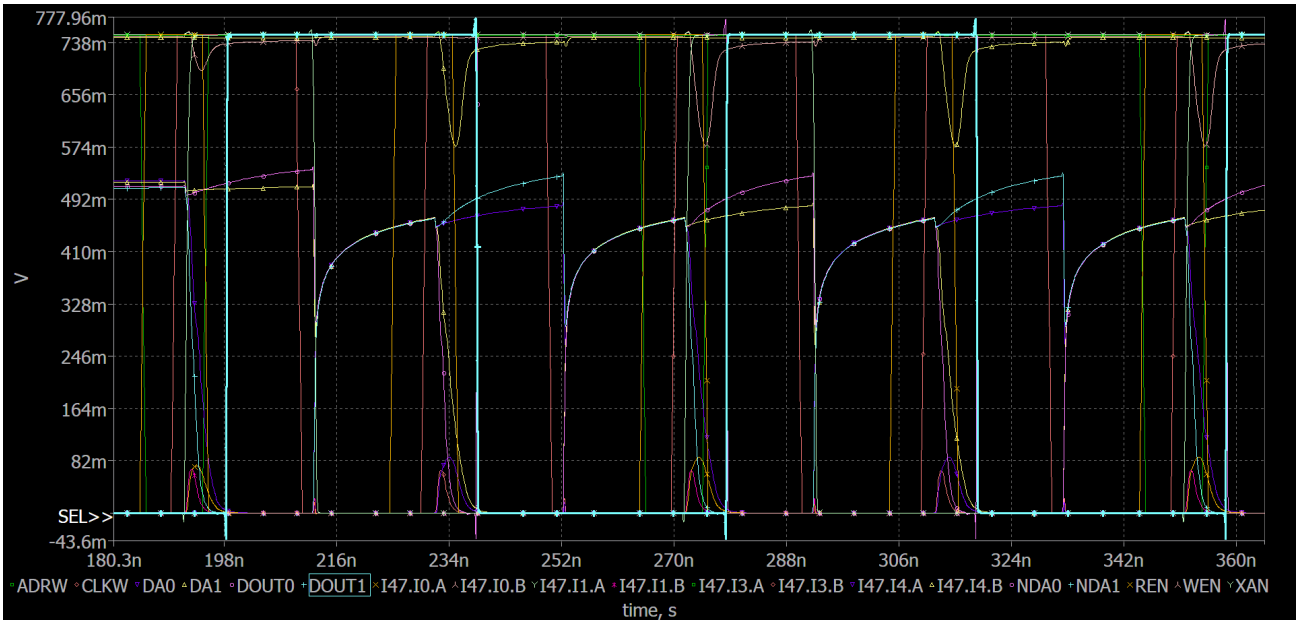


Рис. 2.16. Часова діаграма комірки пам'яті при W=0.25 мкм, L=0.30 мкм, S=0.84, V_{DD}=0.75 В.

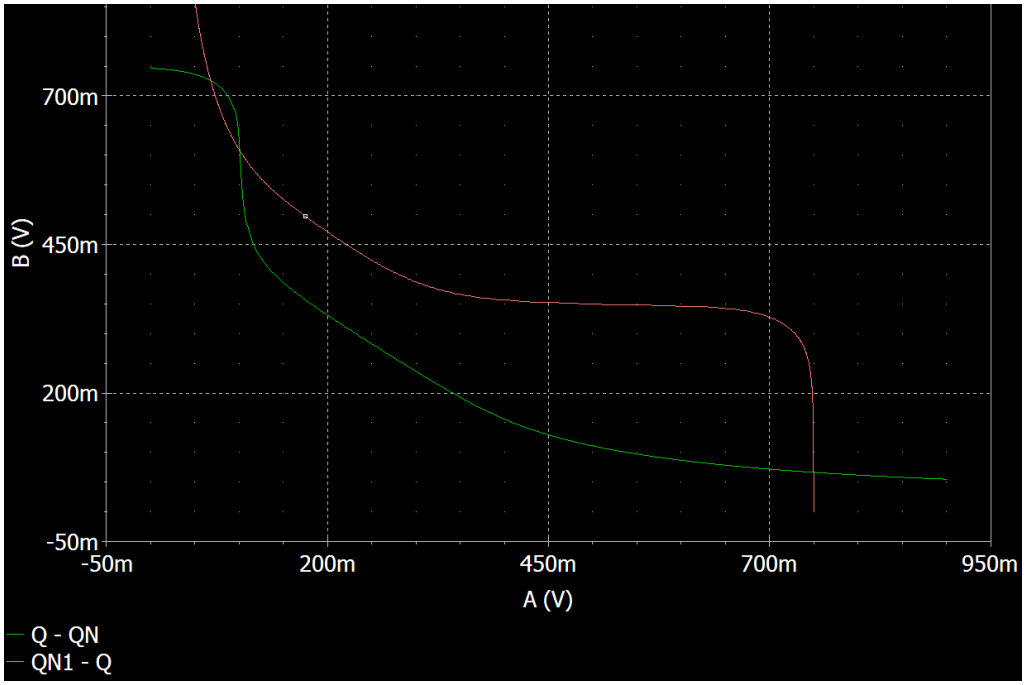


Рис. 2.17. Butterfly-діаграма комірки пам'яті при W=0.25 мкм, L=0.30 мкм, S=0.84, V_{DD}=0.75 В.

Результати моделювання (рис. 2.18–2.19) демонструють, що після подальшої оптимізації геометричних параметрів транзисторів мінімальна робоча напруга у режимі читання зменшується до 0.60 В [27, 28]. З butterfly-діаграми (рис. 2.20) добре видно, що при V_{DD} = 0.60 В комірка перебуває на межі стійкості, тоді як при V_{DD} = 0.62 В операція читання виконується коректно.

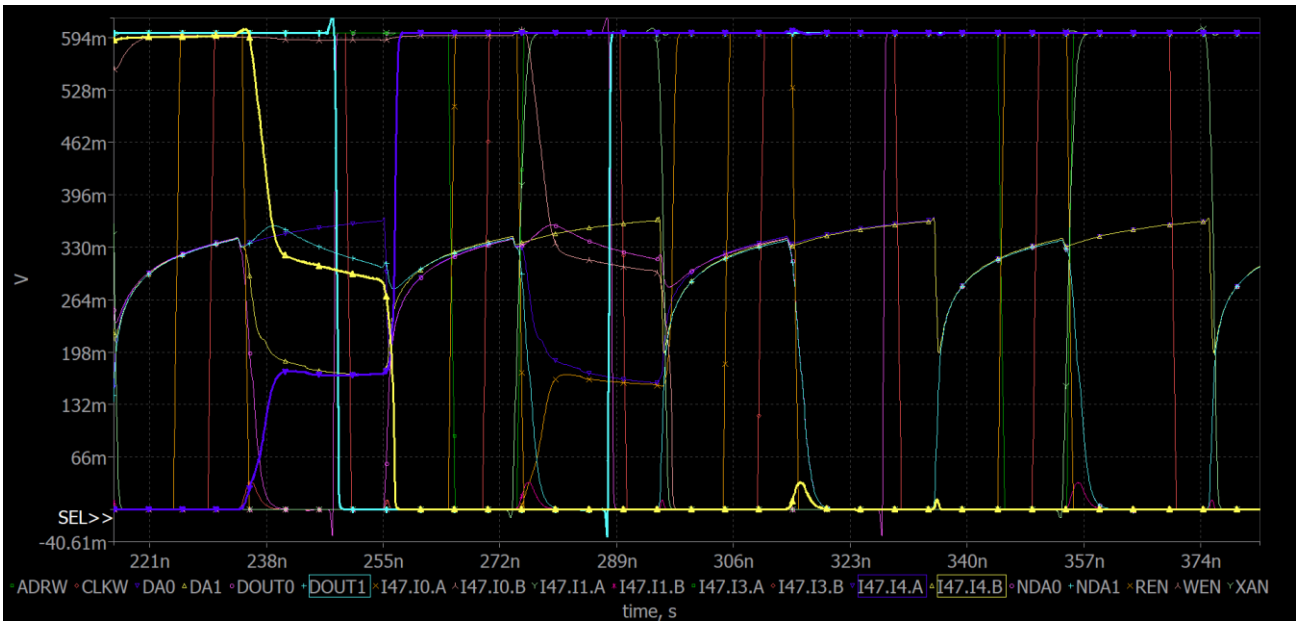


Рис. 2.18. Часова діаграма оптимізованої комірки пам'яті при W = 0.47 мкм, L = 0.30 мкм, S = 0.84, V_{DD} = 0.60 В.

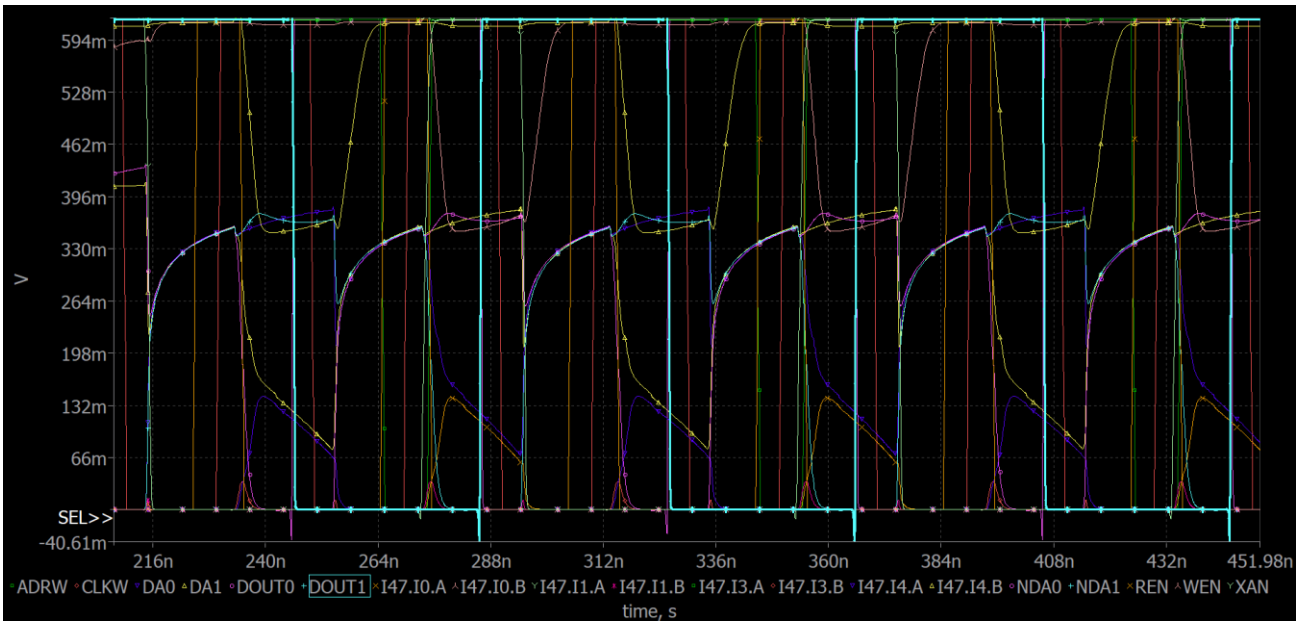


Рис. 2.19. Часова діаграма оптимізованої комірки пам'яті при W = 0.47 мкм, L = 0.30 мкм, S = 0.84, V_{DD} = 0.62 В.

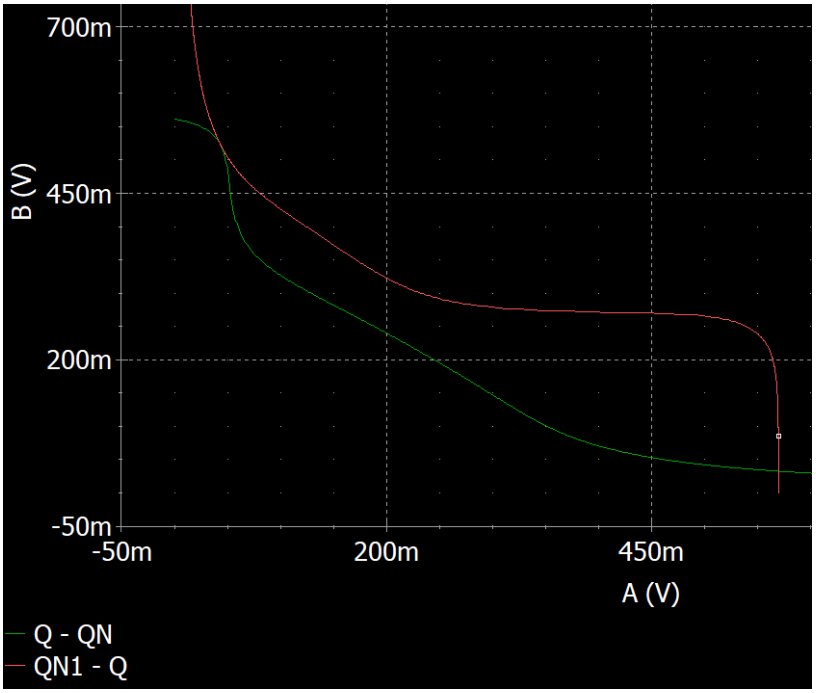


Рис. 2.20. Butterfly-діаграма оптимізованої комірки пам'яті при W = 0.47 мкм, L = 0.30 мкм, S = 0.84, V_{DD} = 0.60 В.

У режимі зберігання даних butterfly-діаграма має три точки перетину (рис. 2.21), що відповідає двом стійким станам рівноваги та підтверджує надійне зберігання інформації навіть за мінімальної напруги живлення

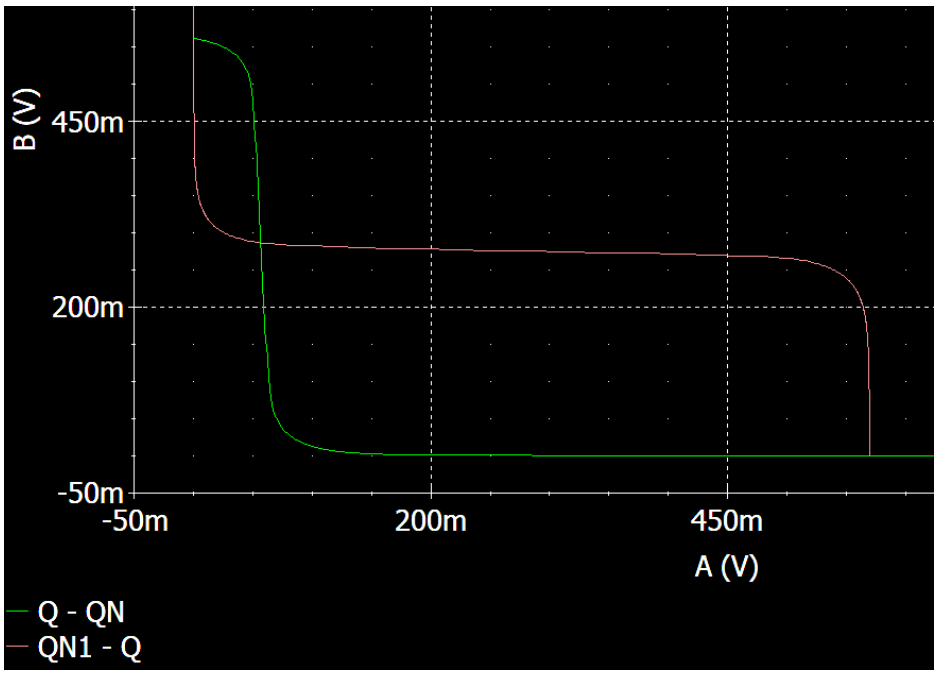


Рис. 2.21 Butterfly-діаграма оптимізованої комірки в режимі зберігання даних при W = 0.47 мкм, L = 0.30, S = 0.84, V_{DD} = 0.62 В.

Для розробки та перевірки методики вибору параметрів 8T-комірки використано 180-нм CMOS-технологію з коефіцієнтом масштабування $S = 0.84$ технології. Порівняння цих результатів із первинними для S = 1 (Рис. 2.22–2.23) показує, що масштабування технології призводить до зменшення запасу статичної завадостійкості комірки, що потребує відповідної оптимізації геометричних параметрів транзисторів [4–6, 10, 27, 28].

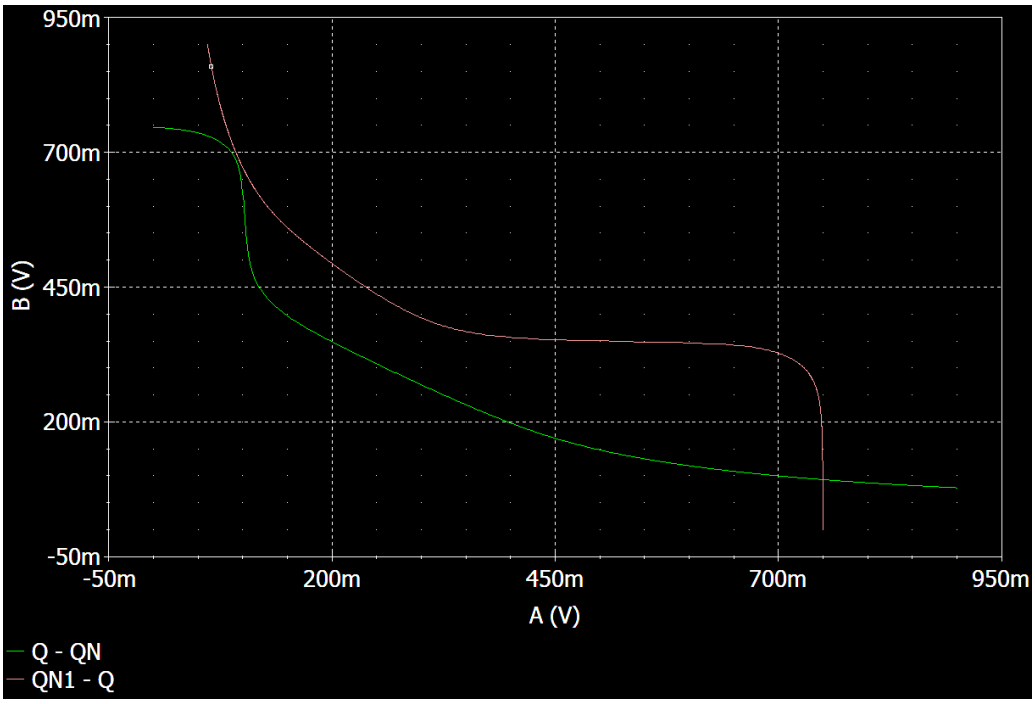


Рис. 2.22 Butterfly-діаграма комірки пам'яті при W=0.25 мкм, L=0.18 мкм, S=1, V_{DD}=0.60 В.

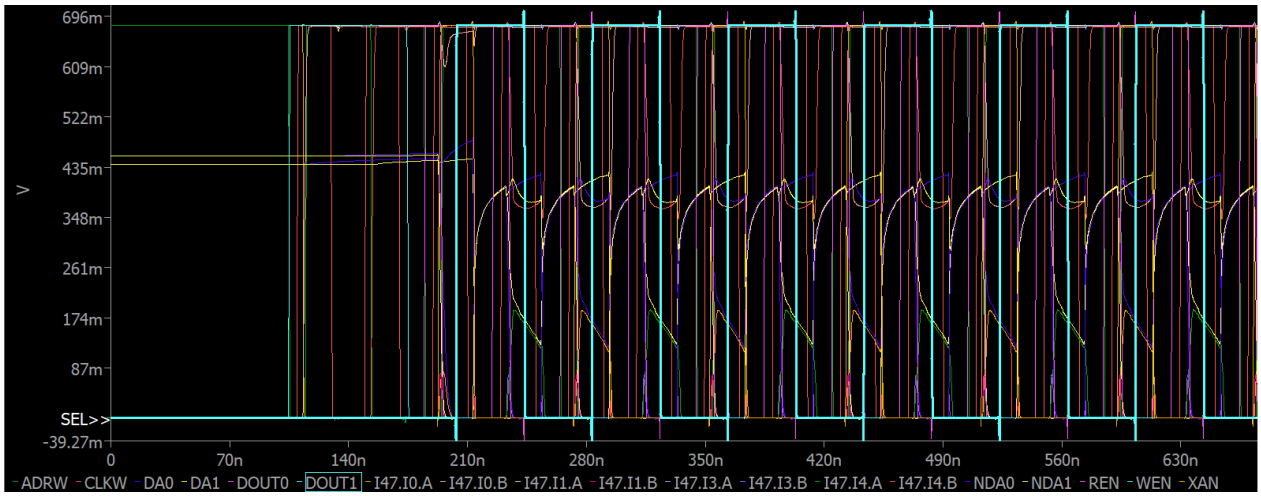


Рис. 2.23 Часова діаграма комірки пам'яті при W=0.25 мкм, L=0.18 мкм, S=1, V_{DD}=0.63 В.

Окремо, для експериментального підтвердження статистичного характеру SNM-розподілу, проаналізовано результати вимірювань реальних тестових структур 8T dual-port SRAM-комірок у кремнії, виготовлених у 250-нм CMOS-технології, результати яких буде розглянуто далі.

2.7. Визначення WNM за результатами моделювання

Запас завадостійкості під час запису (Write Noise Margin, WNM) визначається як мінімальне значення напруги на бітовій лінії, необхідне для зміни логічного стану комірки SRAM [7].

Для визначення величини WNM використано метод SWEEP. Під час моделювання на обидві бітові лінії подається напруга V_{DD}, тоді як у вузлах зберігання комірки встановлюються логічні стани «0» та «1». Далі виконується покрокове зменшення напруги на одній із бітових ліній від рівня V_{DD}, що дозволяє визначити мінімальну напругу, за якої відбувається перемикання стану комірки.

Запас завадостійкості під час запису переважно визначається співвідношенням між провідністю навантажувальних (pull-up) та керувальних транзисторів комірки (pull-up ratio) [7, 17-19]. Зі збільшенням коефіцієнта pull-up ratio величина WNM зменшується, тому для забезпечення надійного запису співвідношення параметрів транзисторів комірки повинно бути відповідним чином оптимізоване.

Зведені результати моделювання SNM та WNM наведено в табл. 2.4, а відповідні WNM-діаграми для основних PVT-кутів — на рис. 2.24–2.26.

Таблиця 2.4

Результати моделювання SNM та WNM

PVT-кут	**SNM (режим утримання) , мВ**	**SNM (режим читання), мВ**	**WNM , мВ**
0.8 В, ss, -40°C	302	174	205
0.9 В, ss, 85°C	338	215	283
1.0 В, tt, 25°C	353	234	370
1.1 В, ff, -40°C	369	257	463
0.8 В, sf, -40°C	346	212	158

0.8 В, fs, -40°C	296	177	342
0.8 В, sf, 85°C	336	219	181
0.8 В, fs, 85°C	262	168	382
0.8 В, ff, -40°C	367	208	303
0.8 В, ss, 85°C	279	180	301
0.8 В, ff, 85°C	222	197	339

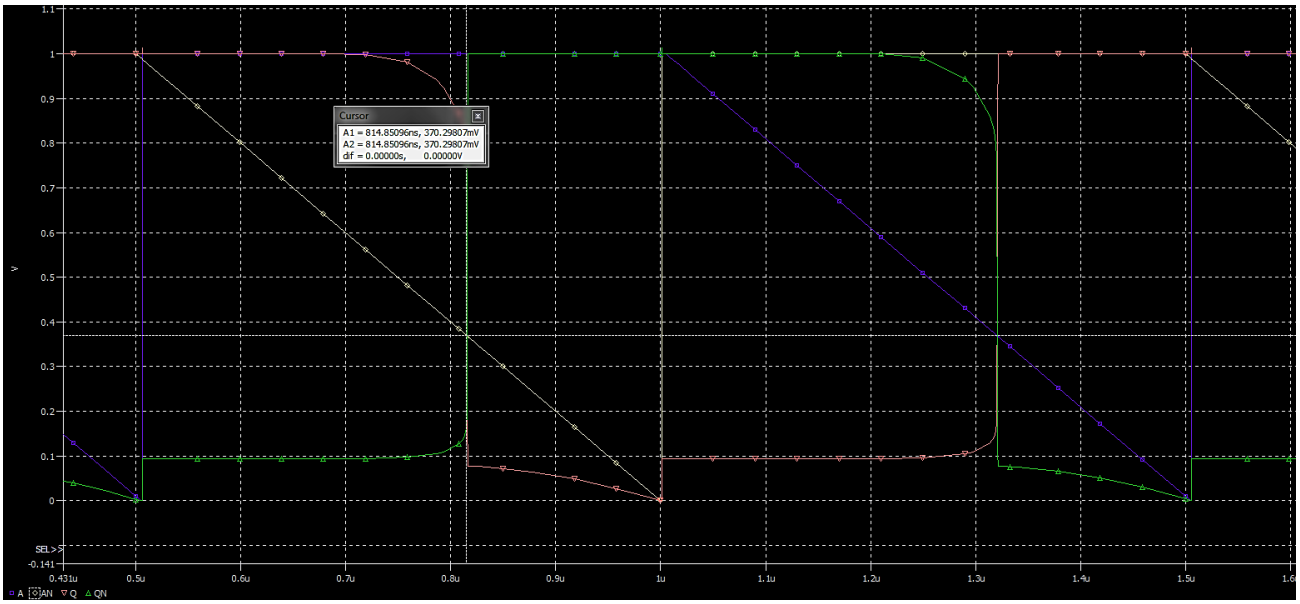


Рис. 2.24 WNM діаграма комірки пам'яті при T = 25°C, P = TT, V_{DD} = 1.00 В.

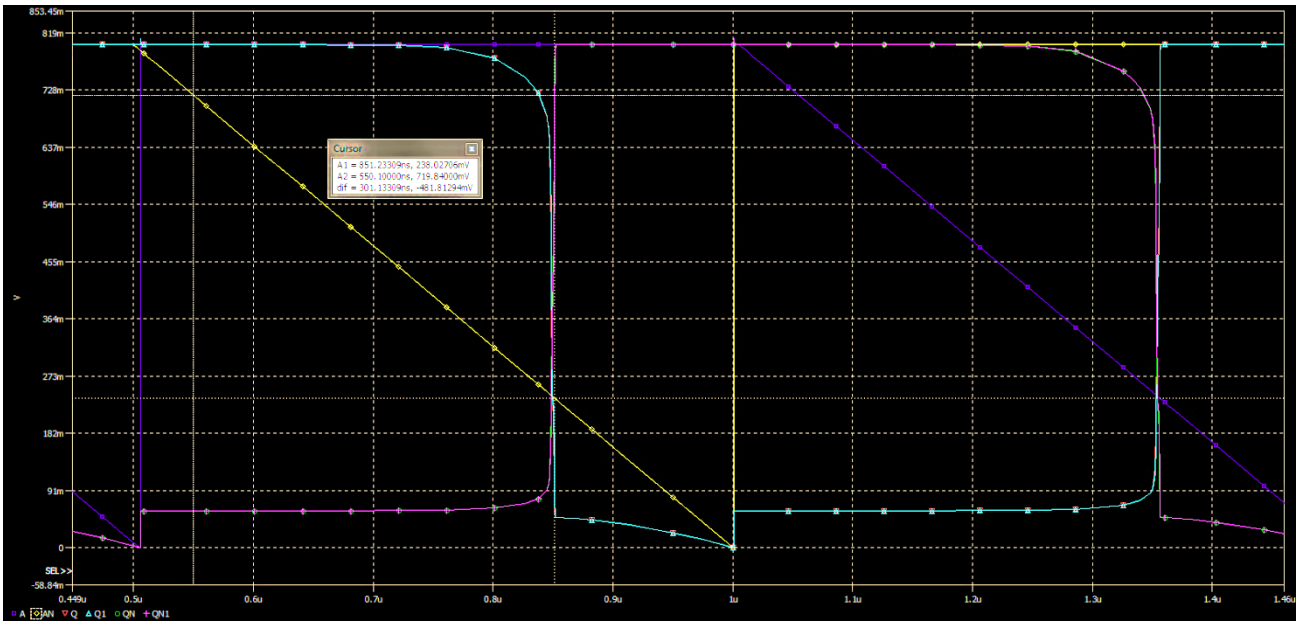


Рис. 2.25 WNM діаграма комірки пам'яті при T = 85°C, P = SS, V_{DD} = 0.80 В.

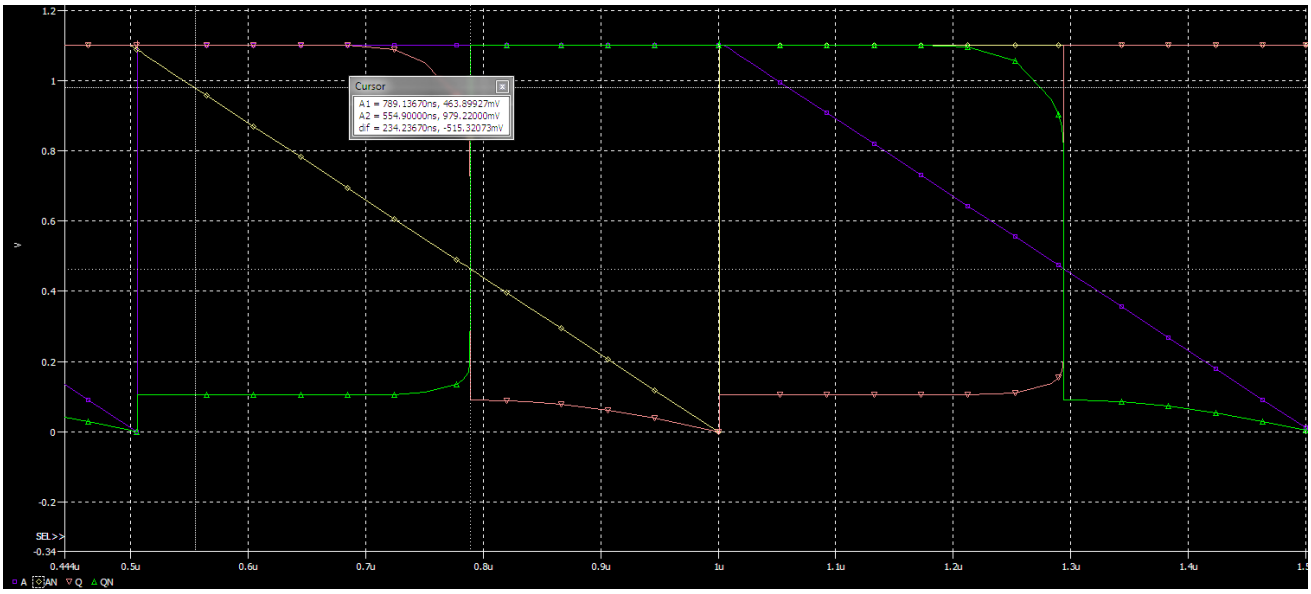


Рис. 2.26 WNM діаграма комірки пам'яті при T = -40°C, P = FF, V_{DD} = 1.1 В.

У результаті проведеної оптимізації отримано комірку пам'яті з мінімальною напругою для надійної операції читання $V_{DD\,\min}$ = 0.6 В [27, 28]. Для цього ширину керуючого n-канального транзистора інвертора було збільшено до W = 0.47 мкм, а довжину каналу адресного транзистора порту зчитування — до L = 0.30 мкм. Площа оптимізованої комірки становить 3.08 × 3.245 = 9.995 мкм².

Для порівняння, площа вихідної комірки з мінімальною напругою зчитування 0.68 В, при W = 0.25 мкм для керувального транзистора інвертора та L = 0.18 мкм для адресного транзистора порту зчитування, становить 3.08 × 3.025 = 9.317 мкм². Таким чином, запропонована оптимізація забезпечує зменшення мінімальної робочої напруги живлення на 80 мВ (з 0.68 В до 0.60 В) ціною збільшення площі комірки приблизно на 7 %.

Такий результат є фізично закономірним, оскільки підвищення стійкості комірки досягається шляхом збільшення геометричних розмірів найбільш критичних транзисторів, що, своєю чергою, зменшує статистичний розкид їх порогових напруг.

2.8 Оптимізація параметрів комірки пам'яті для компілятора регістрових файлів за 130-нм технологією

Компілятор регістрових файлів RFSRAM 130 нм був розроблений відповідно до вимог фабрики для забезпечення максимальної тактової частоти 400 МГц при номінальній напрузі живлення V_{DD} = 1.2 В ± 10 % з додатковою можливістю роботи в режимі зниженого енергоспоживання при V_{DD} = 0.7 В ± 10 %. Технологія є базовою, масштабування не використовувалося, тобто всі моделювання та розрахунки виконувалися без застосування коефіцієнта масштабування геометричних параметрів.

Компілятор підтримує генерацію трьох типів регістрових файлів:

- із єдиним сигналом синхронізації (CLK) для портів запису та зчитування;
- із двома незалежними сигналами синхронізації (CLKR, CLKW) для портів запису та зчитування;
- із двома незалежними сигналами синхронізації (CLKR, CLKW) для портів запису та зчитування, доповненими контролером конфліктів (Conflict Controller), який запобігає виникненню невизначених результатів зчитування у випадку, коли після початку операції читання виконується запис у той самий рядок пам'яті.

Крім того, у компіляторі передбачено можливість генерації реалізацій регістрового файла із вбудованим блоком самотестування (Built-In Self-Test, BIST). Використання цього режиму дозволяє зменшити кількість контактних майданчиків кристала та спростити тестування готового виробу. Архітектуру та принципи роботи блоку BIST детально розглянуто в Розділі 4.

Основною метою оптимізації є забезпечення працездатності комірки пам'яті за найгірших умов експлуатації (worst-case corner): технологічний кут SS,

температура 85 °C та напруга живлення V_{DD} = 0.7 В ± 10 %, що відповідає режиму зниженого енергоспоживання [4–10, 27, 28].

Для досліджуваної 130-нм CMOS-технології при напрузі живлення ядра 1.2 В товщина підзатворного оксиду становить 2 нм. За цих умов коефіцієнт A_{Vt} дорівнює 4.3 мВ·мкм для n-канальних транзисторів та 5.2 мВ·мкм для p-канальних транзисторів.

У табл. 2.5 наведено геометричні параметри NMOS-транзисторів інвертора комірки пам'яті до та після проведення оптимізації. У табл. 2.6 подано розміри NMOS- і PMOS-транзисторів мінімальної геометрії, параметри яких у процесі оптимізації не змінювалися.

Таблиця 2.5

Геометричні параметри NMOS-транзисторів інвертора комірки пам'яті

$NMOS_{inv}$, L=0.13 мкм	До оптимізації	Після оптимізації
W, мкм	0.2	0.3
6σ мВ, S=1	113	87

Таблиця 2.6

Розміри NMOS- і PMOS-транзисторів мінімальної геометрії

Тип MOS	NMOS	PMOS
W/L, мкм	0.15/0.13	0.15/0.13
6σ, мВ	131	158

Порівняння butterfly-діаграм (рис. 2.27–2.28) показує, що збільшення ширини керувального NMOS-транзистора інвертора з 0.20 до 0.30 мкм дозволяє зменшити мінімальну напругу надійного зчитування з 0.72 В до 0.60 В [27, 28].

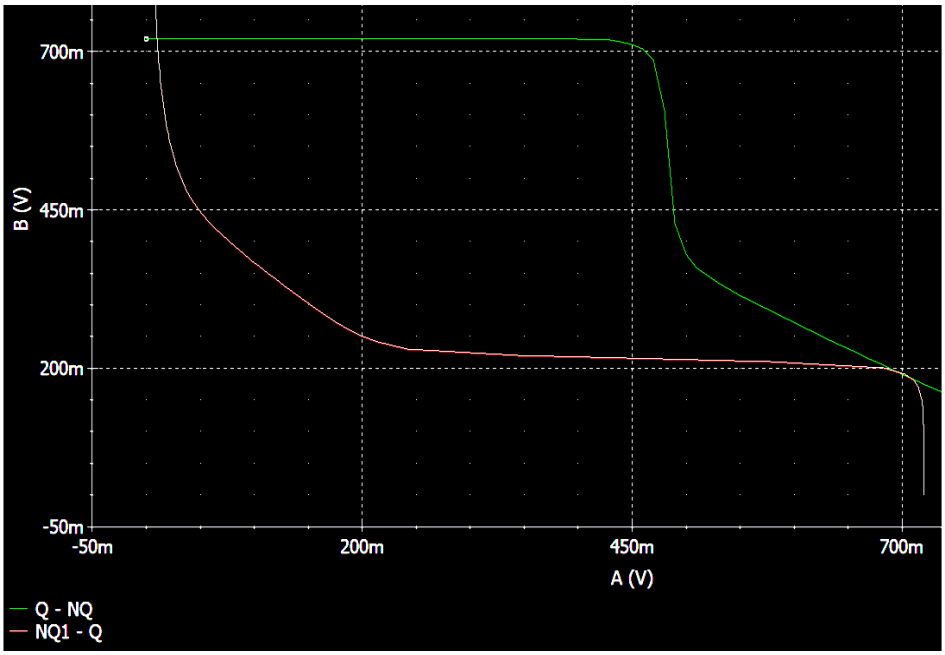


Рис. 2.27. Butterfly-діаграма комірки пам'яті при W=0.20 мкм, L=0.13 мкм, V_{DD} = 0.72 В.

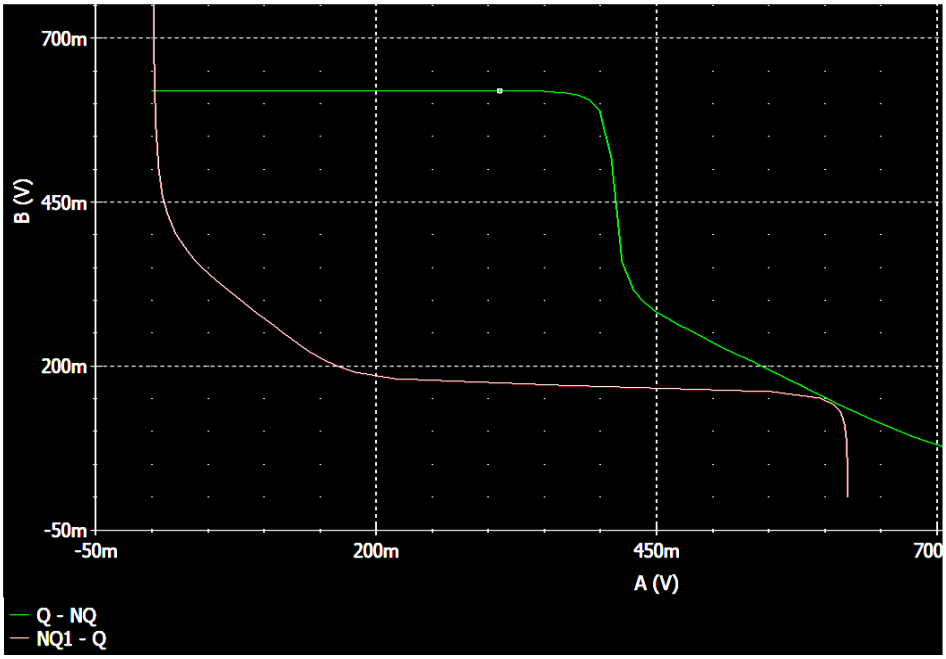


Рис. 2.28. Butterfly-діаграма комірки пам'яті при W=0.30 мкм, L=0.13 мкм, V_{DD} = 0.60 В

Як видно з часових діаграм (рис. 2.29–2.30), при V_{DD} = 0.75 В комірка працює коректно, тоді як при зниженні напруги живлення до 0.70 В під час операції зчитування відбувається зміна її внутрішнього стану, що свідчить про втрату стійкості.

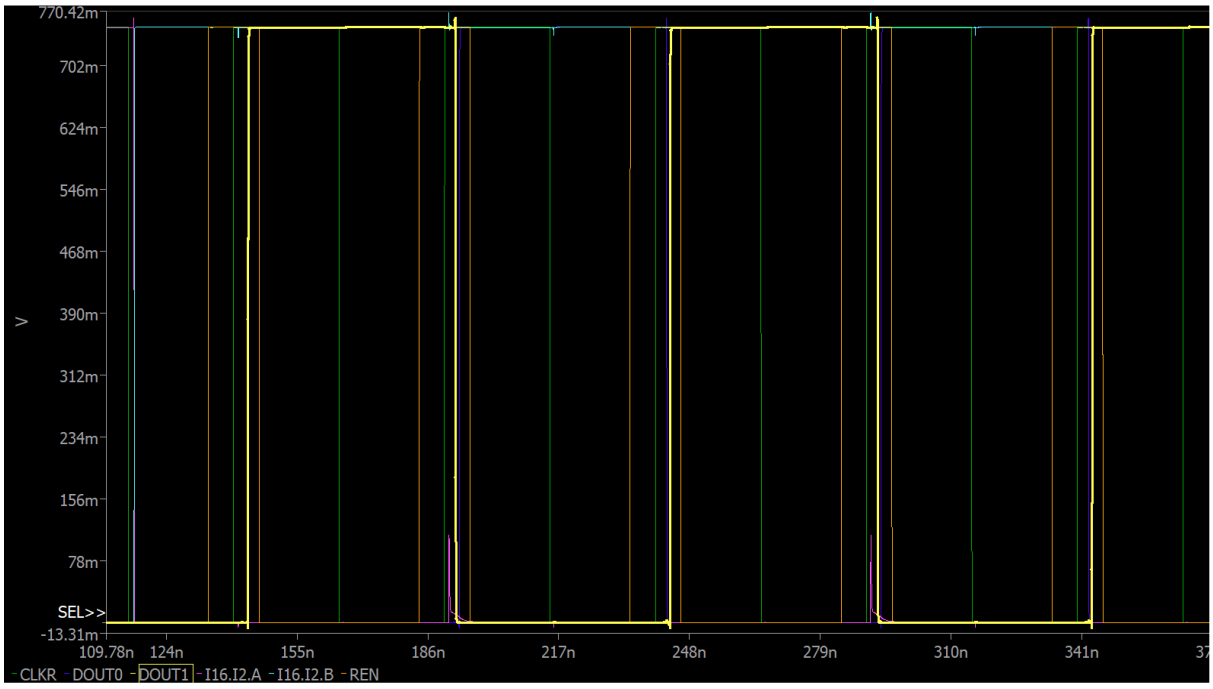


Рис. 2.29. Часова діаграма комірки пам'яті при W = 0.20 мкм, L=0.13 мкм, V_{DD} = 0.75 В.

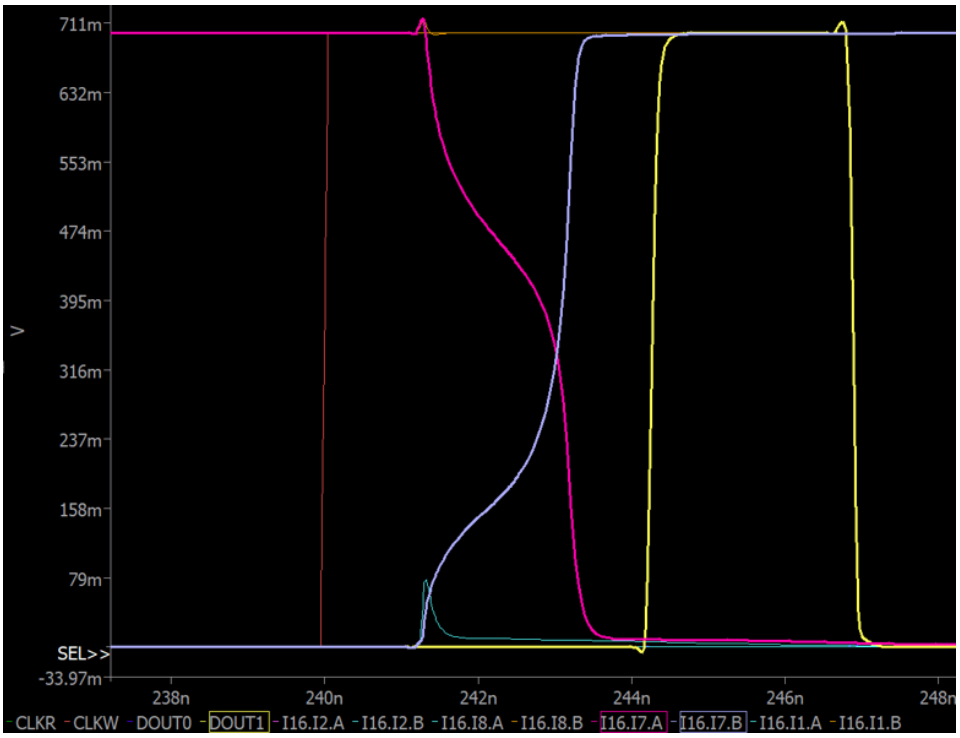


Рис. 2.30 Часова діаграма комірки пам'яті при W=0.20 мкм, L=0.13 мкм, V_{DD}=0.70 В

Результати моделювання на часових діаграмах (рис. 2.31–2.32) підтверджують працездатність оптимізованої комірки при мінімальній напрузі живлення V_{DD} = 0.60 В. При підвищенні напруги до 0.63 В комірка працює зі збільшеним запасом стійкості.

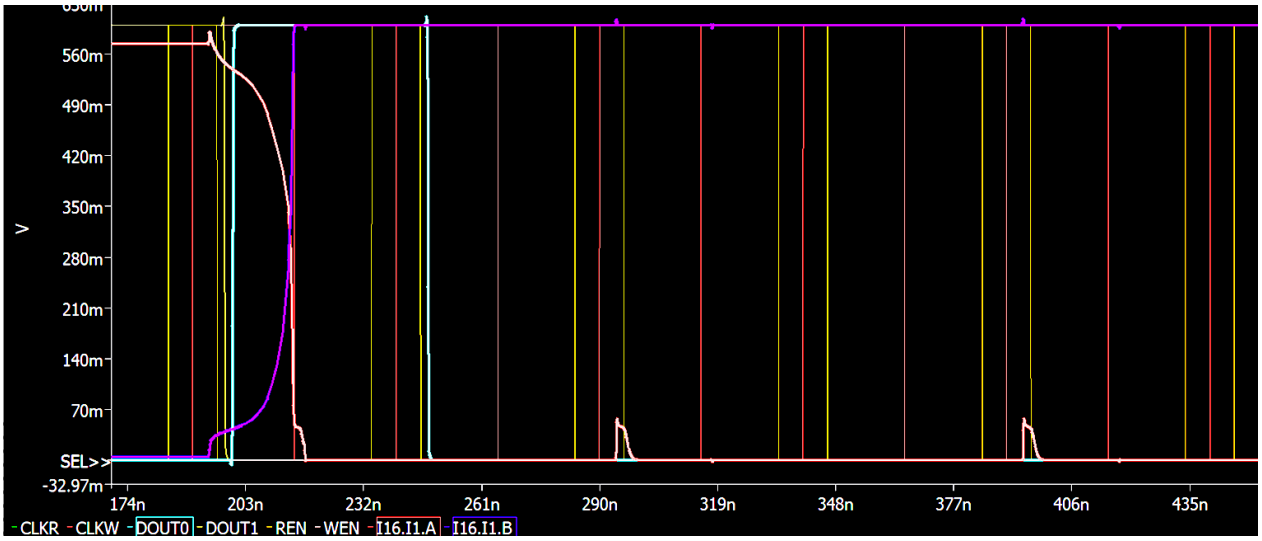


Рис. 2.31 Часова діаграма оптимізованої комірки при W=0.30 мкм, L=0.13 мкм V_{DD} = 0.60 В

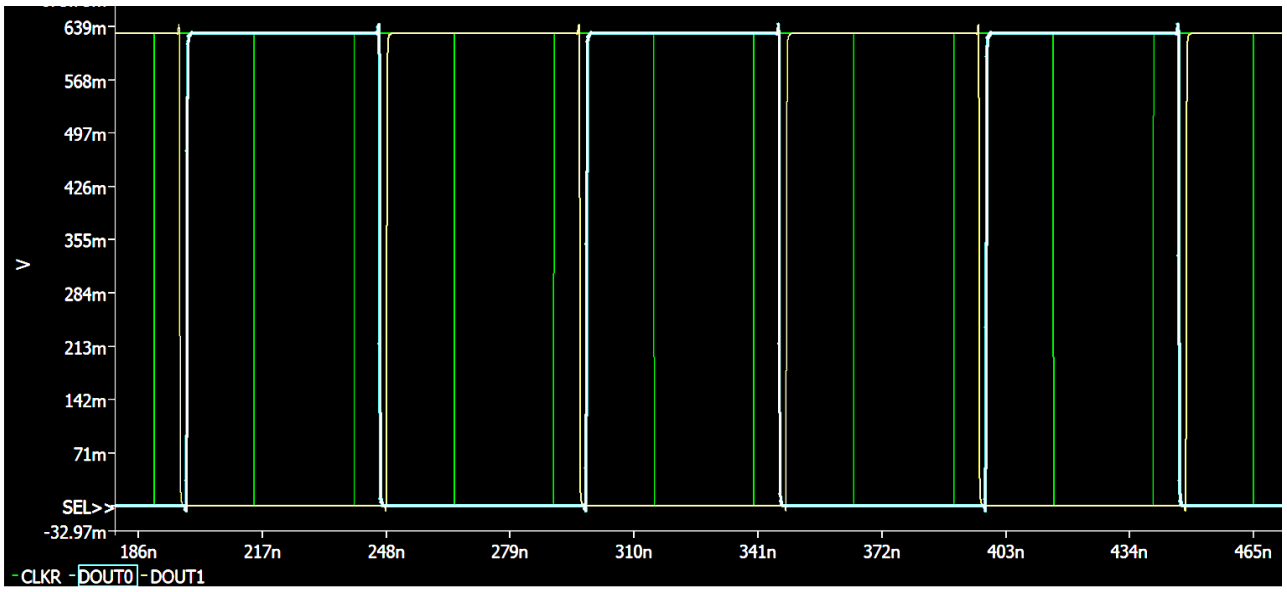


Рис. 2.32 Часова діаграма оптимізованої комірки при W=0.30 мкм, L=0.13 мкм, V_{DD} = 0.63 В.

Отже, запропонована оптимізація є раціональним компромісом між площею кристала та мінімальною напругою живлення [18, 19, 27, 28]. Без цілеспрямованої

оптимізації втрати площі, необхідні для досягнення аналогічного рівня завадостійкості, були б істотно більшими.

2.9. Експериментальний статистичний аналіз SNM-розподілу 8T dual-port SRAM-комірок

Основною задачею даного експерименту є точне визначення запасу статичної завадостійкості (SNM) комірок SRAM та регістрових файлів при роботі на знижених напругах живлення. Варіації порогових напруг транзисторів спричиняють статистичний розкид значень SNM, що може негативно впливати на стабільність зберігання та зчитування даних [4-10].

Постановка задачі на статистичний аналіз:

- експериментальне дослідження розподілу SNM у тестових структурах двопортових 8T SRAM-комірок, виготовлених за CMOS-технологією 250 нм;
- перевірка відповідності експериментального розподілу SNM нормальному (Гаусовому) закону з метою підтвердження коректності моделей, заснованих на статистиці невідповідності параметрів транзисторів [4, 5, 8, 9];
- порівняння експериментальних результатів із теоретичними розрахунками на основі моделі коефіцієнта площинної невідповідності A_{Vt0}
- для оцінки можливості її застосування до більш сучасних планарних технологічних процесів.

Результатом роботи має стати верифікована методика прогнозування стабільності комірок пам’яті, яка може бути використана при проєктуванні низькопотужних SRAM та регістрових файлів у субмікронних CMOS-технологіях.

2.9.1 Передумови статистичного аналізу SNM

Оскільки SNM є функцією порогових напруг транзисторів V_{tx}, V_{ty} та V_{tz} комірки пам'яті, а варіації цих напруг описуються нормальним законом розподілу, то за умови малих відхилень від номінального режиму та можливості лінеаризації функції SNM поблизу робочої точки її розподіл також можна наближено вважати нормальним. Таким чином, статистичні характеристики SNM можуть бути оцінені на основі моделей невідповідності параметрів транзисторів:

$$V_{tx} \sim N(\mu_x\ , \sigma_x^2) \quad \text{(n-канальний MOS транзистор інвертора комірки)}$$
$$V_{ty} \sim N\left(\mu_y\ , \sigma_y^2\right) \quad \text{(p-канальний MOS транзистор інвертора комірки)}$$
$$V_{tz} \sim N(\mu_z\ , \sigma_z^2) \quad \text{(n-канальний адресовий MOS транзистор)} \qquad (2.5)$$

$$Cell \sim N = (\mu_x + \mu_y + \mu_z\ , \sigma_x^2 + \sigma_y^2 + \sigma_z^2) \qquad (2.6)$$

Це припущення є ключовим для подальшого використання статистичних методів оцінки стійкості пам'яті [4-10].

2.9.2 Методика експериментального дослідження

Для перевірки гіпотези про нормальний характер розподілу SNM були проведені експериментальні дослідження двопортової 8T SRAM-комірки, розробленої для одного з європейських виробників мікросхем. Схему вимірювання комірки (SNM testbench) та її топологію, виготовлену за 250-нм CMOS-технологією, наведено на рис. 2.33–2.34.

Робота 6T та 8T SRAM-комірок ґрунтується на попередньому заряджанні бітових ліній до напруги живлення. Під час зчитування логічний стан визначається розрядженням однієї з бітових ліній, тоді як друга зберігає початковий потенціал. Це створює різницю напруг, необхідну для коректного визначення стану комірки [7, 11, 13, 21].

Оскільки в більшості архітектур SRAM-комірок до двох портів зчитування підключено дві бітові лінії, одна з яких відповідає логічній «1» (V_{DD}), а інша — логічному «0» (GND), під час операції зчитування одна з бітових ліній починає розряджатися, тоді як потенціал другої практично не змінюється.
У даному дослідженні основну увагу приділено початковому етапу операції зчитування, коли після завершення попереднього заряджання (precharge) обидві бітові лінії ще перебувають під напругою живлення V_{DD}. Саме в цей момент вплив статистичного розкиду порогових напруг транзисторів на величину запасу статичної завадостійкості (SNM) є максимальним, що робить цей режим найбільш критичним для оцінювання стійкості комірки пам'яті [4-10].

На рис. 2.33 показано тестову схему для визначення SNM на базі конвенційної 8T-комірки пам'яті двопортової SRAM, у якій для спрощення аналізу розімкнено зворотні зв’язки між інверторами комірки, що дозволяє отримати передавальні характеристики інверторів у формі Lissajous curves.

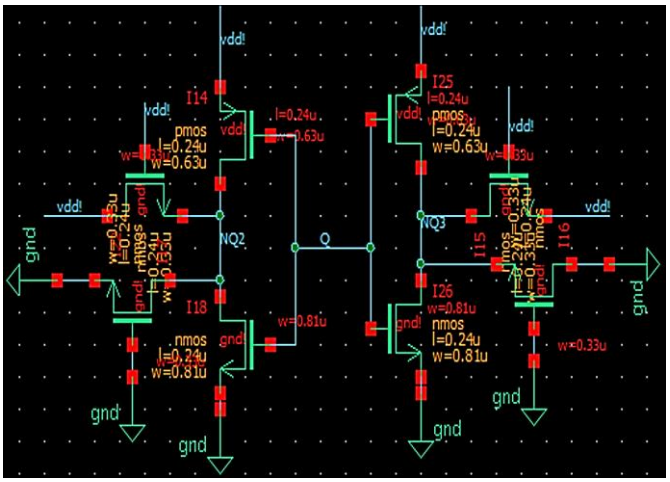


Рис. 2.33 – Тестова схема для визначення SNM 8T комірки пам'яті

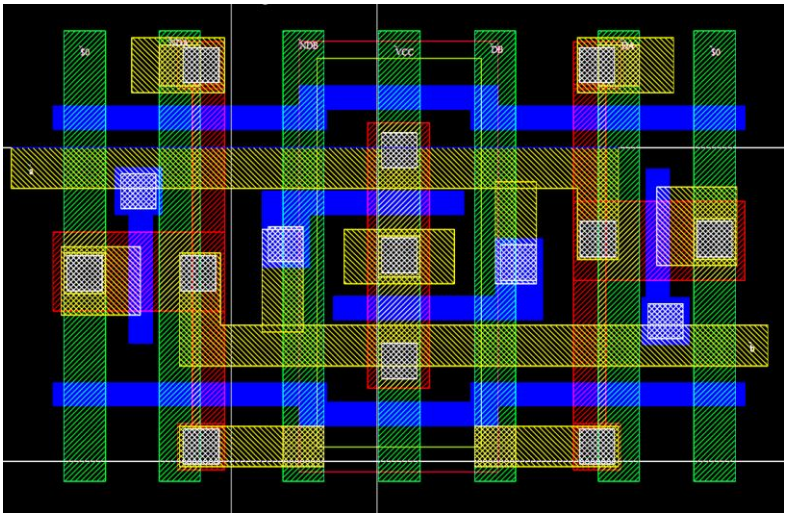


Рис. 2.34. Топологія 8T комірки двопортової SRAM 250nm

2.9.3 Розрахунок струмів і потужності споживання блока пам'яті

Розглянемо результати розрахунку струмів і потужності споживання блока пам'яті 20×42, згенерованого розробленим компілятором регістрових файлів [29]. Даний блок був виготовлений на фабриці та успішно протестований у кремнії, підтвердивши працездатність при мінімальній напрузі живлення 0.75 В.

Розрахунки виконувалися з використанням файлів екстрагованих нетлістів для моделювання spicenl.lib, сформованих засобами фізичної верифікації SmartDRC/LVS компанії Silvaco, для двох варіантів моделі:

- без урахування паразитних ємностей;
- з урахуванням паразитних ємностей, екстрагованих із топології [30, 31].

До екстракції були включені паразитні ємності, утворені шарами полікремнію, металізації та контактів. Екстракцію виконано для найгіршого технологічного випадку (worst-case), що відповідає мінімальній товщині міжшарових діелектриків і максимальній товщині провідних шарів. За таких умов паразитні ємності набувають найбільших значень.

Під час екстракції мінімальне значення паразитної ємності було обмежено величиною 1×10^{-16} Ф. Сформований spicenl.lib файл містить 10348 транзисторів та 8996 паразитних ємностей. Під час його генерації топологію було масштабовано з коефіцієнтом S = 0.84.

Розрахунок струмів споживання проводився на тактовій частоті 20 МГц для двох температурних режимів: –40°C (технологічний кут SS) та +125 °C (технологічний кут FS). Результати моделювання наведено в табл. 2.7 та табл. 2.8.

Таблиця 2.7

T= -40˚C, ss.

V_{DD}, В **I, мкА / P, мкВт**	**0.75**	**0.8**	**1.0**	**1.8**
I - без паразитних ємностей	45	48.6	54.8	113.8
I - з паразитними ємностями	66	76.2	97.1	170.5
P - без паразитних ємностей	31.5	38.9	54.8	204.8
P - з паразитними ємностями	46.2	61	97.1	306.9

Таблиця 2.8

T=+125˚C, fs

V_{DD}, В **I, мкА / P, мкВт**	**0.75**	**0.8**	**1.0**	**1.8**
I - без паразитних ємностей	52.3	63.4	80.3	158.1
I - з паразитними ємностями	78.1	90.3	119.3	233.5
P - без паразитних ємностей	31.5	38.9	54.8	204.8
P - з паразитними ємностями	46.2	61	97.1	306.9

Далі дослідимо вплив напруги попереднього заряду шин даних порту читання на струм споживання пам'яті. Для цього було виконано екстракцію нетліста для моделювання роботи регістрового файла з організацією 56×24 при технологічному куті SS, температурі 27°C, тактовій частоті 25 МГц та коефіцієнті масштабування 0.84.

Моделювання проводилося для двох варіантів топології, що відрізнялися рівнем попереднього заряду шин даних порту читання: V_{DD} та $V_{DD} - V_{th}$.

Таблиця 2.9

Вплив напруги попереднього заряду бітових ліній на струм і потужність споживання

Напруга попереднього заряду, В	V_{DD}, В Струм споживання, мкА, Споживана потужність мкВт	0.7	1.8
$(V_{DD} - V_{th})$	I	44.3	146.4
	P	31.01	263.52
V_{DD}	I	57.2	157
	P	40.04	282.6

Як видно з результатів, наведених у таблиці 2.9, використання оптимізованої комірки пам'яті разом зі зменшенням напруги попереднього заряду шин даних порту читання до рівня $V_{DD} - V_{th}$ забезпечує зменшення струму споживання та

споживаної потужності приблизно на 23 % при напрузі живлення V_{DD} = 0.7 В, що є суттєвим покращенням енергоефективності блока пам'яті [29].

2.9.4 Експериментальні результати

Було досліджено дві орієнтації комірки на кристалі (cell0 – із розміщенням паралельно X-осі та cell1 – перпендикулярно до неї), що дозволило врахувати можливий вплив топологічних особливостей. Результати показали, що більшість комірок, а саме 98 з 100 для Cell0 та 97 зі 100 для Cell1 є працездатними, при цьому незначна кількість дефектів спостерігається в окремих позиціях на пластині. На рис. 2.36–2.39 наведено статистичні розподіли експериментально виміряних значень SNM для комірок Cell0 і Cell1, а також для відповідних вузлів зберігання даних A та B.

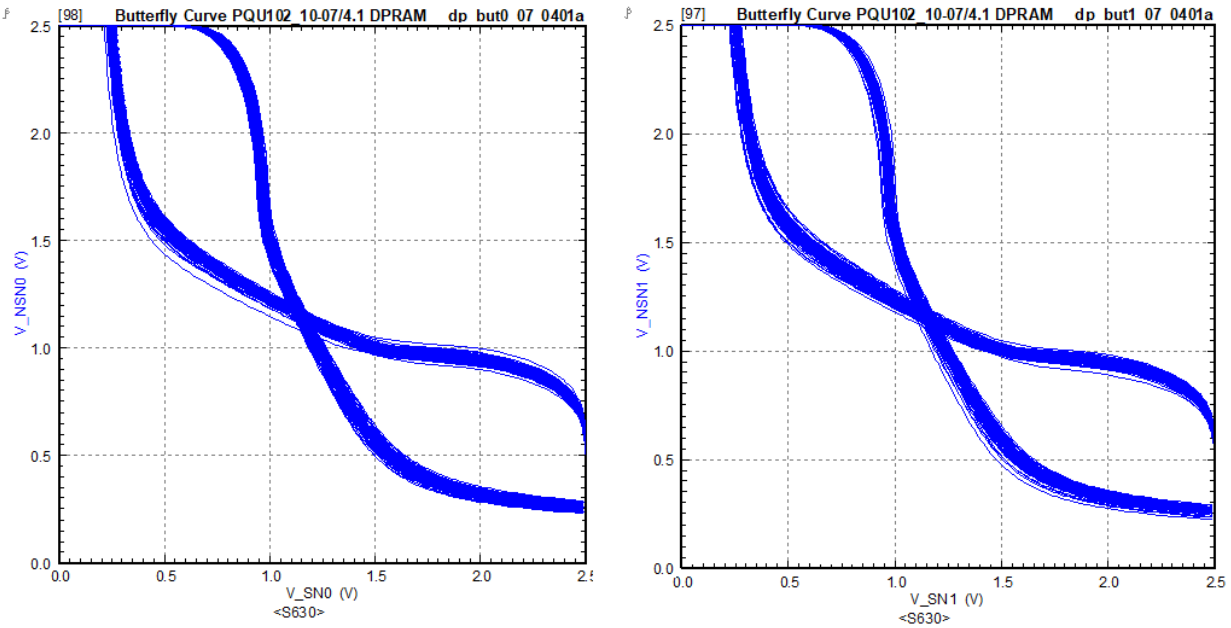


Рис. 2.35 – Butterfly-криві для cell0 та cell1

Вимірювання SNM виконувалися із застосуванням тестера серії Agilent 93000 SOC, а отримані дані оброблялися спеціалізованим програмним забезпеченням, яке оцінювало відповідність нормальному розподілу на основі коефіцієнтів

асиметрії (As) та ексцесу (Ex), а також за допомогою критерію Шапіро–Уїлка. Розподіли виміряних значень SNM наведені на рис. 2.36 - 2.39:

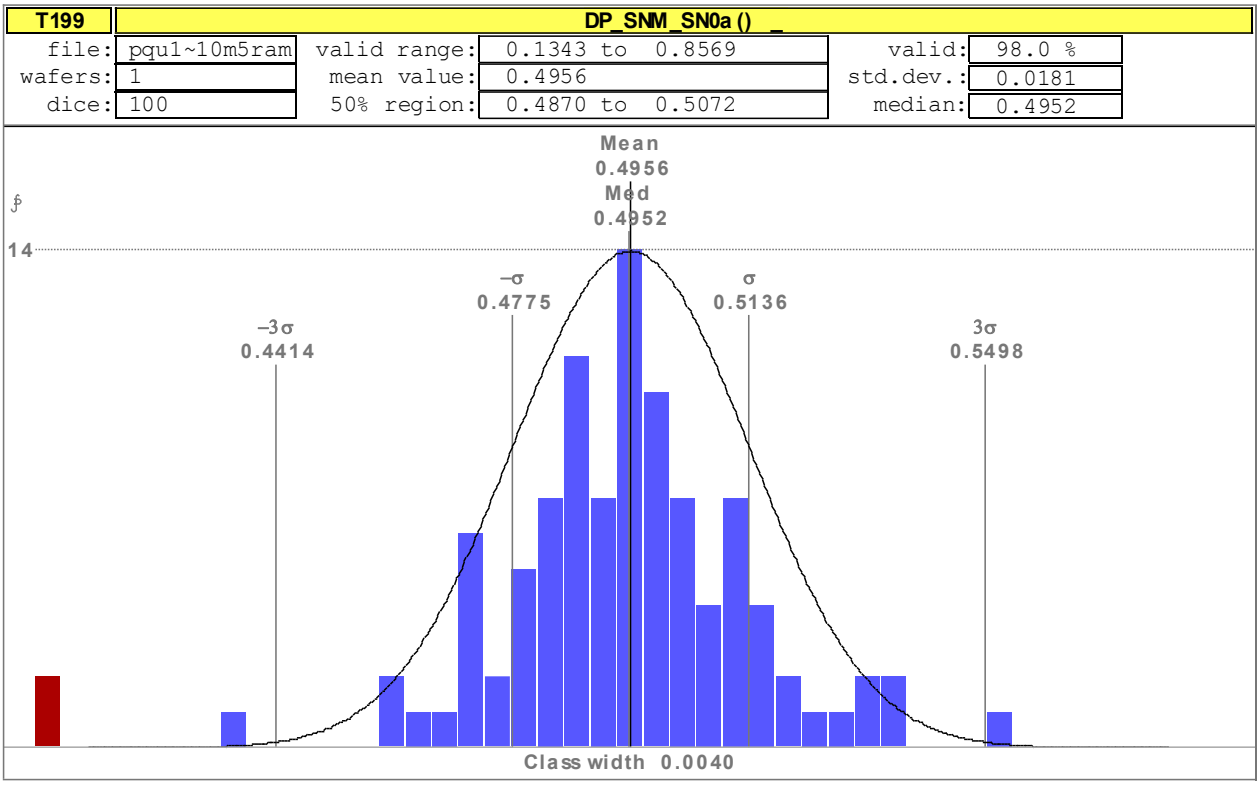


Рис. 2.36 – Статистичний розподіл SNM для cell0: а) **SN0a**

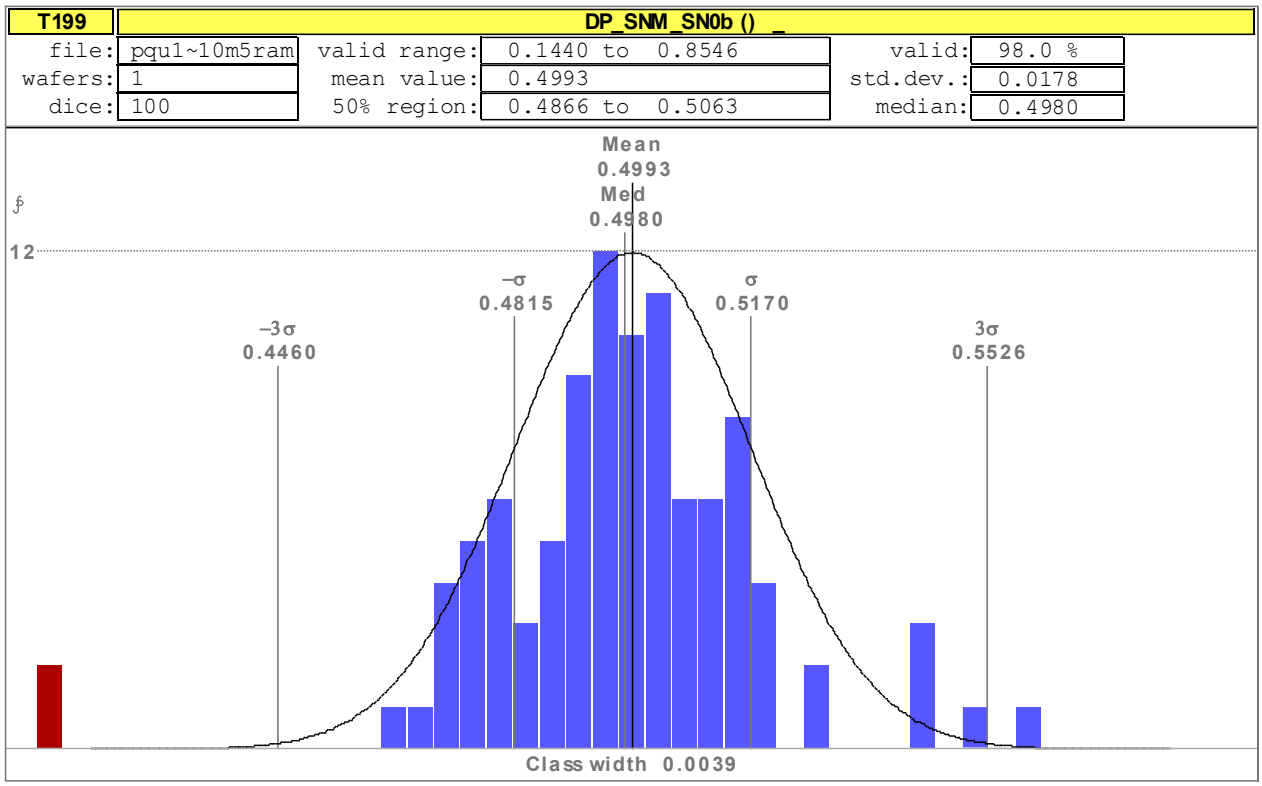


Рис. 2.37 - Статистичний розподіл SNM для cell0: b) **SN0b**

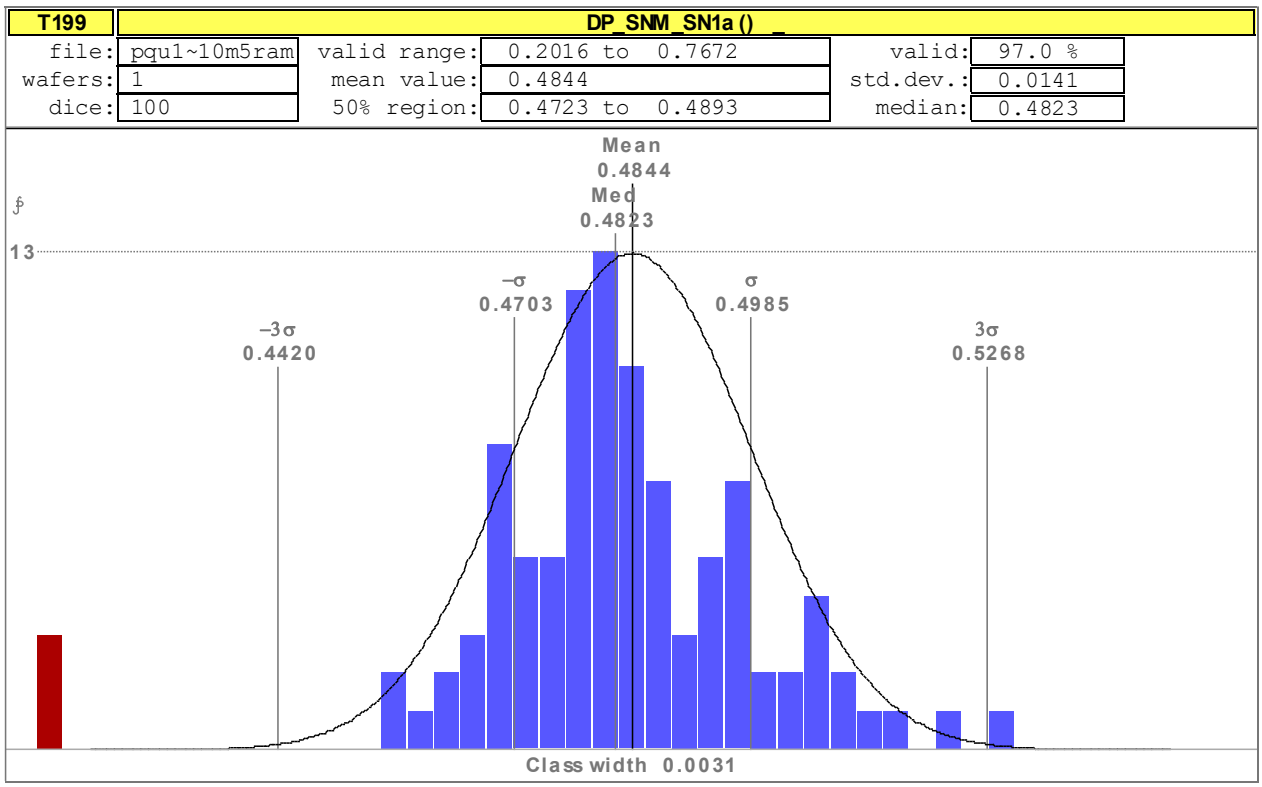


Рис. 2.38 – Статистичний розподіл SNM для cell1: a) **SN1a**

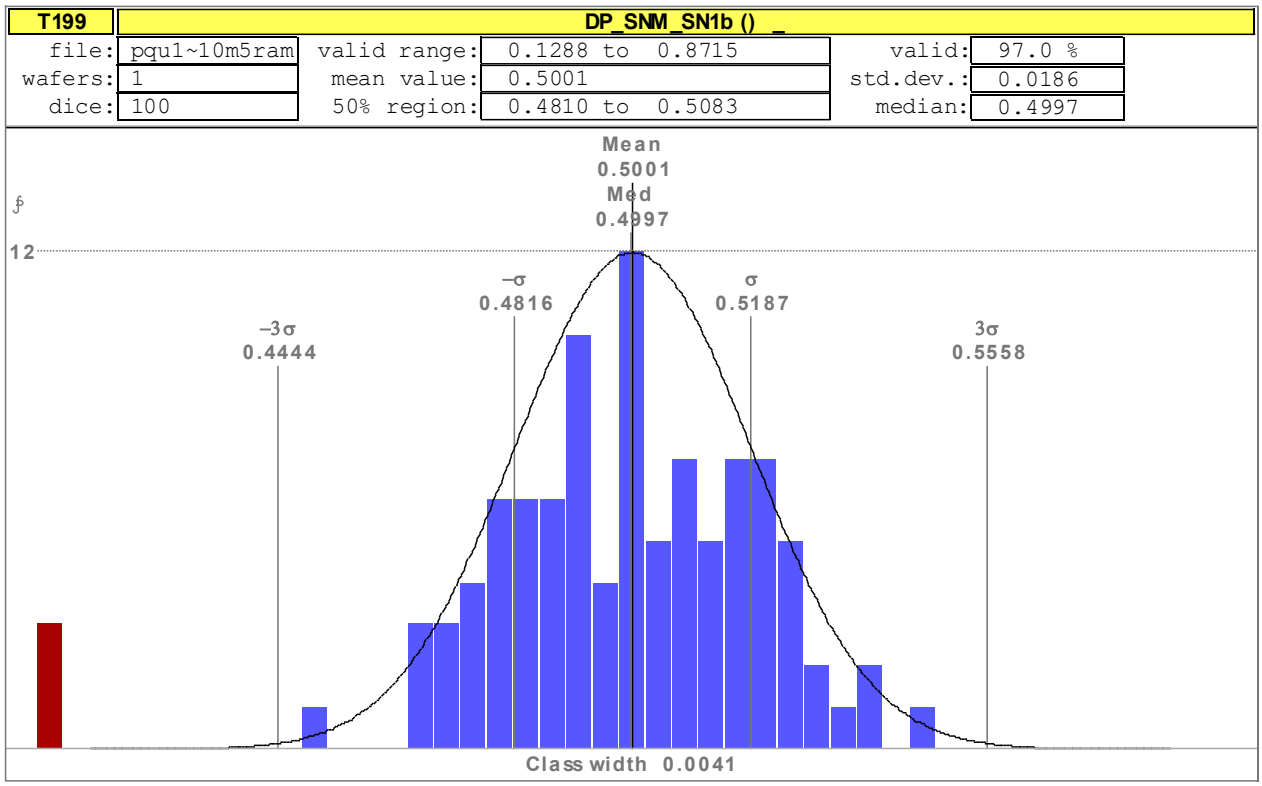


Рис. 2.39 – Статистичний розподіл SNM для cell1: b) **SN1b**

Дефектні комірки на гістограмах позначені червоним кольором. Слід зазначити, що вибірка обсягом 100 вимірювань є недостатньою для достовірної оцінки поведінки розподілу в області ±3σ, тому подальший аналіз було обмежено перевіркою кількості значень, що потрапляють у діапазон ±σ відносно середнього значення.

Для оцінки відповідності експериментальних даних нормальному закону розподілу було проаналізовано частку значень у межах ±σ, а також виконано порівняння середнього арифметичного та медіани вибірок [4, 5, 8, 9]. Підсумкові результати вимірів наведені у табл. 2.10.

Таблиця 2.10

Аналіз експериментальних даних у межах стандартного відхилення

Назва зразка чипу пам'яті	DP_SNM_SN 0a	DP_SNM_SN 1a	DP_SNM_SN 0b	DP_SNM_SN 1b

Середньоквадратичне відхилення (σ), В.	0.0181	0.0141	0.0178	0.0186
Кількість інформативних точок у межах діапазону ±σ (стандарт 0.6827)	0.7346	0.6907	0.7142	0.639
Середнє значення SNM (Mean), В	0.4956	0.4844	0.4993	0.5001
Медіана (Med), В	0.4952	0.4823	0.4980	0.4997
Різниця (Mean – Med), В	0.0004	0.0023	0.0013	0.0004

Як видно з отриманих результатів, медіанні значення лише незначно відрізняються від середніх, за винятком вибірки **_SN1a.** Таке відхилення можна пояснити недостатнім обсягом вибірки, оскільки для більш достовірної статистичної оцінки кількість вимірювань доцільно збільшити щонайменше на порядок. Для інших вибірок значення середньоквадратичного відхилення практично збігаються, а середнє значення **σ** становить **0.0182.**

Отримані результати не суперечать гіпотезі про нормальний характер розподілу SNM та підтверджують доцільність використання mismatch-based підходу для інженерного прогнозування стабільності 8T-комірок пам’яті [4-10]. Це, у свою чергу, свідчить про те, що варіації параметрів окремих транзисторів комірки пам’яті також можуть бути описані нормальним законом розподілу.

Результати узгоджуються з прийнятим припущенням про статистичний розкид порогових напруг. Також було встановлено, що різниця між середнім значенням та медіаною є незначною, що додатково підтверджує симетричність розподілу.

2.9.5 Теоретичний розрахунок стандартного відхилення

Для перевірки експериментальних результатів було виконано теоретичний розрахунок стандартного відхилення SNM на основі моделі розузгодження параметрів транзисторів [4, 5, 8, 9]. Зв’язок між стандартним відхиленням

порогової напруги та геометричними параметрами транзистора описується залежністю:

$$\sigma_{V_t} = \frac{A_{V_t}}{\sqrt{2 \cdot W \cdot L}} \tag{2.7}$$

де:

- A_{V_t}— коефіцієнт розузгодження,
- W, L— ширина та довжина транзистора.

Значення коефіцієнта невідповідності A_{V_t} (коефіцієнта Пелгрома) розраховано за моделлю, наведеною в роботі [5], для 250-нм CMOS-технології з товщиною підзатворного оксиду 5.7 нм. Відповідно до технологічної специфікації процесу значення A_{V_t} становить 5.37 мВ·мкм для NMOS-транзисторів та 6.89 мВ·мкм для PMOS-транзисторів.

На основі параметрів транзисторів комірки (табл. 2.11) було визначено стандартні відхилення для окремих гілок.

Таблиця 2.11

Розміри та значення стандартних відхилень транзисторів комірки пам'яті

Тип MOS	$NMOS_{down}$	$NMOS_{address}$	$PMOS_{up}$
W, мкм	0.81	0.33	0.63
L, мкм	0.24	0.24	0.24
σ, В	0.00861	0.01349	0.01255

Для інженерного зіставлення використано спрощену оцінку сумарного розкиду, у якій внески вважаються незалежними та підсумовуються квадратично. Така

оцінка не замінює повного статистичного моделювання чутливості SNM до параметрів усіх транзисторів:

$$\sigma_{cell} = \sqrt{\sigma_{n_down}^2 + \sigma_{n_addr}^2 + \sigma_{p_up}^2} \quad (2.8)$$

Підставляємо значення:

$$\sigma_{cell} = \sqrt{(0.00861)^2 + (0.01349)^2 + (0.01255)^2} \approx 0.0204 \text{ В} \quad (2.9)$$

Отже, 0.0204 В — це теоретичне стандартне відхилення SNM для гілки 8T-комірки, отримане як корінь із суми квадратів стандартних відхилень трьох транзисторів [4, 5, 8, 9].

Експериментальне значення (береться конкретно для вибірки **DP_SNM_SN0a)**:

$$\sigma_{SNM}^{exp} \approx 0.0181 \text{ В} \quad (2.10)$$

Розбіжність між теоретичним та експериментальним значенням становить близько 11 %, що є допустимим та підтверджує адекватність використаної моделі. Це дозволяє для комірок пам’яті типу регістрового файлу, реалізованих на базі енергоефективних 8T-комірок SRAM, у яких зниження енергоспоживання досягається за рахунок зменшення напруги живлення, для технологічних норм 180 нм, 152 нм (отриманих шляхом масштабування 180-нм технології) та 130 нм використовувати розрахункові значення коефіцієнта розузгодження площі A_{vto}.

Висновки до другого розділу

У результаті проведених досліджень:

1. Зниження V_{DD} є ефективним методом зниження енергоспоживання RFSRAM, але призводить до деградації SNM/WNM і підвищення чутливості до розкиду параметрів MOS-транзисторів.
2. Для low-voltage RFSRAM доцільним є використання 8T-комірки з розділеними портами читання і запису.
3. Номінальне значення SNM без урахування статистичного розкиду V_T є недостатнім критерієм для оцінювання стабільності пам'яті.
4. Запропоновано використовувати 6σ-модель розкиду порогових напруг та butterfly-критерій граничного стану для визначення допустимих розмірів транзисторів.
5. Встановлено, що найбільший вплив на стабільність у режимі читання мають n-канальні транзистори інверторів комірки та транзистори порту читання.
6. Експериментальний аналіз SNM-розподілу тестових 8T dual-port SRAM-комірок підтвердив доцільність використання нормального закону розподілу та mismatch-based підходу для прогнозування стабільності low-power пам'яті.
7. Отримані результати створюють основу для інтеграції SNM/WNM-орієнтованої методики вибору параметрів 8T-комірки у параметричний RFSRAM memory compiler.

РОЗДІЛ 3. Архітектура та алгоритмічне забезпечення компілятора двопортової RFSRAM-пам’яті з урахуванням post-layout SPICE-характеризації

3.1. Конфігурація та функціональні можливості компілятора RFSRAM

Це високопродуктивний двопортовий синхронний компілятор реалізацій пам’яті регістрового файла RF, спроєктований за технологією 0.18 мкм із підтримкою коефіцієнта технологічного масштабування 0.84. Кожна згенерована реалізація пам’яті оптимізується для напруги живлення в діапазоні 0.9–1.1 В, але також підтримує роботу при напругах до 1.98 В у режимі підвищеної швидкодії та при 0.7 В у режимі зниженого енергоспоживання [1, 2, 27, 28].

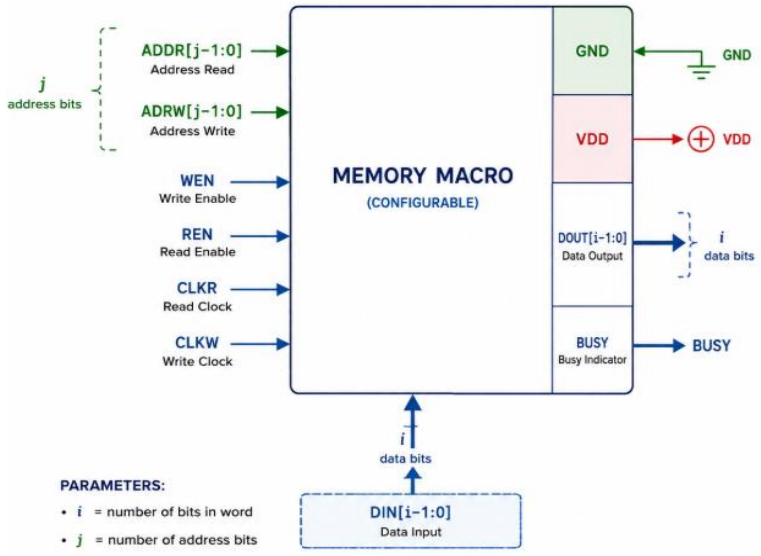


Рис. 3.1. Блок-схема двопортового макросу RFSRAM 1R1W з описом вхідних та вихідних сигналів.

Сумісні технологічні процеси: **TSMC 0.152 мкм CMOS Logic General Purpose 1P6M 1.8 В / 3.3 В**

- Тип пам’яті: двопортовий регістровий файл **1R1W**

- Топологія регістрового файлу використовує 3 шари металізації (**M1–M3**)
- Обрані пріоритети оптимізації: низьке енергоспоживання та площа
- Доступний діапазон конфігурації кількості слів: 16-256, із кроком 4.
- Доступний діапазон конфігурації довжини слова: 8 -72, із кроком 1 біт
- Два незалежні тактові сигнали CLK для читання та запису
- Роздільні вхідні та вихідні порти
- Максимальна частота: 200 МГц
- Температурний діапазон: -40°C +125°C
- Полярність тактового сигналу — активація операцій читання та запису за спадним фронтом.
- При одночасному читанні та записі за однією і тією самою адресою завжди зчитуються нові дані (функція **write-through**)
- Додаткові опції: генерація вбудованого блоку самотестування (BIST) із автоматичним підключенням та розміщенням на кристалі пам'яті [1, 2].

До вхідних сигналів належать:

ADRW[7:0] — входи адреси пам’яті.

Цей адресний порт використовується для вибору комірки пам’яті, у яку здійснюється запис під час циклу запису.

ADRR[7:0] — входи адреси пам’яті.

Цей адресний порт використовується для вибору комірки пам’яті, з якої здійснюється читання під час циклу читання.

DIN[39:0] — порти вхідних даних. Це вхідна шина даних, яка використовується для запису даних у комірку пам’яті, визначену адресним вхідним портом під час циклу запису.

WEN — зовнішній службовий сигнал, що дозволяє операцію запису у порт запису (**Write Strobe Enable**).

Коли сигнал дозволу запису має логічний високий рівень, пам’ять перебуває в режимі циклу запису.

REN — зовнішній службовий сигнал, що дозволяє операцію читання по порту читання (**Read Strobe Enable**).

Коли сигнал дозволу читання має логічний високий рівень, пам’ять перебуває в режимі циклу читання.

CLKR — тактовий сигнал для операції читання. Це зовнішній тактовий сигнал пам’яті для режиму читання.

CLKW — тактовий вхід для операції запису. Це зовнішній тактовий сигнал пам’яті для режиму запису.

До вихідних сигналів належать:

DOUT[39:0] — вихідна шина даних.

До сигналів живлення належать:

GND — Низький рівень напруги живлення.

VDD — Високий рівень напруги живлення.

3.2. Архітектура макросу пам’яті, згенерованої компілятором RFSRAM

Двопортова пам’ять типу Register File (RF) керована одним сигналом CLK є SRAM-пам’яттю, побудованою на основі 8-транзисторної комірки та має два порти: один порт для запису даних у пам’ять і один порт для читання даних із пам’яті [20–22, 27, 28]. Кожен порт має власний дешифратор рядків, а також окремі адресні порти слова для порту читання та порту запису. RF SRAM активується за спадним фронтом сигналу CLK. Архітектура RF верхнього рівня показана на Рис. 3.2:

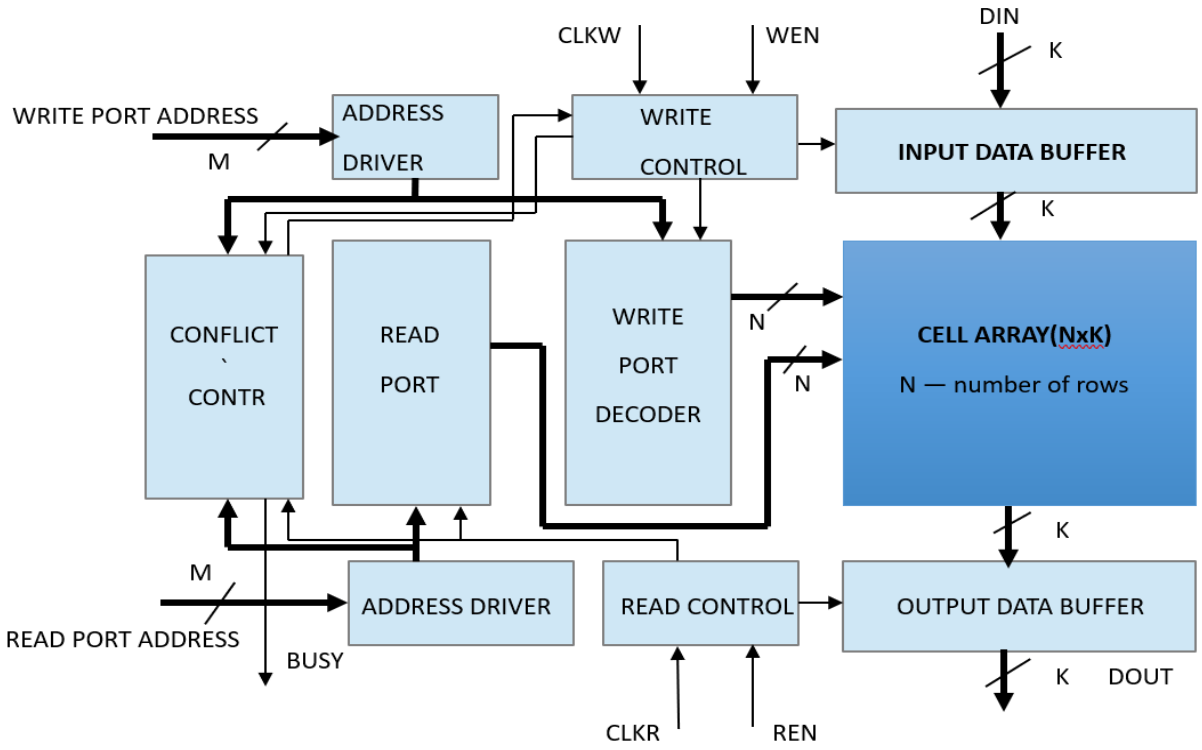


Рис. 3.2. Узагальнена структурна схема двопортового макросу RFSRAM 1R1W, що генерується компілятором.

Взаємне розташування базових блоків на блок-схемі відповідає їхньому геометричному розміщенню в дизайні топології. Приклад такого розміщення наведено на Рис. 3.3 на прикладі реалізації RF1 20x42 [29].

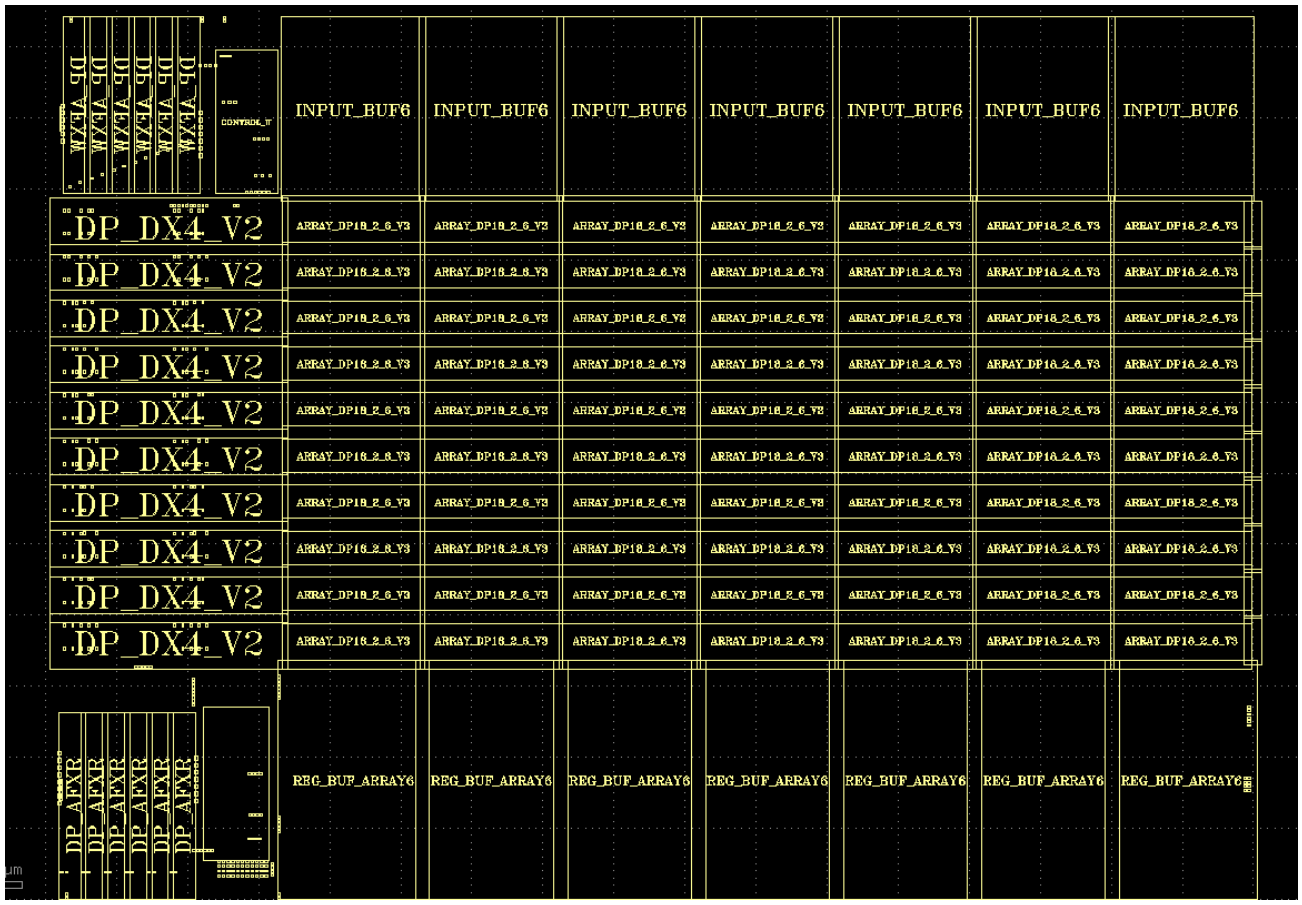


Рис. 3.3. Схема топологічного компонування базових блоків RFSRAM на прикладі реалізації із конфігурацією 20x42.

Розташування комірок у матриці RF відносно сигналів ADDR та DIN показано на Рис. 3.4:

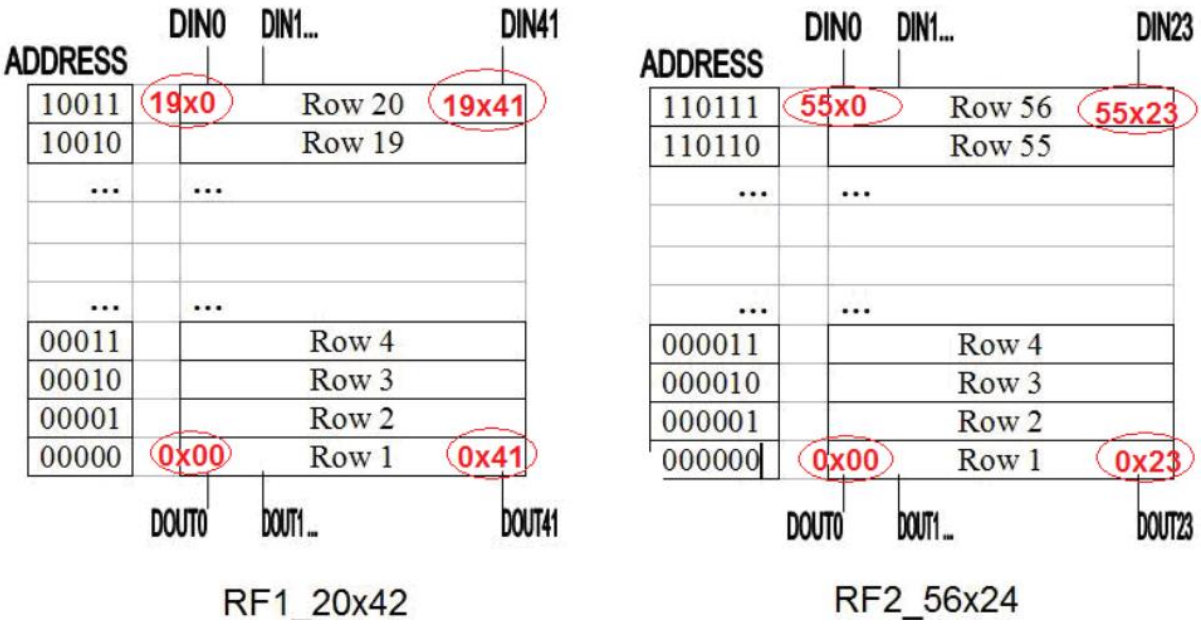


Рис. 3.4. Мапа розташування комірок у матриці RFSRAM для обраних реалізацій.

3.3. Формування топологічної структури макросу пам’яті

3.3.1 Збірка матриці

Базовим елементом матриці є блок LEAF_CELL_ARRAY — bit-cell array, що містить по 2 рядки адрес портів запису та читання і 8 пар стовпців даних [1, 2, 27, 28]. Під час складання адресні входи порту читання ARXi необхідно з’єднати між собою по рядку та розмножити по рядках — усього row рядків. Аналогічно виконати з’єднання для адресних входів порту запису AWXi.

Стовпці порту читання позначаються як DA, NDA, а стовпці порту запису — DB, NDB. Кількість блоків по рядках становить row/2. Кількість блоків по стовпцях становить col/8, що відповідає довжині слова word_width і загальній кількості col стовпців.

3.3.2. Збірка вихідних блоків даних порту читання

LEAF_REG_BUF — це ім’я блоку. Кількість таких блоків дорівнює довжині слова, або половині кількості стовпців col/2. Блок підключається до двох стовпців

і містить комутатор із входами YR0, YR1. Входи блоків DAi, NDINi підключаються до відповідних виходів матриці DAi, NDAi. Виходи блоків DOUTi є зовнішніми портами макросу пам'яті. Сигнали PZP і RENI, спільні для всіх блоків LEAF_REG_BUF, необхідно об'єднати та надалі з'єднати з відповідними виходами блока CONTROL_R.

3.3.3. Збірка вхідних блоків даних порту запису

LEAF_INPUT_BUF — це ім'я блоку. Кількість таких блоків дорівнює половині кількості стовпців col/2. Блок містить комутатор із входами Y0, Y1. Зовнішні порти блока — DINi є також зовнішніми для всього макросу пам'яті. Виходи Di, DNi підключаються до портів запису матриці DBi, NDBi. Входи CLKWH усіх блоків необхідно об'єднати та з'єднати з виходом CLKWH блока CONTROL_W.

3.3.4. Збірка дешифратора рядків

LEAF_DXDP — це ім'я блоку. Він реалізований із розрахунку 1 блок на 2 рядки.

Кількість дешифраторів визначається як: $row/2$. Сигнали R1 і W1 уже підключені всередині блока. Сигнали R2 і W2 підключаються виходами адресних формувачів:

- ADRR_OUT2
- ADRRN_OUT2
- ADRW_OUT2
- ADRWN_OUT2

починаючи з ADRRN_OUT2 та ADRWN_OUT2, тобто у послідовності: 11001100. Розряди R3–R7 та W3–W7 підключаються по рядках однаково та послідовно від: 00000 до: 11111.

3.3.5. Збірка адресних формувачів портів читання та запису

LEAF_ADFPR, LEAF_ADFPW — це топологічне ім’я блоків читання та запису відповідно.

Формувач ADRW0 входить до складу блока LEAF_CONTROL_W.

Вхідні порти:

- ADRR0–ADRRj та ADRW1–ADRWj, де j – кількість адресних біт. Наприклад для макросу RF2 56x24 j = 6, тому адреси перебираються від 000000(0x00) до 110111 (55x0), а для макросу RF1 20x42 j = 5, адреси перебираються відповідно від 000000(0x00) до 10011 (19x0).

Виходи цих комірок використовуються для прошивки дешифраторів:

- ADRR_OUTi, ADRRN_OUTi — прямий та інверсний сигнали порту читання при $i = 1 \ldots 7$;
- ADRW_OUTi, ADRWN_OUTi — прямий та інверсний сигнали порту запису при $i = 1 \ldots 7$.

При $i = 0$, тобто для ADRR0, виходи YR0, YR1 необхідно з’єднати з однойменними входами блока LEAF_REG_BUF. Вхід CLKRH блока LEAF_ADFPR необхідно з’єднати з однойменним виходом блока LEAF_CONTROL_R. Вхід CLKWX блока LEAF_ADFPW необхідно з’єднати з однойменним виходом блока LEAF_CONTROL_W.

3.3.6. Підключення блоку керування портом читання

LEAF_CONTROL_R — топологічне ім’я блоку.

Зовнішні вхідні порти та макросу пам’яті: REN, CLKR. Внутрішні вихідні сигнали блоку: CLKRH, RENI, PZP

3.3.7. Підключення блоку керування портом запису

LEAF_CONTROL_W — топологічне ім’я блоку.

Зовнішні вхідні порти блока та макросу пам’яті : ADRW0, CLKW, WEN. Внутрішні вихідні сигнали блоку: CLKWH, CLKWX, Y0, Y1.

3.4. SPICE-модель макросу пам’яті та стан моделювання SRAM_MODEL

3.4.1 Параметри моделі

Schematic-based модель **SRAM_MODEL** призначена для моделювання роботи RF у режимах читання та запису. Паразитні ємності та опори моделі були екстраговані з топології [30, 31]. У цьому поданні макрос описується еквівалентною електричною схемою з активними транзисторними елементами та пасивними RC-компонентами; схемотехнічне моделювання дає змогу визначати напруги вузлів, струми гілок і часові характеристики пам’яті. Модель складається з таких елементів:

1. RF_cell_array — масив комірок пам’яті, що складається з чотирьох комірок, розміщених у різних кутах масиву. Живлення — vdd!
   XA, XAN — адреси рядка порту читання; XB, XBN — адреси рядка порту запису.
   DA0, DA1, NDA0, NDA1 — лінії даних порту читання.
   DB0, DB1, NDB0, NDB1 — лінії даних порту запису.

Усередині RF_cell_array представлені такі паразитні елементи:

R0, R2 — паразитні опори адресних ліній рядка порту читання;

R1, R3 — паразитні опори адресних ліній рядка порту запису;

C0–C6 — паразитні ємності між лініями даних;

C8–C11 — паразитні ємності ліній даних відносно землі;

I2, I9 — сумарні адресні транзистори рядка порту читання з шириною, пропорційною кількості стовпців;

I6, I26 — сумарні адресні транзистори рядка порту запису з шириною,

пропорційною кількості стовпців;

I8, I11, I13, I15, I17, I21, I23, I25 — транзистори для моделювання сумарних p/n-переходів ліній даних та ємності Міллера, ширина яких пропорційна кількості рядків; R4–R11 — паразитні опори ліній даних [30, 31].

2. ADFP — драйвери адрес портів читання та запису. Їхнє навантаження у вигляді паразитних ємностей (C4, C5), транзисторів I26, I28 та паразитних опорів (R3, R4) залежить від кількості рядків. Живлення — vdd2!
3. DXDP – блок дешифраторів рядків портів читання та запису. Живлення -vdd3!
4. CONTROL_R — блок керування портом читання. Живлення — vdd6!
5. CONTROL_W — блок керування портом запису. Живлення — vdd5! Він генерує внутрішній сигнал CLKWH при WEN = high, який надходить до адресного драйвера порту запису, дешифратора рядків та вхідних буферів.
6. REG_BUF — підсилювач зчитування та вихідний буфер. Живлення — vdd1!, vdd7! для підсилювача зчитування.
7. INPUT_BUF — вхідний буфер. Живлення — vdd4!
8. Окремі джерела живлення моделі використовуються для вимірювання струмів споживання різних схем, а також для моделювання впливу паразитних опорів різних ліній живлення [29].
9. Вхідні сигнали:

Назва джерела напруги	**Призначення сигналу**
V16	CLK (тактуючий сигнал)
V4	REN (дозвіл читання)
V5	WEN (дозвіл запису)
V3	ADRR (адреса порту читання)
V8	ADRW (адреса порту запису)
V7, V10	DIN0, DIN1 (вхідні дані)

10. Опис стану моделювання. Назва стану — **SRAM_MODEL_152G**:

Назва параметру	Значення параметру
col, row	кількість стовпців, кількість рядків
bit	розрядність слова
tp	період тактового сигналу
vp	напруга живлення
tr, tf	фронт наростання та спадання адресних сигналів
trc, tfc	фронт наростання та спадання сигналу CLK
trw, tfw	фронт наростання та спадання сигналів REN і WEN;
trd, tfd	фронт наростання та спадання сигналу DIN
tsu, tho	часи встановлення та утримання адреси
tsud, thod	часи встановлення та утримання вхідних даних
tsuw, thow	часи встановлення та утримання сигналів REN/WEN

3.4.2 Контрольний список моделювання

Часове моделювання було виконано для таких режимів:

1. Запис у всю пам’ять (WEN = high, REN = 0) тестового шаблону checkerboard0 та подальше читання всієї пам’яті (WEN = 0, REN = high) для перевірки записаних даних.
2. Одночасний запис і читання за однаковими адресами (ADRR = ADRW, WEN = high, REN = high) для таких тестових шаблонів:
    - solid1;
    - solid0;
    - checkerboard0.
3. Одночасний запис і читання за різними адресами (WEN = high, REN = high) для тестового шаблону checkerboard0.

Часове моделювання для вимірювання часу доступу та струму споживання виконувалося для такого тесту: одночасний запис і читання за однаковими адресами (ADRR = ADRW, WEN = high, REN = high) з тестовим шаблоном checkerboard0, при навантажувальній ємності $CL_{min} = 0$, $CL_{max} = 0.5$ пФ та фронтах

наростання/спадання адресних сигналів tr, tf у межах: мінімум 0.1 нс, максимум 2 нс. Для цих умов також вимірювалися затримки внутрішніх сигналів, необхідні для розрахунку часів встановлення та утримання [29, 31].

3.4.3 Результати моделювання часу доступу

Моделювання часу доступу T_{acc} виконувалося на основі екстрагованих нетлістів із урахуванням паразитних ємностей, наданих замовником та отриманих за допомогою EDA інструмента Silvaco SmartDRC/LVS [30, 31].

Моделювання проводилося для всіх крайових умов процесу з двома варіантами навантаження виходів:

- частина виходів без навантаження: *CL* = 0 пФ;
- частина виходів із навантаженням: *CL* = 0.5 пФ.

Для аналізу було обрано чотири комірки, розташовані у нижньому та верхньому лівому/правому кутах масиву. Відповідні вихідні сигнали: DOUT0 для лівого кута та DOUT23 для RF2 56×24; DOUT41 для RF1 20×42 для правого кута.

Адреси портів читання/запису для цього моделювання задавалися таким чином:

Реалізація пам'яті: RF1 20x42: 00000 → 10011 → 00000 …

Реалізація пам'яті: RF2 56x24: 000000 → 110111 → 000000 …

Результати моделювання часу доступу для всіх крайових умов процесу для обох реалізацій пам'ятей RF1 та RF2 наведені в табл. 3.1 для випадку ненавантажених виходів ($CL = 0$).

Найгірші значення часу доступу для ненавантажених виходів становили: T_{acc} = 20.21 нс для пам'яті RF2 56x24, та T_{acc} = 19.42 нс для пам'яті RF1 20x42 при таких умовах: V_{DD} = 0.8 В, ss , -40°C. Але за результатами моделювання зрозуміло, що такі T_{acc} більше ніж достатньо для задоволення вимог замовника відповідно до специфікації, що були встановлені для таких умов на рівні 10 МГц.

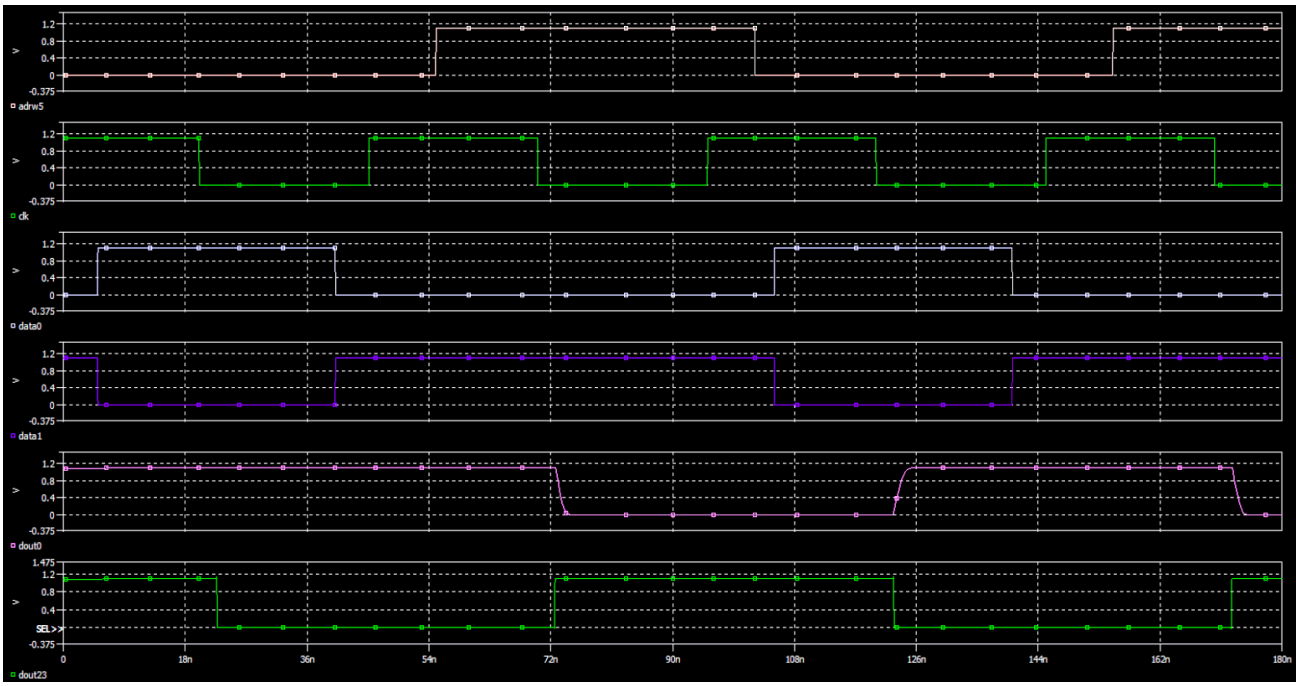


Рис. 3.5 – Часова діаграма виконання тесту checkerboard1 для RF2 56x24 (1.0 В, tt, 25°C)

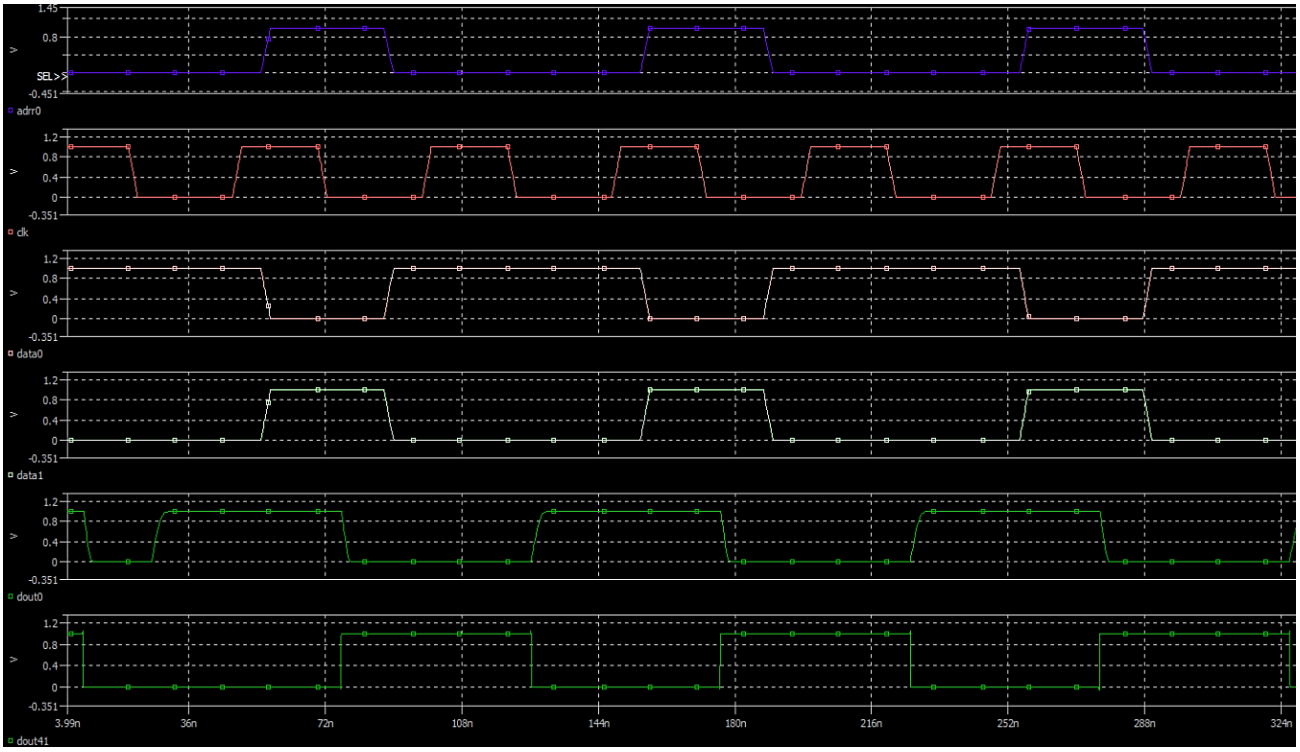


Рис. 3.6 - Часова діаграма виконання тесту checkerboard1 для RF1 20x42 (1.0 В, tt, 25°C)

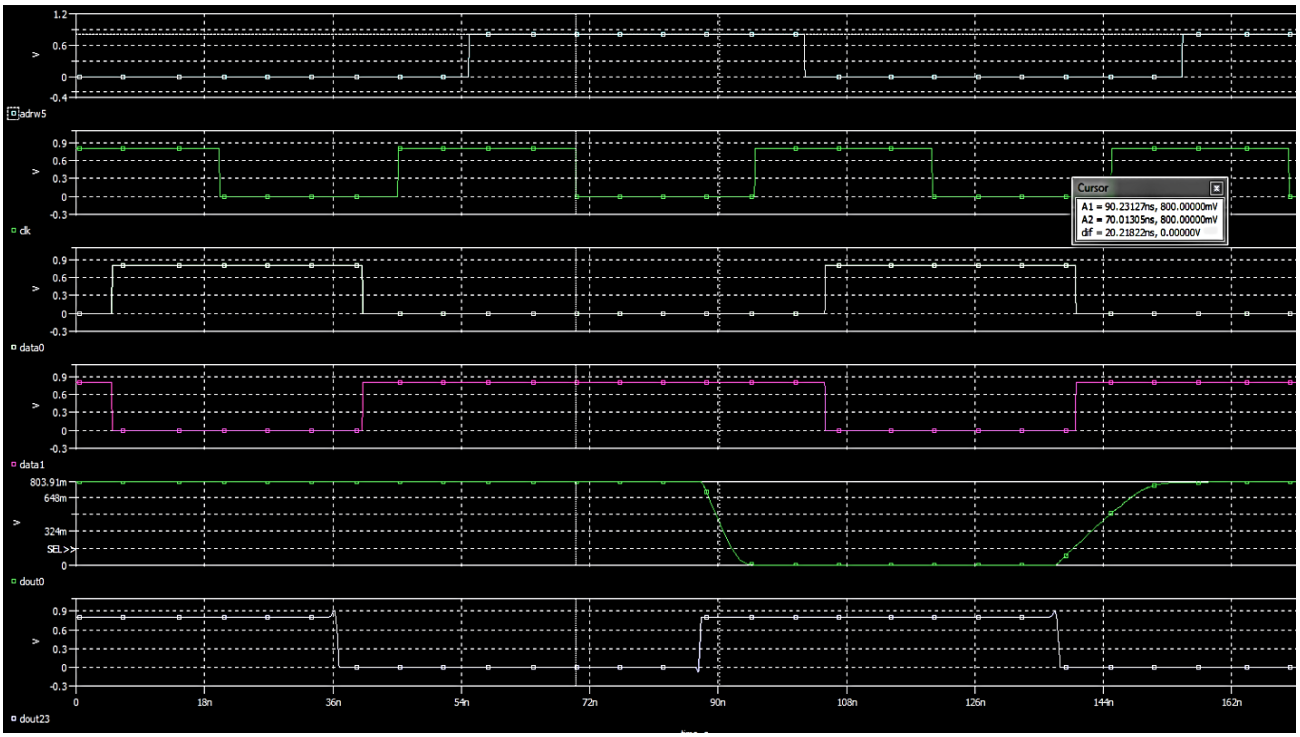


Рис. 3.7 – Часова діаграма виконання тесту checkerboard1 для RF2 56x24 (0.8 В, ss, -40°C)

Таблиця 3.1

Результати моделювання часу доступу для всіх крайових умов процесу для реалізацій пам'ятей RF1 та RF2

Назва кута	**Vdd(B)**	**TJ(C°)**	**Процес**	**T_{acc}, нс CL=0, tr=tf=0.1нс**	
				RF1	**RF2**
Worst	1.62	85	SS	2.15	1.57
	0.8	-40		19.42	20.21
	1.0	-40		5.81	5.5
Typical	0.9	25	TT	5.79	5.48
	1.0	25		4.02	4.03
	1.1	25		3.38	3.16
Best	1.98	-40	FF	0.97	1.0
	1.0	85		3.25	3.09
	1.2	-40		1.96	1.88

Найбільше значення часу доступу було отримано для випадку навантаження виходу: $CL = 0.5$ пФ і становило: T_{acc} = 25.6 нс для пам’яті RF1 20x42 при умовах: V_{DD}= 0.8 В, ss, -40°C. Ці результати показують, що T_{acc} суттєво залежить не лише від напруги живлення та крайових умов технологічного процесу, але й від навантажувальної ємності вихідних сигналів пам'яті. Збільшення CL до 0.5 пФ призводить до помітного зростання T_{acc}, що необхідно враховувати при оцінці максимальної робочої частоти згенерованих RF-макросів [30, 31].

3.4.4. Розведення шин живлення головних блоків верхнього рівня

Приймається, що зовнішнє підключення до шин V_{DD} та GND виконується у всіх чотирьох кутах пам’яті, причому напруга в цих точках дорівнює: $VDD = V_{DD}$, $GND = 0$. Це припущення використовується для оцінки падіння напруги в лініях живлення та перевірки допустимих середніх струмів у металевих з’єднаннях [29].

Блок ADFP — це збірний блок драйверів адрес портів читання та запису. Драйвери адрес порту читання розташовані знизу зліва, а драйвери адрес порту запису — зверху. Кожна реалізація пам’яті містить 5 або 6 блоків ADFP, які підключаються до кільця живлення лінією M1 шириною: W = 0.35 мкм.

Максимальний середній струм споживання шести блоків ADFP при: V_{DD} = 1.98 В, F = 10 МГц становить: $I_{ADFP,max}$ = 43 мкА після масштабування топології. Допустима межа струму для цієї лінії: I_{limit} < 330 мкА.

Отже, запас по допустимому струму становить приблизно: $\frac{330}{43} \approx 7.7$, що свідчить про достатність вибраної ширини лінії живлення для цього блока.

Блок CONTROL_R — це схема керування портом читання, розташована в нижній лівій частині пам’яті.

Блок підключений:

- до кільця GND лінією ME1 шириною: W_{GND} = 0.31 мкм з допустимим струмом: $I_{limit,GND}$ = 270 мкА;
- до кільця V_{DD} лінією ME1 шириною: W_{VDD} = 0.36 мкм з допустимим струмом: $I_{limit,VDD}$ = 300 мкА.

Максимальний середній струм споживання блока CONTROL_R становить: $I_{CONTROL_R,max}$ = 9.78 мкА. Таким чином, навіть для вужчої GND-лінії запас за струмом становить: $\frac{270}{9.78} \approx 27.6$, що підтверджує достатність розведення живлення для блока керування читанням.

Блок CONTROL_W — схема керування портом запису, розташована у верхній лівій частині пам'яті. Блок підключається до кілець GND та V_{DD} лініями ME1 шириною: W = 0.36 мкм з допустимим струмом: I_{limit} = 300 мкА. Максимальний середній струм споживання блока: $I_{CONTROL_W,max}$ = 3.9 мкА. Запас за допустимим струмом становить: $\frac{300}{3.9} \approx 76.9$ що свідчить про значний запас надійності ліній живлення.

Блок INPUT_BUF — збірний блок вхідних буферів, розташований у верхній частині пам'яті. Кожен блок INPUT_BUF підключається до кілець V_{ss} / GND лініями ME1 шириною: W = 0.23 мкм з допустимим струмом: I_{limit} = 200 мкА. Максимальний середній струм споживання блока становить: $I_{INPUT_BUF,max}$ = 2.8 мкА з відповідним запасом за струмом: $\frac{200}{2.8} \approx 71.4$. Таким чином, лінії живлення вхідних буферів мають значний запас відносно допустимого середнього струму.

Блок DXDP — збірний блок дешифраторів рядків портів читання та запису, розташований з правого боку масиву пам'яті. Блок підключено до кільця GND лінією ME1 шириною: W = 0.6 мкм з допустимим струмом: I_{limit} = 500 мкА. Максимальний середній струм споживання DXDP становить: $I_{DXDP,max}$ = 2 мкА із запасом за струмом $\frac{500}{2} = 250$.

Отже, розведення живлення дешифраторів рядків має дуже великий запас щодо середнього струму.

Блок REG_BUF — збірний блок підсилювачів зчитування та вихідних буферів, розташований знизу відносно масиву комірок пам'яті. Підсилювачі зчитування підключаються до кільця V_{DD} лінією ME1 шириною: W = 0.5 мкм з допустимим струмом: I_{limit} < 420 мкА. Ця лінія є спільною для всіх блоків **REG_BUF**. Для найгіршого випадку, що відповідає 42 стовпцям слова, сумарний струм споживання підсилювачів зчитування становить: $I_{SA,total}$ = 48.3 мкА при питомому споживанні одного підсилювача: I_{SA}= 0.115 мкА/МГц.

Інша частина REG_BUF підключається до кілець VDD/GND лініями ME1 шириною: W= 0.23 мкм з допустимим струмом: I_{limit} = 200 мкА. Максимальний середній струм споживання REG_BUF без навантажувальної ємності становить: $I_{REG_BUF,unload}$ = 3.4 мкА. Для навантаженого виходу з ємністю CL до цього значення додається динамічна складова:

$$I(\text{мкА}) = CL(\text{пФ}) \cdot V_{DD}(\text{В}) \cdot F(\text{МГц}) \qquad (3.1)$$

Для: CL = 0.5 пФ додатковий середній струм становить: I_{add} = 10 мкА. Таким чином, навіть з урахуванням навантажувальної ємності середній струм залишається суттєво меншим за допустимі обмеження ліній живлення.

Блок RF CELL ARRAY розташований у центральній та правій частинах пам’яті. Масив комірок підключено до правої частини кілець живлення V_{DD}/GND такими лініями:

- ME1 для V_{DD}, ширина: W_{VDD}= 0.23 мкм;
- ME3 для GND, ширина: W_{GND}= 1 мкм.

Ця частина кільця живлення використовується лише для підключення матриці комірок пам’яті. Такий підхід застосовано з метою зменшення впливу падіння напруги, що виникає під час перемикання інших схем, на запас статичної стійкості комірок пам’яті (SNM).

Інакше кажучи, локальне виділення ділянки живлення для матриці комірок дозволяє зменшити взаємний вплив цифрових периферійних блоків на найбільш чутливу частину пам’яті — бітові комірки, що особливо важливо при роботі на знижених напругах живлення [3-10, 27, 28, 29].

3.4.5 Узагальнення

Аналіз розведення шин живлення блоків верхнього рівня показує, що для всіх основних функціональних блоків пам’яті середні струми споживання значно нижчі за допустимі обмеження відповідних шин металізації. Найбільші відносні навантаження припадають на блоки ADFP та підсилювачі зчитування у складі REG_BUF, однак і для них зберігається достатній запас за струмом.

Особливо важливим є окреме підключення матриці комірок пам’яті до виділеної ділянки кільця живлення навколо матриці, що зменшує вплив перемикань периферійної логіки на SNM комірок пам’яті та підвищує надійність роботи RF при знижених напругах живлення.

3.5. Врахування паразитних RC-параметрів у post-layout характеризації

Із безперервним зменшенням топологічних норм вплив паразитних параметрів, екстрагованих із топології інтегральних схем (IC), стає дедалі суттєвішим. Особливо це стосується мікросхем пам'яті, для яких паразитні ємності та опори міжз'єднань можуть відігравати визначальну роль при розрахунку затримок, часу доступу (access time) та струму споживання [30-32]. Під паразитними компонентами в даному контексті розуміються паразитні ємності та опори, пов'язані з провідними шарами топології, зокрема полікремнієм (polysilicon / poly / gate layer) і металевими шарами (metal layers).

Параметри металевих шарів, міжшарових діелектриків, контактів і via-структур визначають паразитні опори та ємності міжз'єднань, які впливають на електрофізичні характеристики напівпровідникової схеми, часові параметри, частотні обмеження та результати post-layout SPICE-моделювання.

Топологія інтегральної схеми може бути представлена як сукупність активних і пасивних компонентів — MOS- та bipolar-транзисторів, резисторів, конденсаторів, діодів, індуктивностей та інших елементів, описаних у фабричних правилах перевірки відповідності топології схемі LVS (Layout Versus Schematic). LVS-правила застосовуються як для перевірки відповідності топології вихідному електричному опису схеми (schematic netlist), так і для формування нетлістів, придатних для подальшого SPICE-моделювання. У традиційних LVS-правилах з'єднання виводів приладів виконується через шари міжз'єднань, які в базовому варіанті вважаються такими, що мають нульовий опір і нульову ємність [31, 32, 35].

Моделі вбудованих приладів (built-in devices) — MOS, RES, CAP, DIO, BIP, IND — уже містять опис власних параметрів, включаючи відповідні паразитні опори

та ємності [30, 31]. Тому під час екстракції паразитних параметрів такі складові не повинні дублюватися. Під built-in devices у цій роботі розуміються прилади, визначені у фабричній специфікації процесу (Process Specification, PS) та LVS-правилах разом із відповідними шарами, іменами та типами пристроїв. Кількість таких елементів істотно залежить від конкретної технології та може змінюватися від кількох десятків до кількох тисяч. Саме тому коректна екстракція паразитних параметрів стає критично важливою для формування SPICE-сумісного нетліста, який відображає реальну електричну поведінку схеми після топологічної реалізації.

Існують два основні класи методів екстракції паразитних ємностей. Перший клас — це 2D-, 2.5D- та 3D-чисельні методи, що базуються на розв’язанні рівнянь Лапласа, зокрема методом скінченних різниць (finite-difference method), методом скінченних елементів (finite-element method), методом моментів (method of moments) або граничних елементів (boundary-element methods). Основною перевагою таких підходів є висока точність. Водночас їх практичне застосування обмежується значними обчислювальними витратами, тривалим часом екстракції та порівняно невеликими розмірами топологій, які можуть бути оброблені за прийнятний час [32, 36, 37].

Другий клас становлять аналітичні та rule-based методи, у яких параметри моделей паразитних ємностей і опорів визначаються безпосередньо з топології за допомогою інструментів фізичної верифікації. Такі методи, як правило, поступаються чисельним підходам за точністю, однак мають суттєву практичну перевагу: вони дозволяють виконувати екстракцію для інтегральних схем практично довільного розміру, включаючи великі memory-макроси та складні SoC, де застосування повноцінних 3D-чисельних методів є надто дорогим або технічно недоцільним [30, 31, 35, 38].

У даній роботі розглядається застосування та адаптація фабрично сертифікованих правил LVS для екстракції паразитних параметрів. Для виконання паразитної екстракції та перевірки методології на реальних проєктних даних використано інструмент фізичної верифікації Silvaco SmartDRC/LVS, який застосовується як основа для екстракції паразитних ємностей. Запропонована методологія орієнтована на топології, що містять один шар полікремнію або gate/poly layer та до семи металевих шарів M1–M7, характерних для багатьох mature process nodes. При цьому підхід може бути адаптований і до більш складних технологій із M8–M12 металевими шарами з відносно невеликим впливом на продуктивність екстракції.

Ключовою перевагою запропонованого підходу є використання вже існуючих, фабрично сертифікованих LVS extraction rules. Це дозволяє не створювати повністю новий набір правил, а підключати секцію паразитної екстракції як додатковий external include file до наявного rule deck. Відповідні топологічні та технологічні обмеження визначаються у Process Specification (PS) та Design Rules Manual (DRM) конкретної фабричної технології. LVS-перевірка та екстракція паразитних ємностей виконуються інструментом SmartLVS із використанням спеціалізованої підмножини команд PoWerful Ruledeck Language (PWRL). PWRL-rule deck описує активні прилади, паразитні ємності, з’єднання та формує SPICE-compatible netlist, який може бути безпосередньо використаний для SPICE simulations.

У правилах екстракції, окрім шарів приладів, використовуються також ідентифікатори device name, device type та simulation model name. Device type визначає, чи екстрагується прилад як стандартний built-in device, наприклад MOS, RES, CAP, DIO, BIP або IND, чи як custom-defined X-cell. Для задач SPICE-моделювання тип приладу має менше значення, ніж для безпосереднього LVS-порівняння, оскільки вихідний нетліст паразитної екстракції призначений

передусім для моделювання, а не для логічного зіставлення з вихідною схемою. У цьому випадку device name переважно використовується для цілей LVS, тоді як під час формування simulation netlist він замінюється на відповідний simulation model name.

Важливою особливістю запропонованої методології є те, що розмір інтегральної схеми в принципі не обмежується кількістю приладів або обсягом топології. Це суттєво відрізняє rule-based LVS-driven extraction від чисельних методів, які хоча й забезпечують високу точність для локальних 2D-, 2.5D- або 3D-задач, проте погано масштабуються для великих промислових топологій. Тому для задач екстракції паразитних параметрів великих memory type ICs, включаючи SRAM/RFSRAM-макроси, запропонований підхід є практично доцільним компромісом між точністю, продуктивністю та можливістю інтеграції у стандартний потік фізичної верифікації.

У ході екстракції розглядається три основні типи паразитних ємностей. Зазвичай структура провідних шарів, для яких у технологічній документації надаються значення паразитних ємностей, має вигляд, подібний до наведеного на рис. 3.8 [30, 31]:

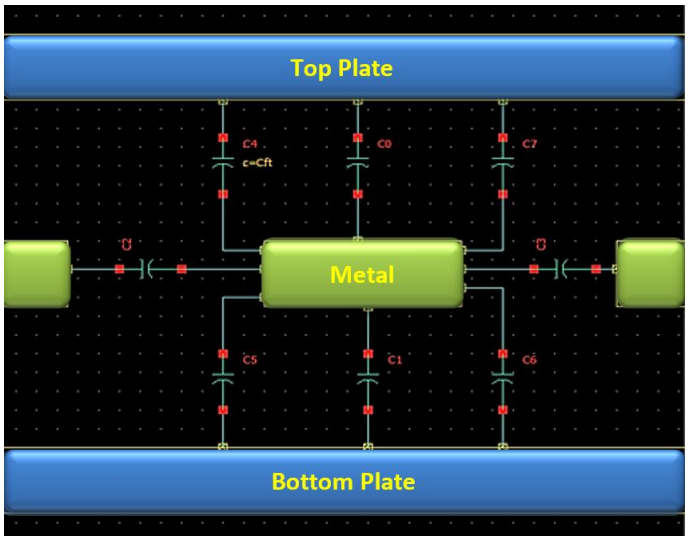


Рис. 3.8. Три типи паразитних ємностей

На рис. 3.8 шар Top Plate відповідає верхньому провідному шару, а Bottom Plate — нижньому провідному шару. Якщо нижній провідний шар зазвичай завжди присутній у вигляді підкладки (Sub), то верхній провідний шар може бути відсутнім [30]. Тому для певного центрального шару значення паразитних ємностей, як правило, подаються у двох варіантах: за наявності верхнього провідного шару та без нього. Ємності C0 і C1 відповідають площинним паразитним ємностям (**area capacitance**), що виникають у зоні перекриття центрального шару з верхнім шаром (Cat) або з нижнім шаром (Cab) відповідно. Ємності C2 і C3 є ємностями зв'язку (**coupling capacitance**) між провідниками, розташованими в одному й тому самому центральному шарі (Cc).

Ємності C5, C6 та C4, C7 відповідають крайовим, або **fringe capacitance**, які виникають між боковими поверхнями центрального шару та верхнім або нижнім провідним шаром.

Для екстракції паразитних опорів і ємностей у межах даного підходу, запропонованого автором, розглядаються лише області міжз'єднань (interconnect regions) — тобто шари полікремнію (polysilicon) та металеві шини (metal buses), які використовуються у правилах LVS extraction rules для формування електричних з'єднань між приладами [31, 33, 34, 40–42].

Визначення назв таких шарів, як правило, не є складним, оскільки вони вже присутні у фабричних LVS rules. Наприклад, контакти до вбудованих резисторів (built-in resistors) можна ідентифікувати через металеві та полікремнієві шари, що описані у правилах LVS і використовуються як шини міжз'єднань для з'єднання built-in devices. Практична перевірка на значній кількості rule decks різних фабрик показала, що в більшості випадків такий спосіб ідентифікації шарів забезпечує коректний результат.

У спрощеній постановці задачі екстракції паразитних опорів усі полікремнієві та металеві лінії міжз’єднань розкладаються на горизонтальні та вертикальні сегменти, які надалі екстрагуються окремо. Такий підхід дозволяє формалізувати процедуру розрахунку паразитного опору для провідних шин різної орієнтації та геометрії.

Схематичне подання типової структури металізації (metal stack) для трьох металевих шарів M1–M3 та шару полікремнію наведено на рис. 3.9.

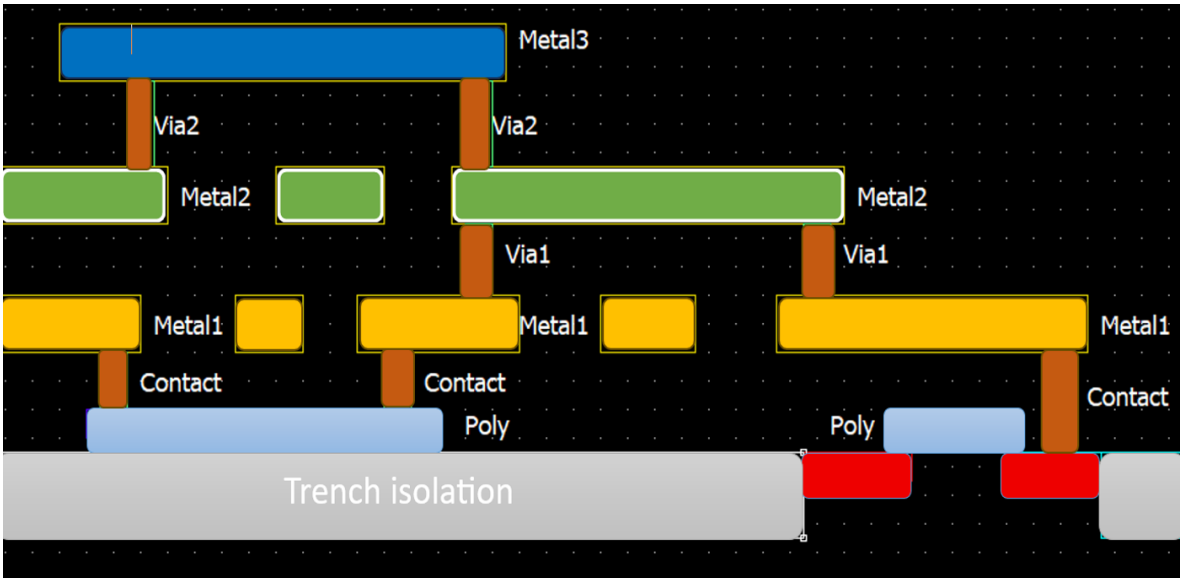


Рис. 3.9 – Трирівнева структура металізації (M1-M3)

На цьому рисунку також показано контакти шару M1 до полікремнію (contact) та міжметалеві з’єднання — vias. Екстракція опорів таких елементів є необхідною, оскільки їхній внесок у загальний паразитний опір не можна вважати незначним. Окремі числові значення технологічних параметрів у цьому підрозділі наведено у знеособленому вигляді, оскільки вони походять із process specification відповідної фабрики та є конфіденційними. Тому в роботі не розкривається назва фабрики, а наведені значення використовуються лише для ілюстрації порядку величин та інженерної значущості контактних і міжшарових опорів.

Наприклад, для технологій 130–180 нм опір контакту до полікремнію зазвичай становить приблизно 10–15 Ω. Якщо для шару M1 прийняти характерний питомий поверхневий опір (sheet resistance) близько: R_s = 0.1 Ω / □, то такий контактний опір еквівалентний опору фрагмента металевої лінії мінімальної ширини 0.16 мкм довжиною приблизно 16–24 мкм.

Детальна методологія rule-based layout-driven parasitic RC extraction розроблена та опублікована в [31]. У даному розділі вона використовується як частина маршруту post-layout characterization для згенерованих макросів RFSRAM.

У межах дисертаційної роботи не дублюється повний математичний апарат Smart-CX/SmartRX, оскільки він наведений у відповідних публікаціях автора. Натомість у цьому розділі розглянуто приклад застосування цих методів для post-layout SPICE-характеризації згенерованих макросів RFSRAM [30, 31].

Окремий сучасний напрям становлять методи паразитної екстракції, що використовують машинне навчання та нейромережеві моделі для прискорення оцінювання паразитних ємностей міжз'єднань. Такі підходи демонструють перспективність для задач capacitance extraction, однак у межах даної роботи пріоритет надано rule-based LVS-driven маршруту, оскільки він безпосередньо інтегрується з фізичною верифікацією, фабричними rule decks та формуванням SPICE-compatible netlist для post-layout моделювання макросів пам'яті [43–47].

3.6. Оцінювання середнього динамічного струму та енергоспоживання згенерованих макросів RFSRAM

Середній динамічний струм (Average Dynamic Current) — це середній за часом струм, який споживає інтегральна схема під час виконання динамічних операцій (читання, запису або перемикання станів). Він визначається як часове

усереднення миттєвого струму протягом повного робочого циклу та характеризує енергію, витрачену на заряджання та розряджання як основних функціональних, так і паразитних ємностей схеми [29, 30, 31]. На відміну від пікового або сумарного миттєвого струму окремих функціональних блоків, середній динамічний струм враховує реальну часову активність блоків і коефіцієнт перемикань.

Математично це можна подати як площу під графіком миттєвого струму, поділену на тривалість періоду:

$$I_{AVG} = \frac{1}{T}\int_0^T I(t)\,dt \quad (3.2);$$

- $I(t)$— миттєвий струм,
- T— період роботи.

Зниження середнього динамічного струму в пам’яті типу RFSRAM є важливим завданням для підвищення енергоефективності, збереження цілісності даних, підвищення надійності та забезпечення стабільної роботи пам’яті в різних режимах експлуатації. Поряд із традиційними вимогами до площі, швидкодії та загального енергоспоживання, для кожного екземпляра Register File (RF) окремо постає задача мінімізації середнього динамічного струму, оскільки саме він безпосередньо впливає на споживану потужність.

Площі розроблених екземплярів пам’яті визначалися для відповідних конфігурацій RF. Відповідно до специфікації технологічного процесу TSMC 152G PDK, який є результатом масштабування 180-нм технології з коефіцієнтом 0.84, середній струм у лініях металізації M1–M5 не повинен перевищувати 1 мА. Для шару M1 мінімальна ширина шини після масштабування становить 0.2 мкм, і для такої лінії допустиме значення середнього струму не повинно перевищувати 200 мкА.

Таблиця 3.2

Приклади розмірів та площ макросів RF

Ім'я макросу RF пам'яті	**Ширина(width) (X-розмір),** мкм	**Висота (height) (Y-розмір),** мкм	**Площа (area),** мкм ^{2}
RF1_20x42	171.745	124.285	21345.3
RF2_56x24	127.480	242.855	30959.2

У таблицях 3.3 і 3.4 наведено значення середніх струмів для пам'яті RF1 20x42 та RF2 56x24 для всіх основних блоків та кутів напруги живлення [29].

Оскільки середній динамічний струм у цьому випадку визначається переважно напругою живлення, ємнісним навантаженням та активністю перемикань, результати згруповано за voltage supply corners.

SPICE simulations проводилися для режиму, у якому сигнали: $WEN = REN = High$, усі адреси змінюються, а вхідний тестовий шаблон даних відповідає типу checkerboard, тобто всі вхідні та вихідні дані перемикаються [29, 31].

Споживана потужність RFSRAM у режимі читання/запису може бути розрахована за даними таблиць 3.3 і 3.4 з використанням наступної формули:

$$P = V_{vdd} \times \left(I_{AFDP} \times k \times 2 + I_{DXDP} \times 2 + I_{INPUT_BUF} \times col + I_{REG_{BUF}} \times col + I_{CONTROL_W} + I_{CONTROL_R}\right) \quad \text{мкВт /МГц} \qquad (3.3)$$

де, k – кількість вхідних адрес [29], col — розрядність слова; струми у таблицях нормовані на частоту, тому результат має одиниці мкВт/МГц. Для абсолютної потужності потрібне множення на робочу частоту роботи пам'яті в МГц.

Наприклад, для пам'яті **RF1 20x42**, що працює при: $V_{DD} = 0.8$ В споживана потужність становить: P = 3.88 мкВт / МГц.

Максимальний **піковий струм** ($\mathbf{J_p}$) для провідникових шин металів M1 — M5:

$$J_p = 50 \times \sqrt{duty_cylce} \quad \text{мкА/мкм}^2 \ (3.4)$$

Таблиця 3.3

Споживання струму з розподілом по блоках для макросу RF1 20x42, мкА **/МГц**

V_{DD}, В	ADFP (r9)	DXDP (r11)	INPUT_BUF (r12)	REG_BUF (r1+r7)	CONTROL_W (r13)	CONTROL_R (r14)
0.8	0.0548	0.038	0.031	0.059	0.12	0.33
0.9	0.0673	0.045	0.0353	0.062	0.136	0.38
1	0.0751	0.05	0.038	0.064	0.153	0.435
1.1	0.081	0.058	0.04	0.081	0.177	0.48
1.2	0.086	0.063	0.05	0.102	0.19	0.537
1.8	0.151	0.11	0.081	0.139	0.314	0.87

Таблиця 3.4

Споживання струму з розподілом по блоках для макросу RF2 56x24, мкА **/МГц**

V_{DD}, В	ADFP (r9)	DXDP (r11)	INPUT_BUF (r12)	REG_BUF (r1+r7)	CONTROL_W (r13)	CONTROL_R (r14)
0.8	0.12	0.033	0.044	0.082	0.14	0.286
0.9	0.143	0.037	0.053	0.097	0.152	0.33
1	0.155	0.042	0.06	0.105	0.173	0.38
1.1	0.18	0.048	0.066	0.115	0.195	0.42
1.2	0.187	0.052	0.075	0.126	0.21	0.46
1.8	0.3	0.087	0.125	0.195	0.35	0.74

Максимальний коефіцієнт заповнення (duty cycle) для періоду внутрішніх сигналів: t_p = 100 нс, що відповідає частоті: F = 10 МГц для обох типів пам'яті не перевищує 0.14 [29].

Це значення отримано для умов: V_{DD} = 0.8 В, SS, -40 С, CL = 0.5 пФ, OUT_BUF.

Отже, **максимальний піковий струм** для лінії мінімальної ширини металізації першого рівня (M1) має бути меншим за 10.5 мА, що приблизно на два порядки перевищує максимальні пікові струми, отримані для будь-якого блока в обох

досліджуваних схемах. Це свідчить про достатній запас за струмовим навантаженням ліній металізації.

Для розрахунку **середнього динамічного струму** була розроблена спеціальна схемотехнічна модель (schematic view), яка містила всі основні блоки RF-пам'яті [29]. Вона дозволяла окремо вимірювати:

- струм однієї бітової комірки пам'яті I_c;
- вхідний струм I_i;
- струм вихідних буферів I_o;
- струм адресного буфера I_a;
- струм дешифратора рядків I_d;
- струм блока керування читанням I_r;
- струм блока керування записом I_w.

Загальний середній динамічний струм I_{tot} для пам'яті RF1 20x42 розраховувався для кожного PVT corner як сума внесків окремих функціональних блоків:

$$I_{tot} = 20 \times 42 \times I_c + 42 \times (I_i + I_o) + I_a \times 12 + I_r + I_w + I_d \times 20 \times 2 \quad (3.5)$$

для пам'яті RF2 56x24:

$$I_{tot} = 56 \times 24 \times I_c + 24 \times (I_i + I_o) + I_a \times 12 + I_r + I_w + I_d \times 56 \times 2 \quad (3.6)$$

Висновок

З урахуванням результатів SPICE моделювань та подальших розрахунків було встановлено, що максимальний середній динамічний струм становить:

$I_{tot} = 2.344$ мкА – для пам'яті RF2 56x24;

$I_{tot} = 1.591$ мкА – для пам'яті RF1 20x42;

Ці значення були отримані для так званого worst-case corner: $V_{DD} = 1.65$ В, $T = 85°\text{C}$, SS. Згідно з еталонними (референсними) даними, наданими фабрикою, середній динамічний струм для пам'яті тієї самої конфігурації 20x42 становив: $I_{ref} = 6.45$ мкА для умов: $V_{DD} = 1.1$ В, $T = 125°\text{C}$, FF.

Отриманий у роботі результат є майже втричі кращим за еталонний зразок, що пояснюється спеціальною оптимізацією, виконаною на рівні окремих функціональних блоків пам'яті [29]. Таким чином, проведене моделювання підтверджує ефективність запропонованого підходу до зниження середнього динамічного струму в RFSRAM та демонструє практичну доцільність використання такої архітектури у низькоенергетичних застосуваннях.

3.7. Програмна реалізація та генерація вихідних файлів компілятора…

Програмна реалізація компілятора пам'яті побудована за модульним принципом і поєднує параметризований скриптовий рівень (Lua) керування генерацією з низькорівневим ядром (C++) роботи з топологічною базою даних (GDS-II). Скриптова частина реалізована мовою Lua і відповідає за опис архітектури пам'яті, вибір базових комірок, формування ієрархії блоків, розміщення елементів, з'єднання внутрішніх шин, формування виводів верхнього рівня, генерацію схемного LVS-нетліста та підготовку бібліотечних timing/power-описів.

На відміну від монолітної програмної реалізації, такий підхід дозволяє відокремити алгоритмічну логіку компілятора від низькорівневих операцій із геометричними примітивами топології [1, 2]. Lua-рівень виконує роль керуючого опису, тоді як виконуваний модуль та відповідні бібліотеки layout SDK забезпечують створення геометричних об'єктів, інстанціювання елементів,

формування контактів, шин, виводів і вихідних файлів. Подібна архітектура відповідає загальному підходу, де скриптова частина поділена на framework-модулі, layout-блоки, technology-dependent files, netlist generation scripts та файли генерації вихідних описів, зокрема LEF, LIB, Verilog, Datasheet, Cell Arrangement.

Для задач даної дисертаційної роботи найбільш суттєвими є такі функціональні частини програмної реалізації:

- формування регулярної матриці 8T-комірок пам’яті;
- збірка верхнього рівня макросу RFSRAM з периферійних блоків;
- генерація схемного нетліста для LVS-перевірки;
- генерація timing/power LIB-файлів на основі результатів SPICE-характеризації;
- відокремлення технологічних параметрів і бібліотеки leaf cells від загальної логіки компілятора.

3.7.1. Модульна структура програмної реалізації

Програмна структура генератора макросів пам’яті складається з декількох логічних рівнів. Перший рівень утворюють framework-модулі, які забезпечують базові механізми роботи з об’єктами, перевірку параметрів, створення ієрархічних блоків, роботу з виводами, посиланнями на дочірні блоки та технологічними параметрами. До цього рівня належать модулі перевірок, допоміжних математичних і мовних операцій, а також основний framework-модуль, що надає API для створення топологічних блоків.

Другий рівень становлять технологічно-залежні модулі. Вони містять опис шарів, мінімальних ширин металевих і полікремнієвих шин, відстаней між провідниками, параметрів контактів, via-структур, pin-шарів, текстових шарів і технологічних обмежень. Відокремлення цих параметрів від основної логіки

генератора дозволяє адаптувати компілятор до різних технологічних процесів без повного переписування алгоритмів формування макросу пам’яті. У документації Lua-частини такий підхід реалізовано через модулі tech.lua і leafcells.lua, де перший містить технологічні параметри, а другий — опис базових leaf-cell блоків.

Третій рівень утворюють layout-блоки. Кожен блок описується як програмний об’єкт із власними параметрами, дочірніми блоками, виводами та процедурою створення геометрії. Для такого об’єкта характерні дві фази побудови: фаза ініціалізації та фаза створення. На фазі ініціалізації визначаються дочірні блоки, параметри, розміри та допоміжні дані, необхідні для побудови макросу. На фазі створення виконується розміщення дочірніх блоків, прокладання шин, встановлення контактів, формування виводів і передача створених геометричних об’єктів у layout database. Такий двофазний підхід дозволяє передавати інформацію як зверху вниз — від верхнього рівня до дочірніх блоків, так і знизу вгору — від дочірніх блоків до блока верхнього рівня.

Четвертий рівень відповідає за генерацію вихідних представлень. До них належать топологічне представлення макросу, схемний нетліст для LVS, абстрактне представлення для розміщення у складі SoC, допоміжні файли фізичної верифікації, а також timing/power-бібліотеки. Окрему роль відіграють netlist generation scripts, які формують ієрархічний опис макросу пам’яті у вигляді SPICE/LVS-compatible netlist [1, 2].

Загальна схема роботи програмної реалізації може бути описана такою послідовністю:



Рис. 3.10 Загальна схема роботи програмної реалізації.

Такий модульний підхід забезпечує повторне використання базових блоків і дозволяє реалізувати компілятор пам’яті як параметризовану систему, що генерує різні конфігурації RFSRAM без ручної побудови кожного окремого макросу.

3.7.2. Алгоритм формування матриці комірок пам’яті

Одним із ключових етапів роботи компілятора є формування регулярної матриці комірок пам’яті. Для RFSRAM-пам’яті така матриця складається з 8T-комірок, які організовані за рядками та стовпцями відповідно до заданої користувачем конфігурації. Вхідними параметрами для цього етапу є кількість слів, розрядність

слова, кількість рядків, кількість стовпців, тип комірки, технологічний процес і правила розміщення допоміжних елементів.

Формування матриці виконується за таким алгоритмом. Спочатку компілятор перевіряє допустимість вхідної конфігурації. Кількість слів і розрядність мають відповідати діапазонам, підтриманим архітектурою компілятора та бібліотекою базових блоків. Після цього вибирається leaf cell 8T-комірки пам’яті, яка відповідає заданому технологічному процесу та вимогам до low-power режиму [20-22, 27, 28].

Далі створюється регулярна структура розміщення комірок. Комірки повторюються у горизонтальному та вертикальному напрямках із кроками, визначеними розмірами leaf cell та технологічними правилами. Для кожної комірки визначається її положення у матриці, орієнтація, належність до відповідного рядка та стовпця, а також підключення до локальних шин читання, запису, живлення та землі.

Після створення регулярної частини матриці додаються допоміжні структури. До них можуть належати крайові або dummy-комірки, елементи вирівнювання топології, локальні контакти до шин живлення, а також технологічні структури, необхідні для збереження регулярності масиву. Наявність таких елементів важлива для забезпечення коректної топологічної реалізації та повторюваності електричних характеристик комірок усередині масиву.

Окремим етапом є формування виводів матриці. Внутрішні виводи окремих комірок піднімаються на рівень матриці та групуються у шини: word-line для портів читання і запису, bit-line для даних, вхідні та вихідні лінії, а також VDD/GND-шини. Така операція відповідає принципу pin promotion, коли локальні виводи дочірніх блоків стають доступними на рівні батьківського блока. У framework-описі Lua-частини для таких задач передбачені спеціальні функції створення виводів, роботи з pin-об’єктами, references та promotion of child pins.

Узагальнений алгоритм формування матриці пам’яті можна подати таким чином:

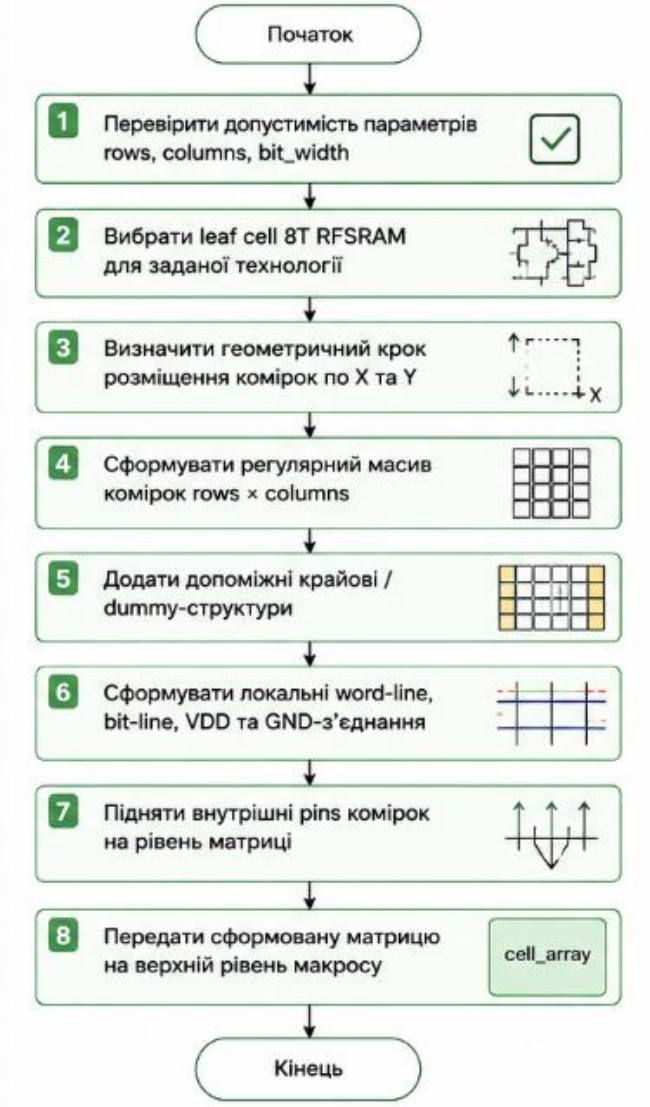


Рис. 3.11 Узагальнений алгоритм формування матриці пам’яті.

3.7.3. Алгоритм збірки верхнього рівня макросу пам’яті

Після формування матриці комірок компілятор виконує збірку верхнього рівня макросу пам’яті. На цьому етапі регулярний масив комірок поєднується з периферійними функціональними блоками: адресними драйверами, дешифраторами рядків, блоками керування читанням і записом, вхідними

буферами, вихідними буферами, sense/read-логікою, схемами контролю конфліктів та шинами живлення [1, 2, 27, 28].

Збірка top-level макросу виконується ієрархічно. Кожний функціональний блок створюється як дочірній блок верхнього рівня. Його параметри залежать від конфігурації пам’яті: кількості рядків, кількості стовпців, розрядності слова, кількості адресних бітів, обраної топологічної орієнтації та вимог до сигналів читання і запису.

У процесі розміщення блоків компілятор враховує не лише логічні зв’язки, але й геометричну структуру майбутнього layout. Наприклад, блоки дешифраторів розташовуються поряд із матрицею комірок з боку word-line, буфери даних — з боку bit-line, а блоки керування — у лівому нижньому куті, що оптимізує трасування та мінімізує довжину критичних керуючих сигналів [30, 31]. Такий принцип дозволяє зменшити паразитні ємності та опори з’єднань, а також спростити подальше post-layout SPICE-моделювання.

Після розміщення функціональних блоків виконується формування внутрішніх шин. До них належать адресні шини портів читання і запису, шини вхідних даних, шини вихідних даних, сигнали дозволу читання і запису, тактові сигнали, внутрішні сигнали керування, а також VDD/GND-шини. Для кожної групи сигналів вибирається відповідний металевий шар, напрямок трасування та набір контактів між шарами.

Загальна послідовність збірки верхнього рівня макросу така:



Рис. 3.12 Узагальнений алгоритм збірки верхнього рівня макросу пам'яті.

Результатом цього етапу є повний топологічний макрос пам'яті, готовий для фізичної верифікації та екстракції паразитних параметрів.

3.7.4. Генерація схемного LVS-нетліста

Для перевірки коректності автоматично згенерованої топології необхідно сформувати схемний нетліст, який відповідає тій самій ієрархії, що й layout-представлення макросу. Такий нетліст використовується під час LVS-перевірки для встановлення відповідності між топологією та схемним описом [31, 35].

Генератор LVS-нетліста виконує формування ієрархічного опису макросу пам'яті. Він створює top-level subcircuit, описує зовнішні виводи пам'яті, формує внутрішні вузли, інстанціює дочірні блоки та підключає їх відповідно до структури, створеної компілятором layout-частини [1, 2, 31]. До таких дочірніх

блоків належать матриця комірок пам’яті, дешифратори, драйвери адреси, блоки керування, буфери даних та інші базові блоки (leaf cells).

LVS-нетліст має точно відображати:

- top-level pins макросу пам’яті;
- імена внутрішніх вузлів;
- ієрархію рівня блоків;
- відповідність зовнішніх (інтерфейсних) портів дочірніх блоків;
- підключення VDD/GND;

Узагальнений алгоритм генерації LVS-нетліста може бути поданий так:



Рис. 3.13 Узагальнений алгоритм генерації LVS netlist.

Коректність роботи генератора LVS-нетліста є критичною для всього маршруту, оскільки помилки у відповідності pin-ів, імен портів, внутрішніх вузлів або ієрархії блоків можуть призвести до LVS mismatch навіть за коректно сформованої топології [31, 35]. Тому в програмній реалізації передбачено використання єдиної структури даних для генерації layout та netlist.

3.7.5. Генерація timing/power LIB-файлів на основі SPICE-характеризації

Найбільш важливим програмним компонентом з точки зору подальшого використання згенерованого макросу пам'яті у цифрових маршрутах проєктування є генератор timing/power LIB-файлів. Якщо layout generator формує фізичну реалізацію макросу, а netlist generator забезпечує LVS-перевірку, то LIB generator перетворює результати SPICE-характеризації у бібліотечний формат, придатний для використання в системах логічного синтезу, статичного timing-аналізу та інтеграції макросу у SoC [1, 2, 29, 31].

Вхідними даними для LIB generator є результати SPICE-моделювання для заданої конфігурації пам'яті та набору PVT-кутів. Для кожного PVT-corner виконуються симуляції, за результатами яких визначаються access time, затримки вихідних сигналів, setup/hold constraints, динамічний струм, енергоспоживання, а також інші параметри, необхідні для опису поведінки макросу пам'яті [29, 31]. Результати таких симуляцій попередньо збираються у спеціальному Lua-файлі у вигляді таблиць characterization data.

У реалізованому маршруті timing/power характеристики подаються у вигляді двовимірних look-up tables. Для timing-характеристик використовуються таблиці залежності затримки від фронту вхідного сигналу та ємнісного навантаження виходу. Для constraint-характеристик використовуються таблиці setup/hold time залежно від відповідних фронтів керуючих і тактових сигналів. Для power-

характеристик використовуються таблиці, побудовані на основі результатів вимірювання динамічного струму або енергії перемикання у SPICE.

Таблиці часових параметрів сформовано для п'яти значень швидкості наростання вхідних сигналів (input slew rate) та п'яти значень ємнісного навантаження на виході. Затримки поширення сигналів описуються двовимірними таблицями розміром 5×5, побудованими за комбінаціями п'яти значень вихідного навантаження та п'яти значень швидкості наростання вхідного сигналу [29].

Часові параметри встановлення (Setup) і утримання (Hold) також подаються у вигляді таблиць 5×5, сформованих для п'яти значень швидкості наростання тактового сигналу та п'яти значень швидкості наростання інформаційного сигналу.

Швидкість наростання сигналів визначається в діапазоні від 10 % до 90 % напруги живлення, тоді як усі часові параметри вимірюються за рівнем логічного перемикання, що відповідає 50 % напруги живлення.

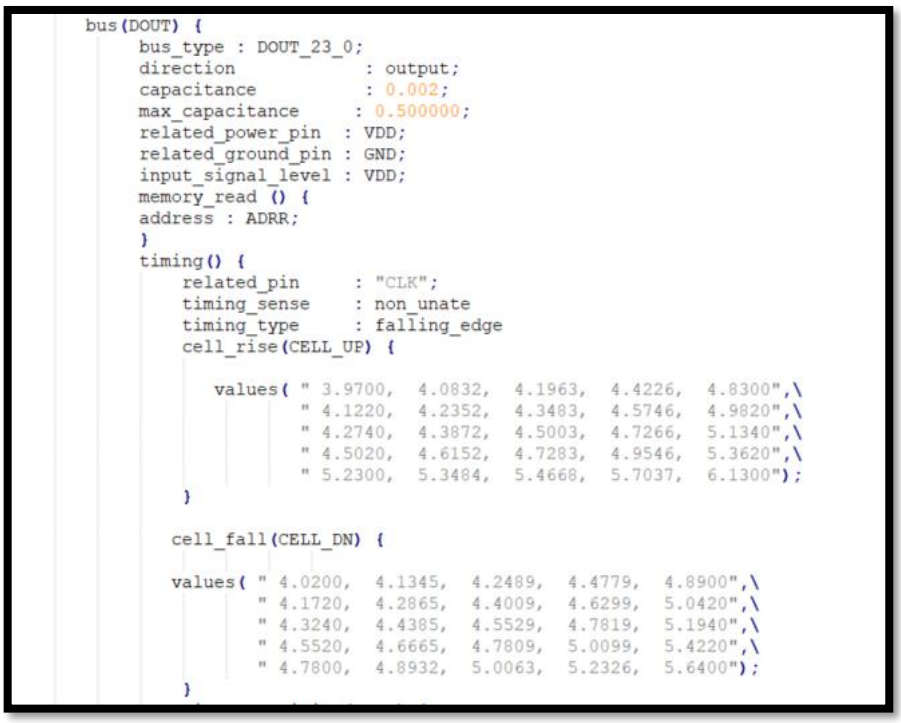
Характеризація виконується для 3 комбінацій технологічних, електричних і температурних умов (PVT-кутів), у результаті чого формуються 3 бібліотеки часових параметрів, кожна з яких відповідає певному поєднанню технологічного процесу, напруги живлення та температури.

Таблиця 3.5

Формат Look Up Table (5x5)

Індекс	**1**	**2**	**3**	**4**	**5**
Вхідний фронт (нс, 10−90 %)	0.10	0.250	0.500	1.000	2.000
Вихідне навантаження (пФ)	0.010	0.100	0.200	0.350	0.500

Приклад розрахунку 5×5 look-up таблиці зі згенерованого LIB-файлу можна побачити на Рис. 3.14:



```
bus(DOUT) {
    bus_type : DOUT_23_0;
    direction           : output;
    capacitance         : 0.002;
    max_capacitance     : 0.500000;
    related_power_pin   : VDD;
    related_ground_pin  : GND;
    input_signal_level  : VDD;
    memory_read () {
    address : ADRR;
    }
    timing() {
        related_pin     : "CLK";
        timing_sense    : non_unate
        timing_type     : falling_edge
        cell_rise(CELL_UP) {

           values( " 3.9700,  4.0832,  4.1963,  4.4226,  4.8300",\
                   " 4.1220,  4.2352,  4.3483,  4.5746,  4.9820",\
                   " 4.2740,  4.3872,  4.5003,  4.7266,  5.1340",\
                   " 4.5020,  4.6152,  4.7283,  4.9546,  5.3620",\
                   " 5.2300,  5.3484,  5.4668,  5.7037,  6.1300");
        }

        cell_fall(CELL_DN) {

        values( " 4.0200,  4.1345,  4.2489,  4.4779,  4.8900",\
                " 4.1720,  4.2865,  4.4009,  4.6299,  5.0420",\
                " 4.3240,  4.4385,  4.5529,  4.7819,  5.1940",\
                " 4.5520,  4.6665,  4.7809,  5.0099,  5.4220",\
                " 4.7800,  4.8932,  5.0063,  5.2326,  5.6400");
        }
```

Рис. 3.14 - Приклад значень 5×5 look-up таблиці зі згенерованого LIB-файлу для шини сигналів DOUT

У шаблонах CELL_UP і CELL_DN рядки відповідають значенням ємності вихідного навантаження, а стовпці — тривалостям фронту вхідного сигналу. Значення таблиць формуються на основі опорних даних SPICE-характеризації та розрахунку проміжних значень у Lua-модулі. Тому кожен із 25 елементів таблиці не обов’язково є результатом окремого прямого SPICE-моделювання. Генерація LIB-файла виконується за такою послідовністю:



Рис. 3.15 Алгоритм генерації LIB-файлу.

Такий підхід дозволяє відокремити трудомістке SPICE-моделювання від генерації бібліотечного опису. Після того як SPICE-результати отримані та внесені у таблиці характеризації, LIB-файли можуть бути сформовані автоматично для різних конфігурацій і PVT-кутів [1, 2, 29, 31]. Це особливо важливо для компілятора пам'яті, оскільки кількість можливих макросів визначається комбінацією кількості слів, розрядності, напруги живлення, температури та технологічного corner.

Реальні фрагменти вихідних файлів налаштування, вибору даних характеризації, формування таблиць та запису LIB-файла наведено в Додатку А із зазначенням джерел і діапазонів рядків**.**

Висновки до Розділу 3

1. Розроблено архітектуру параметричного компілятора двопортової пам’яті RFSRAM типу 1R1W, орієнтованого на генерацію макросів пам’яті з пріоритетом низького енергоспоживання для CMOS-технології 0.18 мкм із підтримкою технологічного масштабування до ефективної норми 0.152 мкм.
2. Визначено функціональну структуру згенерованого макросу RFSRAM, що включає масив 8T-комірок пам’яті, окремі порти читання і запису, адресні драйвери, дешифратори рядків, вхідні та вихідні буфери, блоки керування читанням і записом, а також схему контролю конфліктів для забезпечення коректної роботи при одночасному читанні та записі.
3. Сформовано SPICE-модель SRAM_MODEL, яка враховує структуру основних функціональних блоків макросу пам’яті, параметри конфігурації, окремі джерела живлення для вимірювання струмів по блоках, а також екстраговані паразитні опори та ємності топології.
4. Показано, що для достовірної оцінки часу доступу, затримок внутрішніх сигналів, динамічного струму та максимальної робочої частоти згенерованих макросів RFSRAM недостатньо schematic-level SPICE-моделювання; необхідним етапом є post-layout SPICE-характеризація з урахуванням extracted RC-параметрів.
5. На основі опублікованих автором результатів з rule-based parasitic RC extraction, Smart-CX/SmartRX та RC-merging обґрунтовано використання LVS-rule-driven підходу для формування SPICE-compatible netlist, який поєднує активні пристрої макросу пам’яті з паразитними опорами та ємностями міжз’єднань.
6. На прикладі макросів RF1 20x42 та RF2 56x24 продемонстровано застосування запропонованого маршруту до реальних конфігурацій пам’яті

з різними геометричними розмірами, aspect ratio, кількістю рядків і розрядністю слова.

7. За результатами SPICE-моделювання та розрахунку середнього динамічного струму показано, що block-level оптимізація у згенерованих макросах RFSRAM дозволяє суттєво зменшити струм споживання порівняно з референсною реалізацією пам’яті аналогічної конфігурації.
8. Запропоновано структуру програмного модуля генерації timing/power описів, який використовує результати SPICE-характеризації для формування бібліотечних даних макросу пам’яті, придатних для подальшого використання у маршрутах проєктування SoC.
9. Отримані результати підтверджують практичну придатність розробленого компілятора RFSRAM для автоматизованої генерації енергоефективних макросів пам’яті та створюють основу для експериментальної перевірки їх характеристик у Розділі 4.

РОЗДІЛ 4. Експериментальна та післятопологічна верифікація макросів RFSRAM

4.1 Частотна характеризація регістрових файлів, виготовлених у кремнії, з урахуванням впливу тестового оточення.

Одним із важливих етапів верифікації розробленого компілятора регістрових файлів є експериментальне визначення максимальної робочої частоти згенерованих реалізацій пам’яті [48].

Для цього було використано тестову структуру, виготовлену за 130-нм BiCMOS-технологією в межах MPW-проєкту (Multi-Project Wafer) на одній із

європейських фабрик. До складу тестової структури увійшли три конфігурації регістрових файлів, сформовані компілятором [48]:

- 256×72 біт = 18432 біт (*SC – Single Clock*),
- 84×24 біт = 2016 біт (*DC – Dual Clock*),
- 136×39 біт = 5304 біт (*CC – Conflict Controller*).

Топологію тестової структури наведено на рис. 4.1.

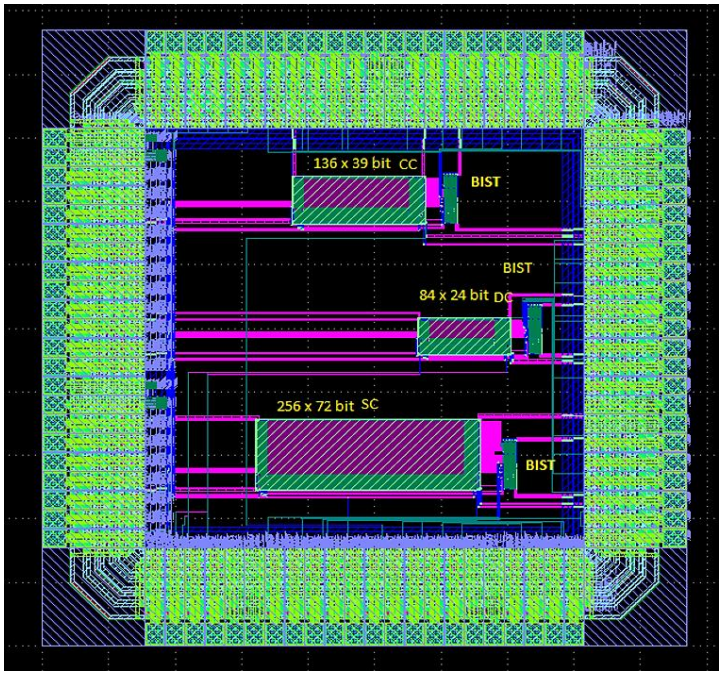


Рис. 4.1 - Тестова структура для проведення функціонального тестування регістрових файлів у кремнії

Максимальна робоча частота регістрового файла визначалася як найбільша частота, за якої пам'ять зберігає коректне функціонування.

Для визначення максимальної частоти, можна використовувати формулу:

$$F_{max} = \frac{1}{T_{min}} \tag{4.1}$$

де T_{min}— мінімальний період тактового сигналу, при якому забезпечується коректне функціонування пам’яті, а також обернене співвідношення

$$T = \frac{1}{F} \tag{4.2}$$

для переходу від частоти до періоду. Ці співвідношення використано для аналізу часових вікон при частотах 125 МГц і 100 МГц, коли змінюється допустиме вікно захоплення сигналу.

Для проведення вимірювань фабрикою було розроблено спеціальну інтерфейсну плату, структурну схему якої наведено на рис. 4.2. Сигнал тактування CLKR проходить через тестову структуру та виводиться назовні як CLKRout, що забезпечує компенсацію затримок вихідних контактів під час вимірювання граничної робочої частоти. Вихідні сигнали даних D0out–D7out і сигнал CLKRout після буферизації та фіксації тригерами надходять до тестового обладнання, яке виконує їх зчитування [48–50].

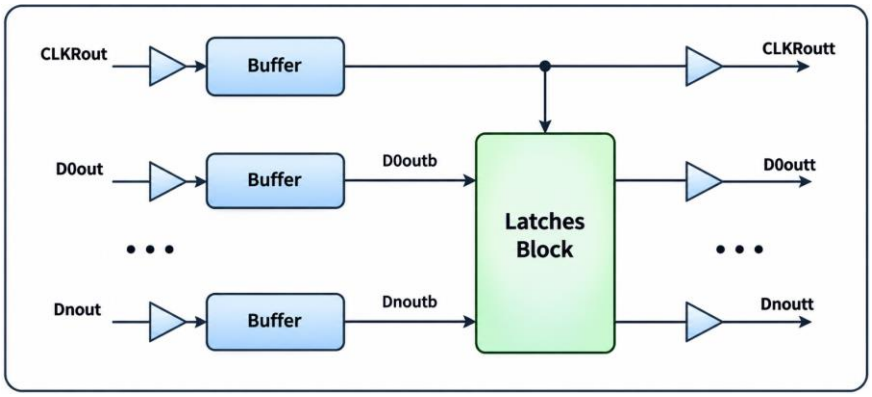


Рис. 4.2 - Структурна схема інтерфейсної плати тестера, використаної для вимірювання максимальної робочої частоти

Для експериментальної перевірки було задіяно 59 тестових кристалів, отриманих із чотирьох різних кремнієвих пластин у межах MPW-проєкту. Результати

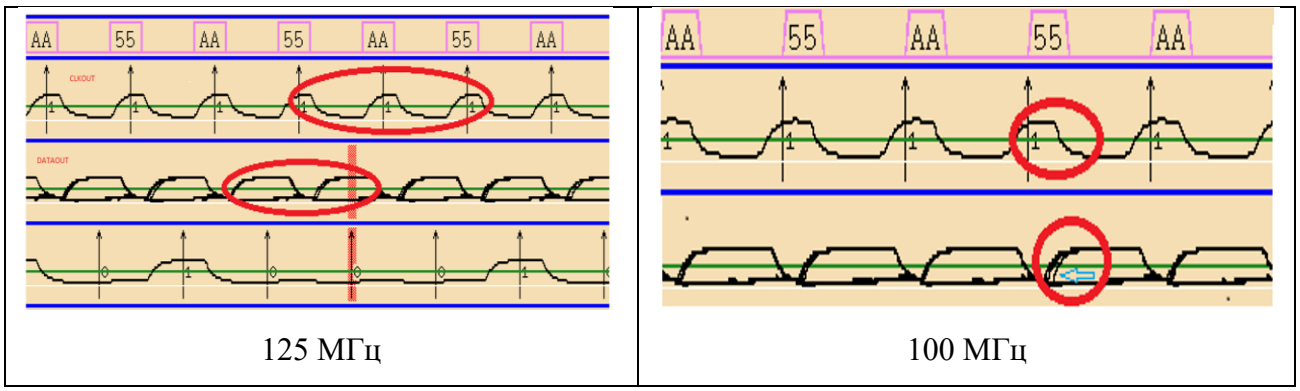
вимірювання максимальної робочої частоти регістрових файлів при напрузі живлення 1.2 В і температурі 25 °C наведено в табл. 4.1.

Таблиця 4.1

Результати виміру максимальної частоти при 1.2 В, 25 °C

	256x72 біт (SC)	**84x24 біт (DC)**	**136x39 біт (CC)**
Fmax, МГц	122.6	149.9	147.5
Fmax, МГц (BIST)	121.0	225.9	173.8

Як видно з табл. 4.1, максимальна експериментально виміряна робоча частота регістрового файла конфігурації 84×24 становила 149.9 МГц, тоді як результати SPICE-моделювання перевищували 400 МГц [48]. Для з'ясування причин такої розбіжності було проаналізовано часові діаграми сигналів CLKRoutt та D0outt–D7outt, наведені на рис. 4.3.



125 МГц 100 МГц

Рис. 4.3 - Приклад часових діаграм вихідних сигналів регістрового файлу при частоті, близькій до граничної

Аналіз показав, що сигнал CLKRoutt на виході тестової структури має несиметричну тривалість логічних рівнів: високий рівень є коротшим за низький. Аналогічне спотворення спостерігається і для вихідних сигналів даних D0outt–D7outt, для яких тривалість високого рівня зменшується приблизно на 2.4 нс. При

цьому вхідний тактовий сигнал CLKR, що надходить із тестового обладнання, має ідеальний duty factor 50 % та амплітуду 3.3 В [48, 51].

Як показано блакитною стрілкою на рис. 4.3, сигнал D7outt має найбільшу затримку поширення серед вихідних сигналів. Із підвищенням тактової частоти саме він першим виходить за межі інтервалу коректного зчитування, сформованого сигналом CLKRoutt, унаслідок чого вихідна защіпка інтерфейсної плати не встигає зафіксувати його логічний рівень. Це призводить до появи помилки читання, а за подальшого збільшення робочої частоти аналогічні помилки виникають і на інших вихідних сигналах.

Причиною виникнення описаного ефекту є різниця між затримками поширення фронтів наростання та спаду сигналів CLKRoutt і D0outt–D7outt на виходах буферів інтерфейсної плати. На рис. 4.4 наведено два характерні випадки [48, 51]. При однакових затримках фронтів помилки читання відсутні. Якщо ж затримка фронту наростання одного із сигналів є більшою, він не встигає бути зафіксованим вихідною защіпкою протягом активного інтервалу сигналу CLKRoutt, що призводить до помилки читання.

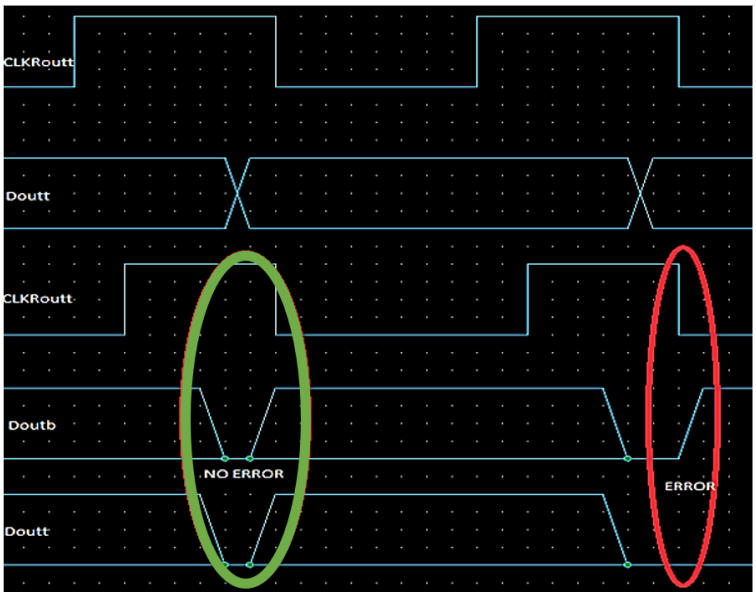


Рис. 4.4. Затримка між наростаючим та спадаючим фронтом сигналу CLK.

Джерелом цієї асиметрії є вхідні та вихідні комірки тестової структури (рис. 4.5). Вхідний контактний майданчик перетворює логічний рівень 3.3 В, що надходить від тестового обладнання, у внутрішній рівень живлення ядра 1.2 В, тоді як вихідний майданчик виконує зворотне перетворення для забезпечення сумісності із тестером [48–50].

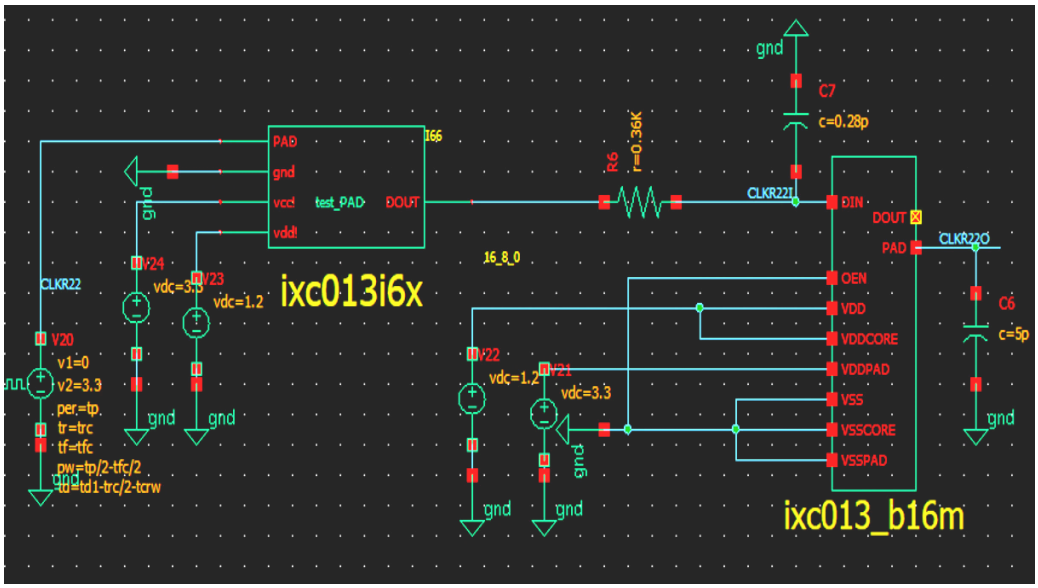


Рис. 4.5. Тестова схема для моделювання сигналу CLK.

Для аналізу цього ефекту було виконано екстракцію нетлістів вхідних і вихідних pad-комірок з урахуванням паразитних опорів і ємностей міжз'єднань. Подальше SPICE-моделювання для технологічного кута SS при напрузі живлення 1.2 В і температурі 25 °C підтвердило, що саме паразитні параметри сигнальних трактів спричиняють асиметрію часових характеристик, яка знижує максимальну частоту, виміряну на апаратному тестовому обладнанні [30, 31, 48].

Результати SPICE-моделювання наведено на рис. 4.6 для тактового сигналу CLKR із частотою 125 МГц. Отримані часові діаграми показують, що тривалість

високого рівня сигналу CLKRout становить близько 4 нс при періоді вхідного сигналу 8 нс.

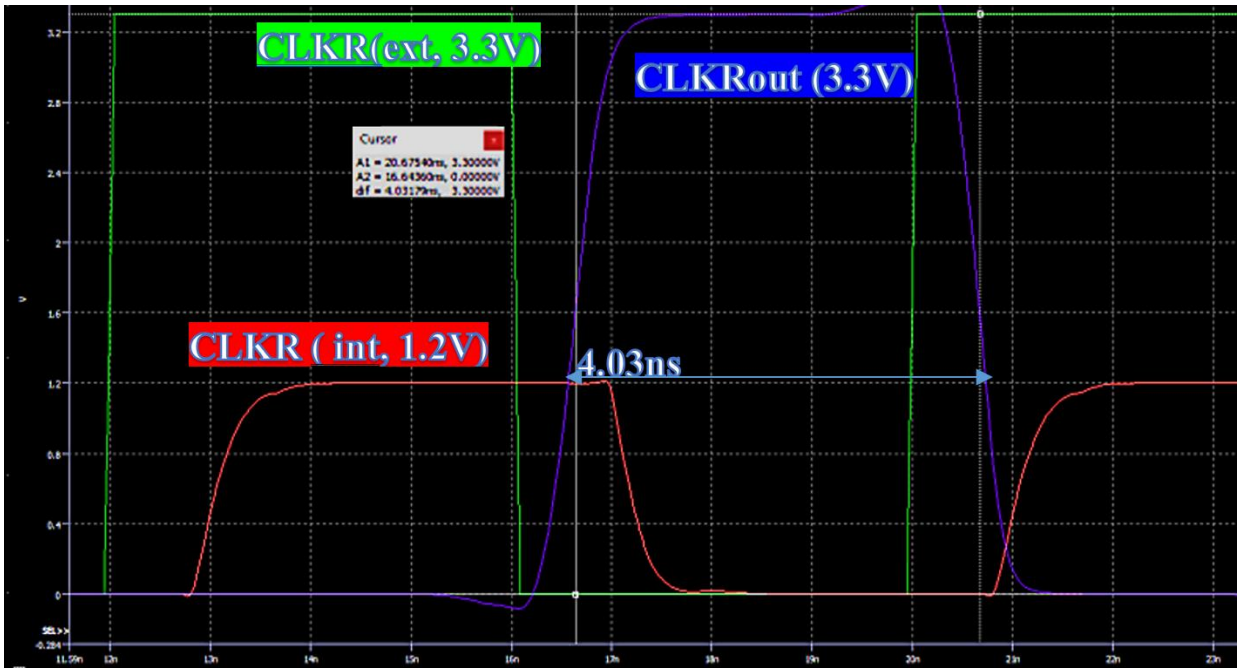


Рис. 4.6. Результати моделювання сигналу CLK з урахуванням паразитних опорів в I/O блоках та внутрішніх ланцюгах.

При цьому затримки фронтів наростання та спаду практично однакові, що свідчить про відсутність асиметрії у вхідних та вихідних контактних комірках тестової структури (блоки ixc013i6x та ixc013_b16m). Додатковий частотний аналіз показав, що вхідні контактні комірки забезпечують коректну роботу на частотах понад 400 МГц, тоді як швидкодія вихідних комірок обмежується приблизно 250 МГц, що визначає граничну частоту функціонування тестової структури для даного PVT-кута [48].

Для регістрового файла конфігурації 84×24 (DC) було виконано моделювання на частоті 200 МГц із використанням тестового шаблону типу checkerboard («шахова дошка») (рис. 4.7).

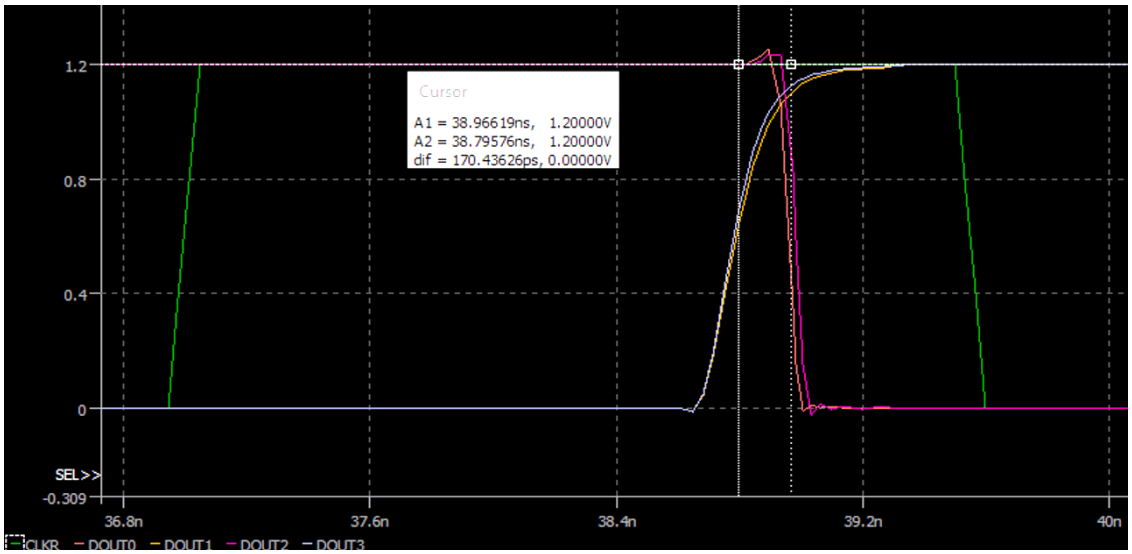


Рис. 4.7 – Результати моделювання сигналів D0out-D3out для інстансу RF 84x24 біт (DC) на частоті 200 МГц

Різниця між затримками фронтів сигналів не перевищувала 0.18 нс, що підтверджує: асиметрія часових характеристик виникає не в тестовій структурі, а в буферних каскадах інтерфейсної плати, які обробляють сигнали CLKRout і D0out–D7out [48, 51].

Результати моделювання регістрового файла 84×24 (DC) на частоті 400 МГц для найгіршого та найкращого технологічних випадків наведено на рис. 4.8.

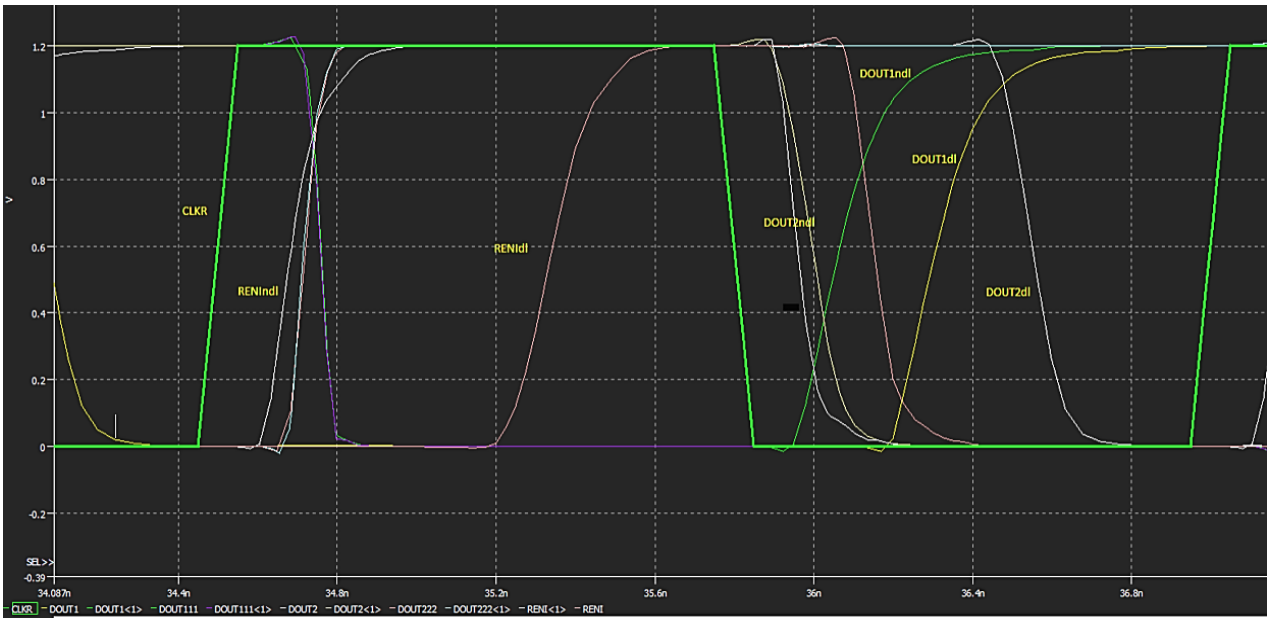


Рис. 4.8 – Результати моделювання RF 84x24 біт (DC) на частоті 400 МГц для “slow” (SS) та “fast”(FF) крайових кутів

Встановлено, що спадний фронт сигналу CLKR не відповідає завершенню операції читання через внутрішні затримки блока керування пам'яттю. Затримка появи даних на виході перевищує половину періоду тактового сигналу, однак функціональна працездатність регістрового файла не порушується. Отже, зчитування даних процесором має виконуватися за переднім фронтом сигналу CLKR, а весь попередній тактовий період слід розглядати як інтервал виконання операції читання.

Таким чином, проведене дослідження дозволило уточнити інтерпретацію результатів кремнієвих випробувань і показало, що виміряна максимальна частота визначається не лише внутрішньою швидкодією згенерованого регістрового файлу, але й обмеженнями тестового тракту, зокрема характеристиками вихідних pad-комірок та асиметрією затримок буферів інтерфейсної плати. Це має принципове значення для коректної верифікації

компілятора пам’яті, оскільки дозволяє відокремити обмеження власне архітектури пам’яті від обмежень засобів вимірювання [48–51].
Одночасно було встановлено, що вхідні pad-комірки здатні працювати на частотах понад 400 МГц, тоді як вихідні pad-комірки мають обмеження на рівні близько 250 МГц, що і формує фундаментальну межу достовірної зовнішньої частотної характеризації даної тестової структури у вибраному PVT-режимі [48].

За результатами дослідження було зроблено висновок, що при виконанні високочастотної характеризації регістрових файлів необхідно приділяти окрему увагу якості тактового сигналу на інтерфейсній платі, насамперед забезпеченню коефіцієнта заповнення, близького до 50 %, а також урахуванню реального частотного обмеження вихідних pad-комірок. Крім того, отримані результати підтвердили коректність застосованого підходу до побудови тестових структур і придатність розробленого компілятора до практичного використання, оскільки при тестуванні як зовнішнім обладнанням, так і засобами BIST досліджені зразки продемонстрували функціональну працездатність [48].

4.2. Вбудовані засоби самотестування BIST у компіляторі регістрових файлів

Однією з важливих практичних задач при створенні компіляторів пам’яті є забезпечення можливості тестування згенерованих макросів на рівні кристала. У сучасних телекомунікаційних та обчислювальних системах, де регістрові файли є вбудованими блоками SoC, застосування зовнішнього тестового обладнання є складним та економічно невигідним.

У зв’язку з цим доцільним є використання підходу **Built-In Self-Test (BIST)**, який дозволяє виконувати тестування безпосередньо на кристалі [56–58].

З метою спрощення інтеграції системи BIST мультиплексори даних були включені до складу схеми пам'яті, як показано на рис. 4.9. Таке рішення полегшує забезпечення часової збіжності при фізичному проєктуванні мікросхеми.

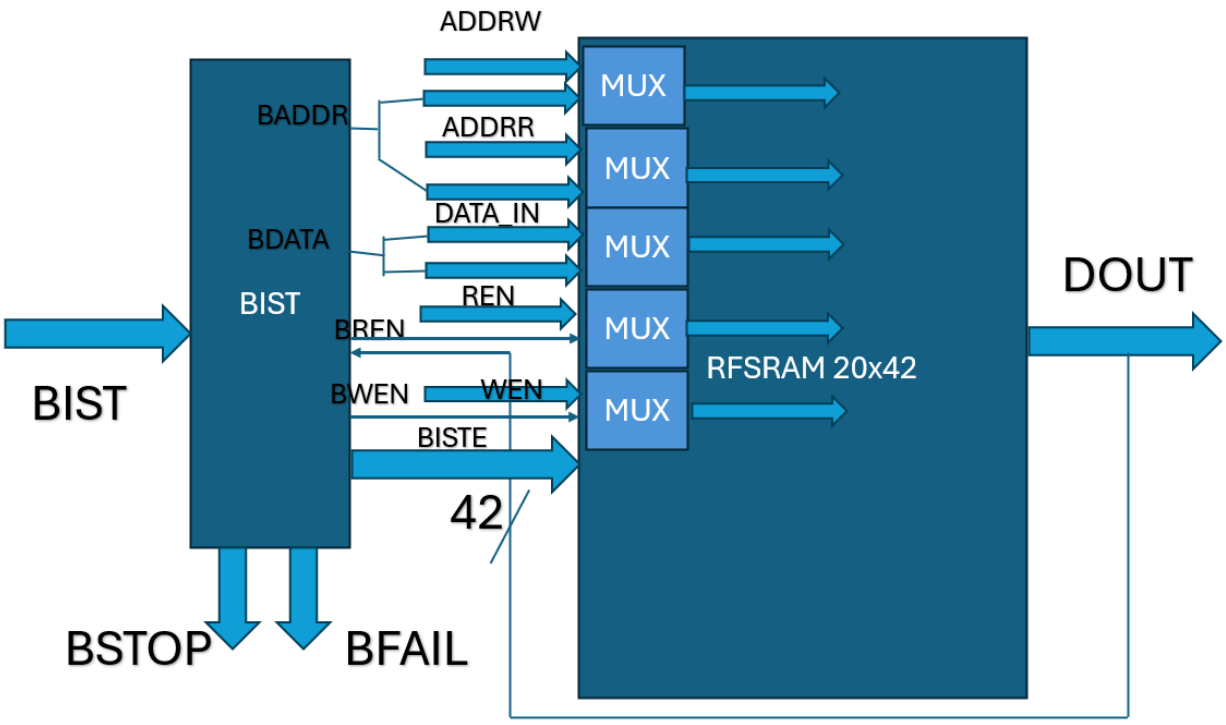


Рис. 4.9 – Структурна схема взаємодії блоків пам’яті та BIST на прикладі RF 20x42

У роботах автора було запропоновано інтеграцію BIST у структуру компілятора RFSRAM, що дозволяє автоматично генерувати тестову інфраструктуру разом із пам’яттю [56].

4.2.1 Архітектура компілятора з підтримкою BIST

Розроблений компілятор регістрових файлів забезпечує генерацію пам’яті з такими параметрами:

- кількість слів: від 16 до 256 (з кроком 4);
- довжина слова: від 8 до 72 біт;
- загальна кількість варіантів: понад 23 000 конфігурацій [1, 2, 56].

Компілятор підтримує три режими тактування:

1. один тактовий сигнал для двох портів;
2. незалежні тактові сигнали для читання та запису;
3. незалежні тактові сигнали з контролером конфліктів доступу.

Останній режим дозволяє уникнути втрати інформації шляхом буферизації операції запису у випадку одночасного доступу до однієї адреси.

4.2.2. Інтеграція BIST у структуру пам’яті

На відміну від традиційних підходів, у запропонованій архітектурі мультиплексори даних, адрес та керуючих сигналів інтегровані безпосередньо в структуру пам’яті [56–58].

Це дозволяє:

- перемикати пам’ять у тестовий режим без зовнішніх інтерфейсів;
- забезпечити узгодження часових параметрів;
- зменшити складність трасування сигналів на рівні кристала.

Основні блоки BIST включають:

- генератор тестових послідовностей;
- генератор адрес;
- компаратори даних;
- блок керування тестом.

Параметри конфігурації блока вбудованого самотестування (BIST) узгоджуються з параметрами регістрового файла та задають розрядність тестованого слова, кількість адрес і довжину тестової послідовності. У режимі тестування в регістровому файлі автоматично активуються мультиплексори порту запису, адресних сигналів, а також сигналів дозволу запису (WEN) і читання (REN). Завдяки цьому, як показано на рис. 4.9, на входи регістрового файла подаються сигнали, сформовані блоком BIST, що усуває необхідність використання

відповідних зовнішніх тестових сигналів. У випадку, коли операція читання ініціюється раніше за операцію запису, запис у поточному такті не виконується. Для такого запиту генерується сигнал зайнятості (BUSY), а вхідні дані та адреса запису тимчасово зберігаються у вхідних регістрах. Після завершення операції читання за цією адресою відкладена операція запису автоматично виконується.

Вхідні сигнали BIST заміщують стандартні сигнали пам’яті через вбудовані мультиплексори, а результати тестування виводяться через сигнали:

- **BFAIL** (BIST Fail) — індикація помилки;
- **BSTOP** (BIST Stop) — завершення тесту [56].

4.2.3 Алгоритм тестування

Для верифікації працездатності регістрових файлів із застосуванням BIST було спроєктовано тестову мікросхему, до складу якої входять три екземпляри регістрових файлів різної ємності та архітектури: 256×72 біт із одним тактовим сигналом (SC), 84×24 біт із двома незалежними тактовими сигналами (DC) та 136×39 біт із двома тактовими сигналами й контролером конфліктів доступу (CC).

Для перевірки працездатності пам’яті використовується послідовність із шести тестових шаблонів:

1. запис логічного «0»;
2. читання «0»;
3. запис логічної «1»;
4. читання «1»;
5. запис шахового шаблону (checkerboard);
6. читання шахового шаблону (checkerboard) [56–58].

Загальна кількість тактів тестування визначається як:

$$N_{cycles} = 6N + 2 \tag{4.3}$$

де N— кількість слів пам’яті [56].

4.2.4 Оцінка ефективності BIST

У результаті оптимізації площі кристала площу блока BIST було суттєво зменшено. Площа одного блока BIST становить лише 6720 мкм², що майже у вісім разів менше за площу одного екземпляра регістрового файла конфігурації 136×39 біт (58800 мкм²) [56]. Порівняльні розміри блоків пам’яті та BIST показані на Рис. 4.10:

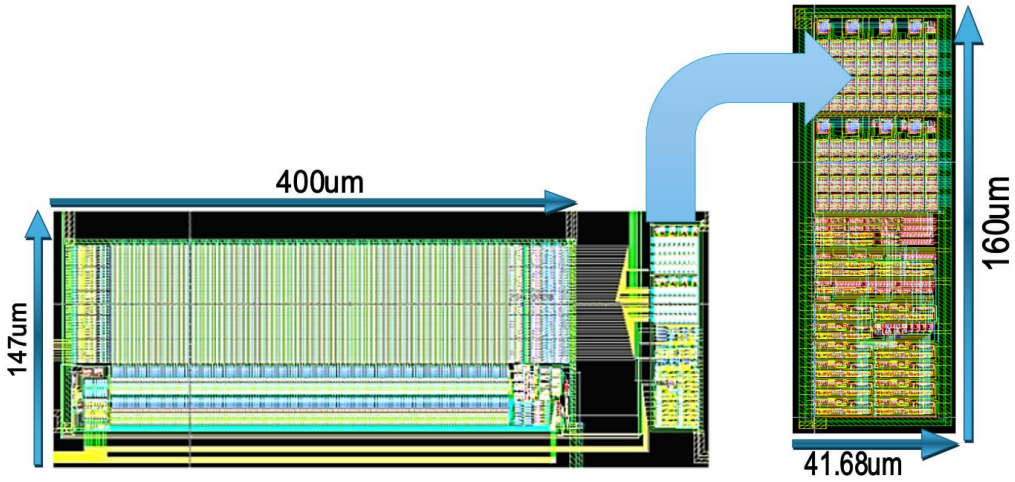


Рис. 4.10 – порівняльні розміри топології реалізації RF пам’яті та блоку BIST у конфігурації 136x39 біт

За відсутності вбудованих засобів самотестування (BIST) для тестування регістрового файла було б необхідно вивести на зовнішні контактні майданчики всі його сигнали. Зокрема, потрібно було б передбачити:

- 8 + 8 контактів для адресних шин портів запису та читання;
- 39 + 39 контактів для шин вхідних і вихідних даних;
- 3 додаткові контакти для керування мультиплексорами адрес і даних, блоками керування регістровим файлом та режимом тестування.

Таким чином, загальна кількість необхідних контактних майданчиків становила б 97, що є практично неприйнятним через значне збільшення розмірів кристала. Крім того, знадобилася б додаткова площа для трасування 97 сигнальних шин від контактних майданчиків до регістрового файла.

Працездатність запропонованої тестової структури була експериментально підтверджена після виготовлення мікросхеми. Тестування виконувалося двома незалежними методами: із застосуванням зовнішнього автоматизованого тестового обладнання та за допомогою вбудованої системи самотестування (BIST) [48, 56]. Після проходження послідовності з шести тестів на декількох екземплярах тестових мікросхем було виявлено дефекти у роботі пам’яті, які система BIST зафіксувала як помилки операцій читання та запису (рис. 4.11–4.12). Решта екземплярів регістрових файлів продемонстрували коректне функціонування у всіх режимах тестування (рис. 4.13).

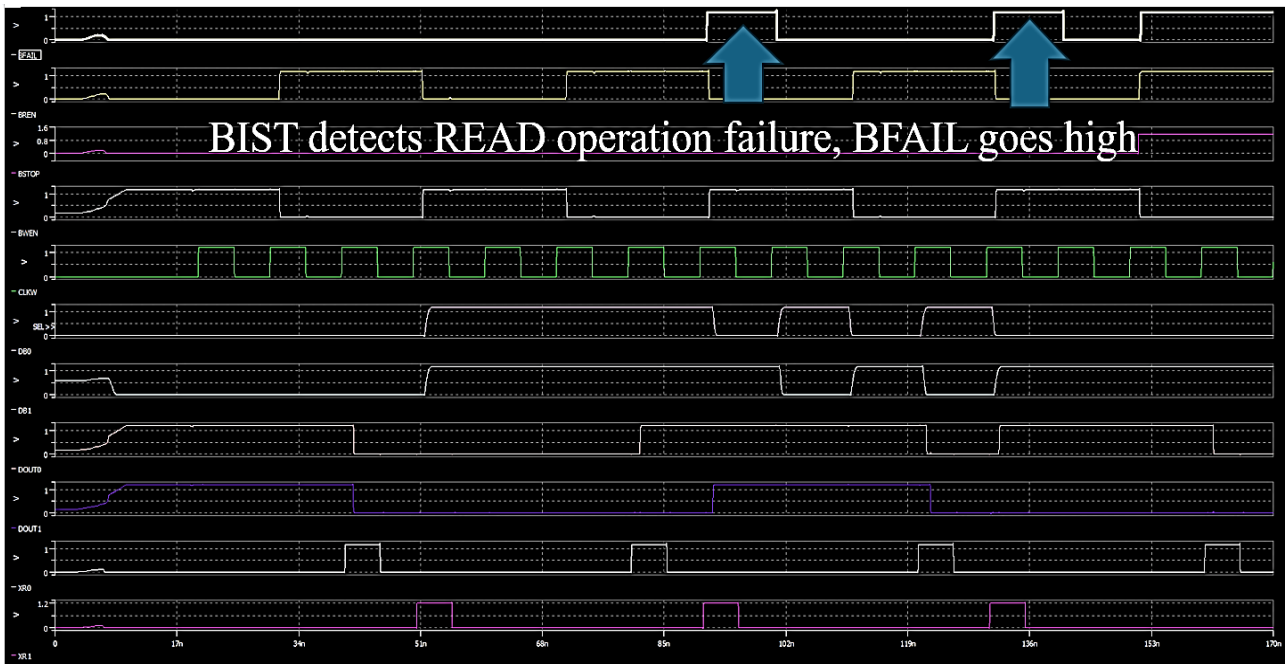


Рис. 4.11 – Приклад виявлення помилки операції читання за допомогою BIST



Рис. 4.12 – Приклад виявлення помилки операції запису за допомогою BIST

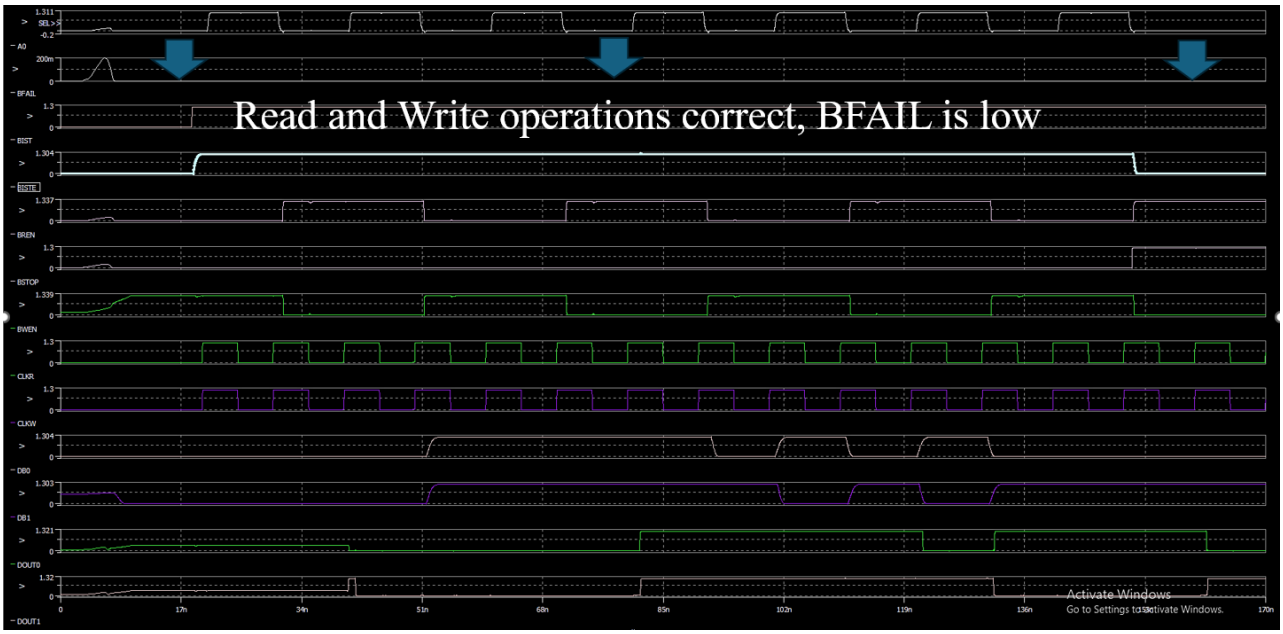


Рис. 4.13 – Результат успішного тестування за допомогою BIST – операції читання та запису виконані коректно

4.3. Post-layout SPICE-верифікація згенерованих макросів RFSRAM з урахуванням паразитних RC-параметрів

4.3.1. Мета та об’єкт post-layout верифікації

Метою даного підрозділу є експериментально-розрахункова перевірка впливу паразитних RC-параметрів топології на часові характеристики згенерованого макросу RFSRAM. На відміну від schematic-level SPICE-моделювання, post-layout SPICE-моделювання враховує паразитні опори та ємності міжз’єднань, контактів і переходів між шарами, що дозволяє уточнити час доступу, форму вихідних сигналів та граничні часові характеристики пам’яті [30, 31].

Як тестовий об’єкт використано макрос регістрового файла RF2 56×24, реалізований на основі 8T static memory cell [29, 31]. Схема містить 14938 транзисторів і була розроблена для технологічних норм 180 нм із коефіцієнтом масштабування $S = 0.84$. Така конфігурація є показовою для оцінювання впливу

паразитних параметрів, оскільки містить регулярний масив комірок, периферійні блоки керування, адресації та вихідні буфери.

Для відокремлення внеску різних типів паразитних параметрів було виконано моделювання декількох варіантів нетліста, що дозволяє порівняти ідеалізований schematic-level випадок із моделями, які враховують лише паразитні опори, лише паразитні ємності або їх спільний вплив.

4.3.2. Формування netT, netR, netC та netRC

На початковому етапі екстракції формуються окремі нетлісти:

- **netT** — базовий еталонний нетліст без паразитної екстракції, який використовується для порівняння.
- **netC** — нетліст, що містить екстраговані паразитні ємності;
- **netR** — нетліст, що містить екстраговані паразитні опори;
- **netRC** – комбінований нетліст, що містить об'єднані паразитні опори та ємності, отриманий об'єднанням netR і netC, придатний для post-layout SPICE-моделювання [31, 33, 34, 40–42].

Об'єднання, тобто формування комбінованого RC-нетліста, виконується за допомогою окремої утиліти SmartMerger.

На відміну від netC, який зберігає кількість вузлів базового нетліста, netR містить додаткові внутрішні вузли, оскільки провідники розбиваються на резистивні сегменти. Тому пряме додавання ємностей із netC до netR неможливе без процедури зіставлення вузлів і активних приладів [31]. Для цього використовується coordinate-based matching активних елементів і подальше перенесення ємностей у резистивну мережу. При цьому обидва нетлісти — netC і netR — містять ті самі активні прилади, що і базовий нетліст, зокрема MOSFETs чи біполярні транзистори.

Суть генерації комбінованого нетліста netRC полягає в такій послідовності дій. Спочатку визначаються транзистори, які мають однакові вузли підключення активного приладу, тобто затвори, стоки, витоки та інші термінали. Для кожного такого вузла в netC визначається кількість N підключень терміналів приладу до відповідного вузла. Далі за нетлістом netC визначаються паразитні ємності C_{ij}, підключені між цими вузлами. До них можуть належати як ємності від глобальних вузлів до VCC або GND, так і міжвузлові ємності між внутрішніми вузлами схеми. Після цього встановлюється відповідність між вузлами підключення i та j активних елементів у netC і netR. Відповідні ємності C_{ij} переносяться у netR шляхом підключення частин ємності: $\frac{C_{ij}}{N}$ до відповідних терміналів приладів. Таким чином, вихідні ємності фактично замінюються еквівалентними π-models, як показано на рис. 4.14 [31, 33, 34]. При цьому внутрішні вузли резисторів залишаються незмінними.

У результаті формується кінцевий комбінований нетліст netRC, який містить як активні компоненти, так і паразитні RC-елементи, забезпечуючи повне представлення схеми, придатне для подальшого схемотехнічного моделювання [31].

Таким чином, π-model representation дозволяє підключити паразитні ємності до відповідних вузлів резистивного нетліста без руйнування структури netR. Це забезпечує формування компактного simulation-ready netRC, який містить як активні прилади, так і паразитні R/C-елементи [31, 33, 34].

Принцип такого перетворення показано на рис. 4.14:

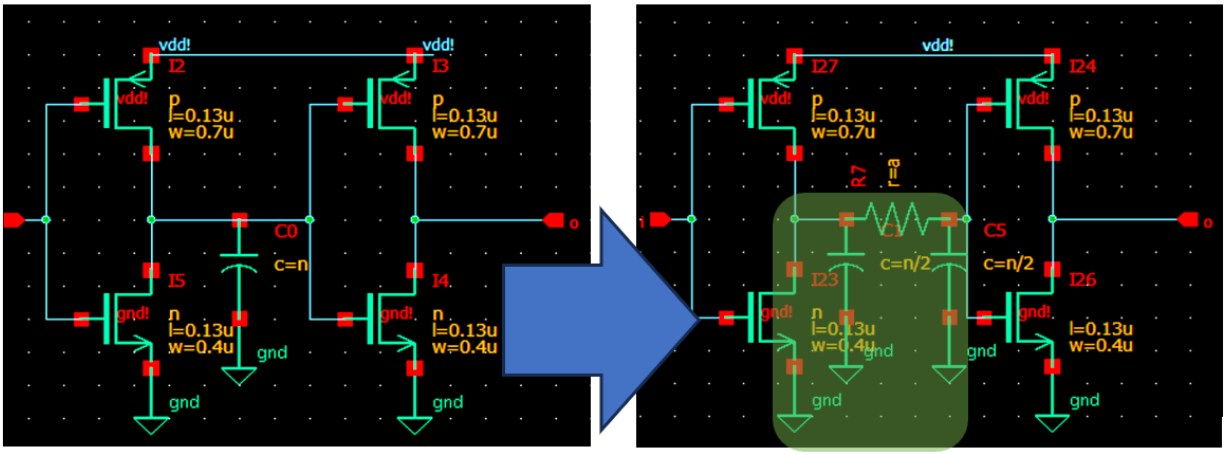


Рис. 4.14. Схематичне представлення об’єднання R- та C-нетлістів у комбінований RC-нетліст на основі π-моделі.

Точність комбінованого нетліста, сформованого з використанням π-models, може бути оцінена шляхом порівняння результатів моделювання для різних варіантів урахування паразитних параметрів [31, 33, 34, 40–42].

4.3.3. Порівняння T/R/C/RC-моделей

На рис. 4.15 наведено результати моделювання затримки вихідного сигналу для тестового регістрового файлу за таких умов:

- **T** — схема без урахування паразитних параметрів;
- **R** — схема з урахуванням лише паразитних опорів;
- **C** — схема з урахуванням лише паразитних ємностей;
- **RC** — схема з об’єднаними паразитними опорами та ємностями.

Точність формування комбінованого нетліста netRC оцінювалася не за повним абсолютним часом доступу, а за відносним внеском похибки RC-об’єднання у загальний приріст затримки, спричинений переходом від базового нетліста без паразитних параметрів netT до комбінованого RC-нетліста netRC. Цей приріст визначається за формулою (4.4):

$$\Delta T_{T \to RC} = T_{acc}^{RC} - T_{acc}^{T} \qquad (4.4)$$

де T_{acc}^{T}— час доступу для базового нетліста без урахування паразитних параметрів, а T_{acc}^{RC}— час доступу для комбінованого нетліста з урахуванням паразитних R- та C-параметрів.

Похибка RC-об'єднання визначається як відносна різниця між інкрементальними внесками R- та RC-моделей, нормована на повний приріст затримки $\Delta T_{T\to RC}$, розрахований за формулою (4.4). Відповідна похибка визначається за формулою (4.5):

$$\delta_{RC} = \frac{|\Delta T_{RC}^{inc} - \Delta T_{R}^{inc}|}{\Delta T_{T\to RC}} \times 100\ \% \qquad (4.5)$$

де ΔT_{R}^{inc}та ΔT_{RC}^{inc}— інкрементальні внески затримки, визначені за результатами порівняння R- та RC-моделей на рис. 4.15. Для результатів, наведених на рис. 4.15, інкрементальний внесок R-моделі становить $\Delta T_{R}^{inc} = 25.97$ пс, а відповідний внесок RC-моделі — $\Delta T_{RC}^{inc} = 30.4$ пс. Загальний приріст затримки при переході від netT до netRC, визначений за графіком, становить:

$$\Delta T_{T\to RC} = 1.26 \text{ нс} + 25.97 \text{ пс} + 30.4 \text{ пс} = 1.316 \text{ нс}.$$

Тоді похибка RC-об'єднань становить δ_{RC} = 0.34 %. Оскільки ця величина є меншою за 1 % від загального приросту затримки $\Delta T_{T\to RC}$, похибка формування комбінованого netRC може вважатися прийнятною для цільового post-layout часового аналізу [31, 33, 34].

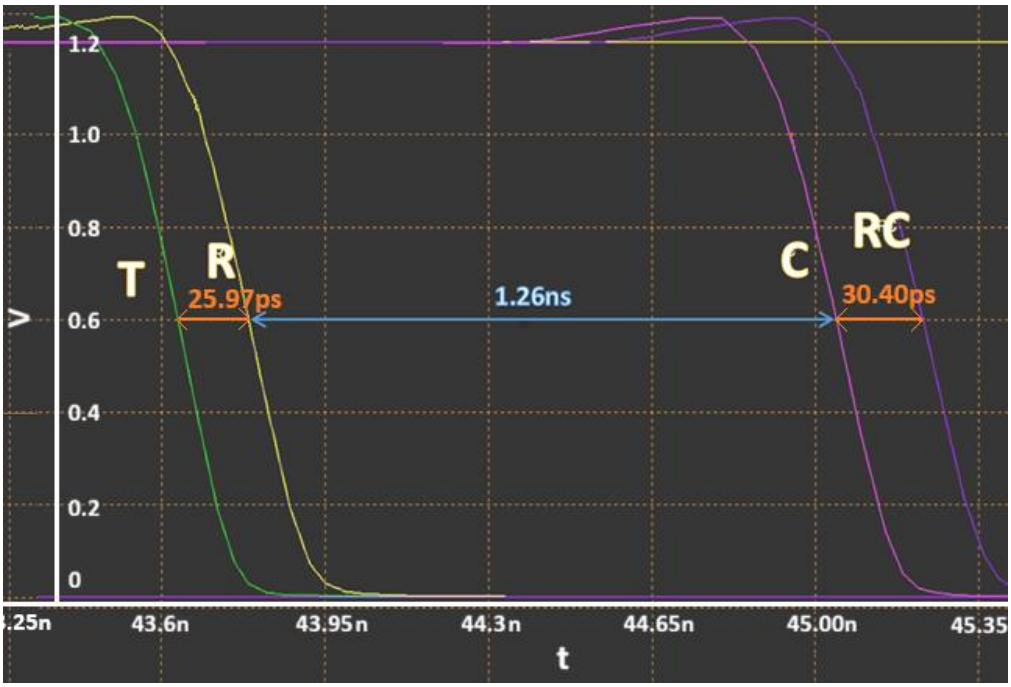


Рис. 4.15 – Затримка вихідного сигналу інстансу RF2 56x24 для нетлістів T, R, C, RC при PVT умовах моделювання: (TT, 1.2 В, 25 °C).

Результати показують, що схема без паразитних параметрів дає найбільш оптимістичну оцінку затримки, тоді як урахування C та RC призводить до помітного зміщення фронтів вихідного сигналу. Це підтверджує, що schematic-only SPICE-модель не є достатньою для фінальної оцінки швидкодії макросу пам'яті [30, 31].

4.3.4 Аналіз впливу scaling та process-parameter variations

Додатково для оцінювання впливу масштабування технології на паразитні параметри було використано регістровий файл конфігурації RF2 56×24, реалізований на основі 8T static memory cell. Схема містить 14938 транзисторів і була розроблена для технологічних норм 180 нм із коефіцієнтом масштабування: $S = 0.84$. Моделювання виконувалося за таких PVT-умовах:

$$Process = SS, Voltage = 1\ \text{В}, Temperature = 27°C$$

Для проведення порівняльного аналізу було екстраговано та підготовлено три нетлісти [31]:

- **spicenl_56x24_T.lib** — базовий нетліст з активними приладами без урахування паразитних параметрів;
- **spicenl_56x24_max_084.lib** — нетліст із паразитними ємностями, розрахованими для випадку максимальних товщин діелектриків і мінімальних товщин провідників;
- **spicenl_56x24_min_084.lib** — нетліст із паразитними ємностями, розрахованими для випадку мінімальних товщин діелектриків і максимальних товщин провідників.

Порівняння чотирьох варіантів моделювання дозволяє кількісно оцінити окремий внесок паразитних опорів, паразитних ємностей і комбінованої RC-моделі у затримку вихідного сигналу.

До обох екстрагованих нетлістів із паразитними параметрами було застосовано коефіцієнт масштабування 0.84. Результати моделювання наведено в табл. 4.2 та на рис. 4.16, де зчитування даних виконується за спадним фронтом тактового сигналу [31]. Різниця часу доступу до вихідних даних T_{acc} між схемами без урахування паразитних параметрів для випадків S = 0.84 (крива A) та S = 1 (крива B) становить: ΔT_{acc}= 0.93 нс.

Для випадку з урахуванням паразитних параметрів результати моделювання для масштабованих нетлістів (S = 0.84) представлені кривими A, D та E. Різниця часу доступу T_{acc} між мінімальним (крива D) та максимальним (крива E) варіантами spicenl.lib-моделей із паразитними параметрами становить лише 0.46 нс.

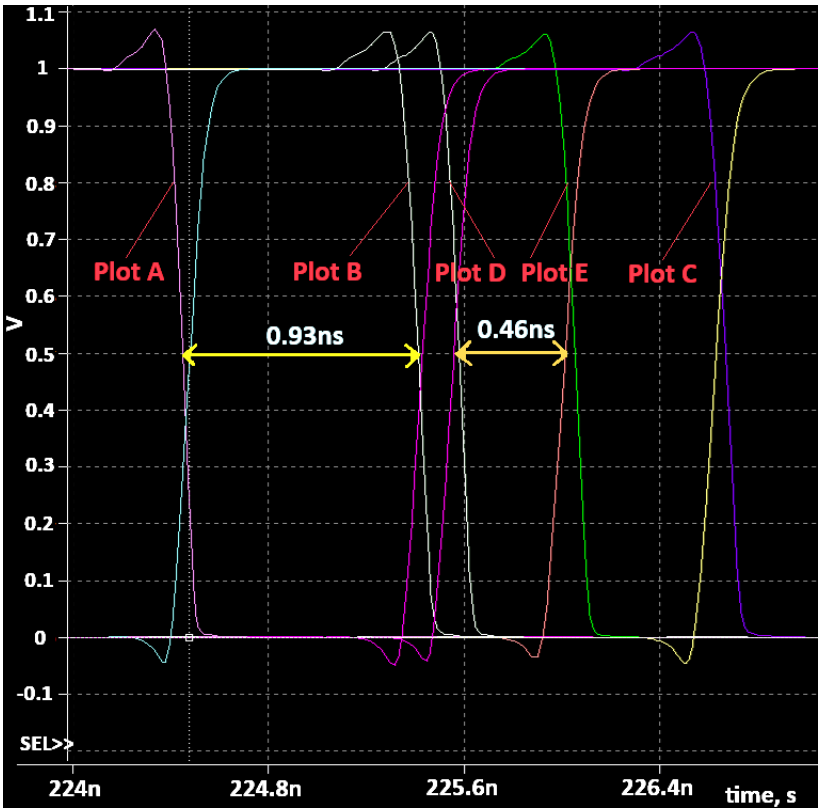


Рис. 4.16. Результати моделювання часу доступу T_{acc} для інстансу RF 56×24, графіки Plots A–E при PVT умовах: process = SS, VDD = 1.0 В, T = 27 °C.

Таблиця 4.2

Результати post-layout SPICE-моделювання часу доступу T_{acc} для RF2 56×24 при різних варіантах масштабування та паразитної екстракції

Крива	Найменування файлу екстрагованого нетлісту	Коеф. масштабування S	T_{acc}, нс
A	spicenl_56x24_T.lib	0.84	4.85
B	spicenl_56x24_T.lib	1	5.78
C	spicenl_56x24_max_084.lib	1	6.9
D	spicenl_56x24_max_084.lib	0.84	6.03
E	spicenl_56x24_min_084.lib	0.84	6.49

Порівняння кривих A і B демонструє вплив геометричного масштабування без урахування паразитних параметрів. Порівняння кривих D і E показує чутливість

часу доступу до граничних значень технологічних параметрів міжшарових діелектриків і провідників при S = 0.84.

4.3.5. Порівняння Tacc без/з урахуванням паразитних параметрів у PVT-умовах

Результати моделювання операцій зчитування даних зведено в табл. 4.3 окремо для переходів із логічної одиниці в нуль (1→0) та з нуля в одиницю (0→1).

Таблиця 4.3

Порівняння часу доступу RF2 56×24 без урахування та з урахуванням паразитних параметрів для різних PVT-умов

	Tacc,нс для CL=0.01 пФ, S=0.84				
Граничні умови (PVT)	Без урахування паразитних параметрів T_{acc}^{base}, нс		З урахуванням паразитних параметрів $T_{acc}^{parasitic}$, нс		ΔT_{acc} %
	1->0	0->1	1->0	0->1	max
ss/1.1 В/-40°C	3.54	3.45	4.92	6.075	76.0
ss/1.1 В /85°C	3.96	3.83	5.53	6.92	44.5
tt/1.2 В/27°C	2.41	2.33	3.33	3.93	40.7
ff/1.3 В/-40°C	1.48	1.43	2.12	2.51	43.0
ff/1.3 В/85°C	1.75	1.84	2.61	2.91	36.7
ss/1.45 В/125°C	2.41	2.47	3.325	4.12	40.0

Як видно з табл. 4.3, урахування паразитних параметрів призводить до збільшення часу доступу в усіх досліджених PVT-умовах. Найбільший відносний приріст спостерігається для переходу 0→1 у corner ss/1.1 В/–40 °C і становить 76 %. Це підтверджує, що паразитні параметри топології є одним із визначальних факторів при оцінюванні швидкодії згенерованого макросу RFSRAM.

Відносна зміна часу доступу внаслідок урахування паразитних параметрів розраховується за формулою (4.6).

$$\Delta T_{acc} = \frac{T_{acc}^{parasitic} - T_{acc}^{base}}{T_{acc}^{base}} \times 100\ \% \quad (4.6)$$

де $T_{acc}^{parasitic}$ та T_{acc}^{base} – значення часу доступу для одного й того самого переходу вихідного сигналу, відповідно з урахуванням та без урахування паразитних параметрів. У табл. 4.3 наведено максимальне значення ΔT_{acc} для переходів 1→0 та 0→1.

Отримані результати демонструють суттєвий вплив паразитних параметрів на T_{acc} регістрового файлу. Зокрема, врахування паразитних ємностей і технологічних варіацій міжшарових діелектриків та провідників призводить до помітної зміни затримки вихідного сигналу, що підтверджує необхідність використання post-layout моделювання для коректної оцінки швидкодії пам'яті.

Екземпляр регістрового файла 56×24 був виготовлений і протестований у кремнії, що підтвердило практичну працездатність згенерованого макросу пам'яті [48]. У поєднанні з результатами post-layout SPICE-моделювання це підтверджує придатність запропонованого маршруту для оцінювання часових характеристик RFSRAM з урахуванням паразитних параметрів.

Проведене post-layout SPICE-моделювання показало, що врахування паразитних RC-параметрів суттєво змінює оцінку часу доступу макросу пам'яті. Використання окремо сформованих netR і netC та їх об'єднання в netRC через π-model дозволяє отримати компактний simulation-ready netlist без втрати timing accuracy в межах допустимої похибки. Таким чином, результати у табл. 4.3 підтверджують необхідність включення post-layout RC-характеризації до маршруту верифікації компілятора RFSRAM [30, 31].

Висновки до розділу 4

1. Проведено експериментальну частотну характеризацію тестових кристалів із трьома конфігураціями регістрових файлів, згенерованих компілятором RFSRAM: 256×72 біт SC, 84×24 біт DC та 136×39 біт CC. Отримані результати підтвердили функціональну працездатність згенерованих макросів пам’яті у кремнії.
2. Встановлено, що експериментально виміряна максимальна частота роботи регістрового файла залежить не лише від внутрішньої швидкодії пам’яті, але й від параметрів тестового тракту, зокрема вихідних pad-комірок, буферів інтерфейсної плати та асиметрії затримок фронтів сигналів.
3. На основі аналізу часових діаграм і SPICE-моделювання вхідних та вихідних трактів показано, що обмеження частотної характеризації тестової структури пов’язане з часовими характеристиками зовнішнього тестового оточення, тоді як самі макроси пам’яті зберігають функціональну працездатність у досліджених режимах.
4. Запропоновано та реалізовано інтеграцію BIST у структуру компілятора RFSRAM, що дозволяє автоматично генерувати тестову інфраструктуру разом із макросом пам’яті та зменшує потребу у великій кількості зовнішніх контактних майданчиків для тестування.
5. Показано, що використання BIST дозволяє виконувати функціональну перевірку операцій читання та запису безпосередньо на кристалі, виявляти помилки пам’яті через сигнал BFAIL та підтверджувати коректне проходження тесту через завершення тестової послідовності.
6. Виконано post-layout SPICE-моделювання регістрового файла RF 56×24 з урахуванням паразитних параметрів топології. Показано, що врахування extracted RC-параметрів суттєво впливає на час доступу та форму вихідних сигналів, тому schematic-level SPICE-моделювання є недостатнім для фінальної оцінки швидкодії макросу пам’яті.

7. На основі порівняння netT, netR, netC та комбінованого netRC підтверджено доцільність використання RC-merging flow із π-model representation для формування компактного simulation-ready нетліста, придатного для post-layout timing analysis.
8. Отримані результати підтверджують практичну придатність розробленого компілятора RFSRAM для генерації працездатних макросів пам’яті, їх функціональної перевірки, BIST-тестування та післятопологічної SPICE-характеризації з урахуванням паразитних параметрів.

ВИСНОВКИ

У даній дисертаційній роботі вирішено науково-технічну задачу розроблення та верифікації методів автоматизованого проєктування двопортової пам’яті регістрового типу RFSRAM на основі 8T SRAM-комірки з пріоритетом низького енергоспоживання, забезпеченням статичної стійкості при знижених напругах живлення та урахуванням паразитних RC-параметрів топології під час post-layout SPICE-моделювання.

1. Проведено аналіз сучасних підходів до проєктування енергоефективної SRAM/RFSRAM-пам’яті та встановлено, що для телекомунікаційних, радіотехнічних, сенсорних, автономних та оборонних систем критичними параметрами є енергоспоживання, час доступу, стабільність збереження даних і працездатність при знижених напругах живлення. Показано, що зниження напруги живлення є ефективним способом зменшення динамічної потужності, однак воно супроводжується деградацією

SNM/WNM, зниженням швидкодії та підвищенням чутливості до технологічних варіацій і паразитних параметрів топології.

2. Обґрунтовано доцільність використання 8T-комірки з розділеними портами читання і запису як базового елемента двопортової RFSRAM-пам’яті з пріоритетом низького енергоспоживання. На відміну від класичної 6T SRAM-комірки, така структура дозволяє зменшити конфлікт між read stability та write-ability, що є особливо важливим для режимів зниженої напруги живлення.
3. Запропоновано SNM-орієнтований підхід до вибору параметрів 8T-комірки RFSRAM, який базується на аналізі butterfly-кривих, критерію граничного стану статичної стійкості та урахуванні статистичного розкиду порогових напруг MOS-транзисторів. Показано, що використання лише номінального значення SNM є недостатнім для оцінювання стабільності low-voltage пам’яті, оскільки воно не враховує вплив технологічних варіацій.
4. Розроблено методику визначення допустимих геометричних параметрів транзисторів 8T-комірки на основі 6σ-моделі розкиду порогових напруг та butterfly-критерію граничної стійкості. Встановлено, що найбільший вплив на стабільність комірки у режимі читання мають n-канальні транзистори інверторів комірки та транзистори порту читання. Показано, що цілеспрямована оптимізація цих транзисторів дозволяє зменшити мінімальну робочу напругу живлення за прийнятного збільшення площі комірки.
5. Виконано експериментальний статистичний аналіз SNM-розподілу тестових 8T dual-port SRAM-комірок і підтверджено доцільність використання нормального закону розподілу та mismatch-based підходу для прогнозування стабільності low-power пам’яті. Отримані результати підтверджують фізичну обґрунтованість застосування статистичного

підходу до оцінювання стабільності 8T SRAM/RFSRAM-комірок при знижених напругах живлення.

6. Розроблено архітектуру параметричного компілятора двопортової пам’яті RFSRAM типу 1R1W, орієнтованого на автоматизовану генерацію макросів пам’яті з пріоритетом низького енергоспоживання. Визначено функціональну структуру згенерованого макросу, що включає масив 8T-комірок пам’яті, окремі порти читання і запису, адресні драйвери, дешифратори рядків, вхідні та вихідні буфери, блоки керування читанням і записом, а також схему контролю конфліктів.
7. Розроблено програмну архітектуру генератора макросів RFSRAM, що включає формування матриці комірок пам’яті, збірку верхнього рівня макросу, генерацію схемного LVS-нетліста та формування timing/power LIB-файлів на основі результатів SPICE-характеризації. Запропонований підхід дозволяє автоматизувати створення макросів пам’яті різної глибини, розрядності та конфігурації, а також забезпечити їхню придатність для подальшої інтеграції у маршрути проєктування SoC.
8. Сформовано SPICE-модель макросу пам’яті, яка враховує структуру основних функціональних блоків, параметри конфігурації, окремі джерела живлення для вимірювання струмів по блоках, а також екстраговані паразитні опори та ємності топології. На прикладі макросів RF1 20×42 та RF2 56×24 продемонстровано застосування запропонованого маршруту до реальних конфігурацій пам’яті з різними геометричними розмірами, aspect ratio, кількістю рядків і розрядністю слова.
9. Обґрунтовано необхідність post-layout SPICE-характеризації з урахуванням паразитних RC-параметрів для достовірної оцінки часу доступу, затримок внутрішніх сигналів, динамічного струму та максимальної робочої частоти згенерованих макросів RFSRAM. Показано,

що schematic-level SPICE-моделювання є недостатнім для фінальної оцінки швидкодії макросу пам’яті, оскільки не враховує вплив паразитних опорів і ємностей міжз’єднань.

10. На основі використання rule-based parasitic RC extraction, формування окремих netR і netC та їх об’єднання у комбінований netRC із застосуванням π-model representation підтверджено доцільність RC-merging flow для створення компактного simulation-ready нетліста, придатного для post-layout timing analysis. Показано, що урахування extracted RC-параметрів суттєво впливає на оцінку часу доступу та форму вихідних сигналів макросу пам’яті.
11. Проведено експериментальну частотну характеризацію тестових кристалів із трьома конфігураціями регістрових файлів, згенерованих компілятором RFSRAM: 256×72 біт SC, 84×24 біт DC та 136×39 біт CC. Отримані результати підтвердили функціональну працездатність згенерованих макросів пам’яті у кремнії. Встановлено, що експериментально виміряна максимальна частота залежить не лише від внутрішньої швидкодії пам’яті, але й від параметрів тестового тракту, зокрема вихідних pad-комірок, буферів інтерфейсної плати та асиметрії затримок фронтів сигналів.
12. Запропоновано та реалізовано інтеграцію блока вбудованого самотестування BIST у структуру компілятора RFSRAM. Показано, що використання BIST дозволяє виконувати функціональну перевірку операцій читання та запису безпосередньо на кристалі, зменшує залежність від зовнішнього тестового обладнання та підтверджує придатність згенерованих макросів пам’яті до практичного використання.
13. Практичне значення отриманих результатів полягає у можливості використання запропонованих методів для створення енергоефективних

IP-блоків двопортової RFSRAM-пам’яті для CMOS-технологій 180 нм, масштабованої 152 нм, 130 нм та інших planar process nodes після відповідної технологічної перекваліфікації. Розроблений підхід може бути застосований при створенні систем на кристалі для телекомунікаційного обладнання, цифрової обробки сигналів, радарних комплексів, систем радіоелектронної боротьби, безпілотних платформ, сенсорних мереж та IoT-пристроїв.

14. Отримані результати підтверджують доцільність поєднання SNM/WNM-aware проєктування 8T-комірки, статистичного оцінювання стабільності, параметричної генерації макросів пам’яті, post-layout SPICE-характеризації з урахуванням extracted RC-параметрів та експериментальної перевірки у кремнії в єдиному маршруті автоматизованого проєктування енергоефективної RFSRAM-пам’яті.

Список використаних джерел

[1] Груданов М. Б., Дудник О. М., Рубанець М. М., Груданов О. М. Генератор топології статичних оперативних пристроїв. *Електроніка та зв’язок*. 2008. № 1. С. 21–25.

[2] Sumanth K., Shourya G., Calhoun B. H. MemGen: An Open-Source Framework for Autonomous Generation of Memory Macros. In: *Proceedings of the IEEE Custom Integrated Circuits Conference (CICC)*. 2021.

[3] Seevinck E., List F. J., Lohstroh J. Static-Noise Margin Analysis of MOS SRAM Cells. *IEEE Journal of Solid-State Circuits*. 1987. Vol. 22, No. 5. P. 748–754. DOI: 10.1109/JSSC.1987.1052809.

[4] Bhavnagarwala A. J., Tang X., Meindl J. D. The Impact of Intrinsic Device Fluctuations on CMOS SRAM Cell Stability. *IEEE Journal of Solid-State Circuits*. 2001. Vol. 36, No. 4. P. 658–665. DOI: 10.1109/4.913744.

[5] Pelgrom M. J. M., Duinmaijer A. C. J., Welbers A. P. G. Matching Properties of MOS Transistors. *IEEE Journal of Solid-State Circuits*. 1989. Vol. 24, No. 5. P. 1433–1439. DOI: 10.1109/JSSC.1989.572629.

[6] Calhoun B. H., Chandrakasan A. P. Static Noise Margin Variation for Sub-threshold SRAM in 65-nm CMOS. *IEEE Journal of Solid-State Circuits*. 2006. Vol. 41, No. 7. P. 1673–1679. DOI: 10.1109/JSSC.2006.873215.

[7] Wang J., Nalam S., Calhoun B. H. Analyzing Static and Dynamic Write Margin for Nanometer SRAMs. In: *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED)*. 2008. P. 129. DOI: 10.1145/1393921.1393954.

[8] Maxim A., Gheorghe M. A Novel Physical Based Model of Deep-Submicron CMOS Transistors Mismatch for Monte Carlo SPICE Simulation. In: *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*. 2001. Vol. 5. P. 511–514. DOI: 10.1109/ISCAS.2001.922097.

[9] Maxim A. Physically-Based Matching Model for Deep-Submicron MOS Transistor. ESSDERC, 2002.

[10] Guo Z. Large-Scale Variability Characterization and Robust Design Techniques for Nanoscale SRAM. PhD Dissertation / Technical Report No. UCB/EECS-2009-190. University of California, Berkeley, 2009.

[11] Prabhu C. M. R., Singh A. K. Low-Power Reliable SRAM Cell for Write/Read Operation. *IEICE Electronics Express*. 2014. Vol. 11, No. 21. P. 1–6. DOI: 10.1587/elex.11.20140913.

[12] Yadav A., Wairya S. Performance and Area Optimization of SRAM Cell in Nanocomputing Application. *International Journal of Computing and Digital Systems*. 2022. Vol. 11, No. 1. P. 1009–1026. DOI: 10.12785/ijcds/110182.

[13] Prasad G. Design of Low Power and High Stable Proposed SRAM Cell Structure. *International Journal of VLSI and Embedded Systems*. 2013. ISSN 2249–6556.

[14] Saeidi R., Sharifkhani M., Hajsadeghi K. A Subthreshold Symmetric SRAM Cell with High Read Stability. *IEEE Transactions on Circuits and Systems II: Express Briefs*. 2014. Vol. 61, No. 1. P. 26–30. DOI: 10.1109/TCSII.2013.2291064.

[15] Kumar H., Tomar V. K. Design of Low Power with Expanded Noise Margin Subthreshold 12T SRAM Cell for Ultra-Low Power Devices. *Journal of Circuits, Systems and Computers*. 2021. Vol. 30, No. 6. Article 2150106. DOI: 10.1142/S0218126621501061.

[16] Shivaprakash G., Suresh D. S. Design of Low Power 6T-SRAM Cell and Analysis for High Speed Application. *Indian Journal of Science and Technology*. 2016. Vol. 9, No. 46. DOI: 10.17485/ijst/2016/v9i46/106144.

[17] Sreekala K. S., Krishnakumar S. RNM Calculation of 6T SRAM Cell in 32 nm Process Node Based on Current and Voltage Information. *Indian Journal of Science and Technology*. 2017. Vol. 10, No. 29. DOI: 10.17485/ijst/2017/v10i29/116049.

[18] Gupta A., Sindal R., Sharma P., Panchal A., Neema V. Methods for Noise Margin Analysis of Conventional 6T and 8T SRAM Cell. *Materials Today: Proceedings*. 2023. DOI: 10.1016/j.matpr.2023.03.800.

[19] Rawat R. M., Kumar V. A Comparative Study of 6T and 8T SRAM Cell with Improved Read and Write Margins in 130 nm CMOS Technology. *WSEAS Transactions on Circuits and Systems*. 2020. Vol. 19. P. 13–18. DOI: 10.37394/23201.2020.19.2.

[20] Achankunju P. L., Sreekala K. S., James M. K. Design and Read Stability Analysis of 8T Schmitt Trigger Based SRAM. *ICTACT Journal on Microelectronics*. 2017. Vol. 2, No. 4. P. 323–328. DOI: 10.21917/ijme.2017.0056.

[21] Huang C.-R., Chiou L.-Y. Single Bit-Line 8T SRAM Cell with Asynchronous Dual Word-Line Control for Bit-Interleaved Ultra-Low Voltage Operation. *IET*

Circuits, Devices & Systems. 2018. Vol. 12, No. 6. P. 713–719. DOI: 10.1049/iet-cds.2018.5150.

[22] Malik N. Comparative Analysis of Low Power 8T SRAM. *International Journal of Electronic Devices and Networking*. 2020. Vol. 1, No. 1. P. 22–27.

[23] Apollos E. C. Performance Analysis of 6T and 9T SRAM. *International Journal of Engineering Trends and Technology*. 2019. Vol. 67, No. 4. P. 88–101.

[24] Pateliya R. Static Noise Margin of 6T and 8T SRAM Cell in 28-nm CMOS. *SSRN Electronic Journal*. 2019. DOI: 10.2139/ssrn.3462932.

[25] Mukherjee D., Mondal H. K., Reddy B. V. R. Static Noise Margin Analysis of SRAM Cell for High Speed Application. *International Journal of Computer Science Issues*. 2010. Vol. 7, Issue 5.

[26] Birla S., Singh R. K., Pattnaik M. Static Noise Margin Analysis of Various SRAM Topologies. *International Journal of Engineering and Technology*. 2011. Vol. 3, No. 3. P. 304–309. DOI: 10.7763/IJET.2011.V3.242.

[27] Груданов О. М. 8Т комірка пам’яті регістрового файлу для технологій 180 нм із низькою напругою живлення. *Вісті вузів. Радіоелектроніка*. 2023. Т. 66, вип. 12. С. 736–745.

[28] Grudanov O. M. Stability Parameters of Register File Bit Cell with Low Power Consumption Priority. *Electronics and Control Systems*. 2023. Vol. 3, No. 77. P. 40–46. DOI: 10.18372/1990-5548.77.17963.

[29] Grudanov O. M. Power Consumption Calculation of Register File Memory. Інтегровані інтелектуальні робототехнічні комплекси (ІІРТК-2024): матеріали XVII Міжнародної науково-практичної конференції, Київ, Україна, 21–22 травня 2024 р. Київ, 2024. С. 444–446.

[30] Grudanov O. M., Grudanov M. B. Smart-CX — Method of Extraction of Parasitic Capacitances in ICs. *Solid-State Electronics*. 2024. Vol. 221–222. Article 109007. DOI: 10.1016/j.sse.2024.109007.

[31] Grudanov O. M., Grudanov M. B., Shutko V. M. Rule-Based Layout-Driven Parasitic RC Extraction for Post-Layout SPICE Simulation of CMOS ICs. *Chips*. 2026. Vol. 5, No. 2. Article 13. DOI: 10.3390/chips5020013.

[32] Kao W., Lo C.-Y., Basel M., Singh R. Parasitic Extraction: Current State of the Art and Future Trends. *Proceedings of the IEEE*. 2001. Vol. 89. P. 729–739.

[33] Sheehan B. N. TICER: Realizable Reduction of Extracted RC Circuits. In: *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. San Jose, CA, USA, 1999. P. 200–203.

[34] van Genderen A. J., van der Meijs N. P. Reduced RC Models for IC Interconnections with Coupling Capacitances. In: *Proceedings of the European Conference on Design Automation*. Brussels, Belgium, 1992. P. 132–136.

[35] van Genderen A. J., van der Meijs N. P., Beeftink F. SPACE: An Accurate Layout-to-Circuit Extractor for High-Speed MOS and Bipolar Circuits. In: *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. San Jose, CA, USA, 2001.

[36] Wang X., Liu D., Yu W., Wang Z. Improved Boundary Element Method for Fast 3-D Interconnect Resistance Extraction. *IEICE Transactions on Electronics*. 2005. Vol. E88-C. P. 232–240.

[37] Synopsys Inc. *Raphael Reference Manual*. Version 2003.09. Synopsys Inc., Mountain View, CA, USA, 2003.

[38] Siemens Digital Industries Software. *Validating Rule-Based Parasitic Extraction Against a Field Solver Solution*. Siemens Digital Industries Software, Plano, TX, USA, 2023.

[39] Dias D., Goes J., Costa T. L. A Parasitic Resistance Extraction Tool Leveraged by Image Processing. In: *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*. Austin, TX, USA, 2022. DOI: 10.1109/ISCAS48785.2022.9937879.

[40] Liao H., Dai W. W. Partitioning and Reduction of RC Interconnect Networks Based on Scattering Parameter Macromodels. In: *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. San Jose, CA, USA, 1995. P. 704–709.

[41] Liao H., Dai W. Model Order Reduction for Distributed Interconnects Based on Physical Structure. In: *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. San Jose, CA, USA, 1995.

[42] Chen P., Niu D., Jin Z., Sun C., Li Q., Yan H. TSA-TICER: A Two-Stage TICER Acceleration Framework for Model Order Reduction. In: *Proceedings of the Design, Automation & Test in Europe Conference & Exhibition (DATE)*. Valencia, Spain, 2024. P. 1–6. DOI: 10.23919/DATE58400.2024.10546520.

[43] Ma Y., Xu X., Yan S., Zhou Y., Zheng T., Ren Z., Chen L. Extraction of Interconnect Parasitic Capacitance Matrix Based on Deep Neural Network. *Electronics*. 2023. Vol. 12. Article 1440. DOI: 10.3390/electronics12061440.

[44] Yang D., Li H., Yu W., Guo Y., Liang W. CNN-Cap: Effective Convolutional Neural Network-Based Capacitance Models for Interconnect Capacitance Extraction. *ACM Transactions on Design Automation of Electronic Systems*. 2023. Vol. 28. P. 1–22. DOI: 10.1145/3564931.

[45] Liu L., Yang F., Shang L., Zeng X. GNN-Cap: Chip-Scale Interconnect Capacitance Extraction Using Graph Neural Network. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. 2024. Vol. 43. P. 1206–1217. DOI: 10.1109/TCAD.2023.3331942.

[46] Yu W., Shen S., Yang D., Li H., Huang J., Pei C. Deep Learning Inspired Capacitance Extraction Techniques. In: *Proceedings of the 30th Asia and South Pacific Design Automation Conference (ASP-DAC)*. Tokyo, Japan, 2025. P. 106–112. DOI: 10.1145/3658617.3703148.

[47] Tsai J.-C., Huang H.-M., Hsu W.-M., Lee P.-T., Yang J.-H., Huang H.-L., Su Y.-J., Wen H.-P. ResCap: Fast-yet-Accurate Capacitance Extraction for Standard Cell Design by Physics-Guided Machine Learning. In: *Proceedings of the 30th Asia and South Pacific Design Automation Conference (ASP-DAC)*. Tokyo, Japan, 2025. P. 1243–1250. DOI: 10.1145/3658617.3697737.

[48] Grudanov O. M., Grudanov M. B., Shutko V. M. Impact of Hardware Test Equipment on Frequency Characterization of 130-nm Register File. In: *2026 IEEE 43rd International Conference on Electronics and Nanotechnology (ELNANO)*. Kyiv, Ukraine, 2026. P. 364–368. DOI: 10.1109/ELNANO63396.2026.11601742.

[49] Agilent Technologies. *RF Measurement Suite for Agilent 93000 SOC Series*. Application Note 5988-4260EN. Agilent Technologies, 2001.

[50] Agilent Technologies. *System Reference — Agilent 93000 SOC Series Tester*. Agilent / Verigy Technical Manual. 2001.

[51] Kim J. et al. Duty-Cycle Distortion Mechanisms in High-Speed I/O Interfaces. *IEEE Transactions on Circuits and Systems I*. 2013. Vol. 60, No. 6. P. 1482–1493.

[52] Chang L. et al. A Variation-Tolerant SRAM Sense-Amplifier Scheme with Improved Read Margin Stability. *IEEE Transactions on Circuits and Systems I*. 2020. Vol. 67, No. 4. P. 1183–1194.

[53] Kim H. et al. Impact of Sense-Amplifier Offset Variation on SRAM Read Access Time. *Solid-State Electronics*. 2018. Vol. 146. P. 47–54.

[54] Gupta S. et al. Improving Yield of SRAM Read Operation Using Adaptive Bit-Line Development Time. *IEEE Transactions on Semiconductor Manufacturing*. 2021. Vol. 34, No. 2. P. 157–166.

[55] Lee Y., Roy K. Statistical Modeling of Sense-Amplifier Triggering and Read Failure for Nanometer SRAM. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*. 2019. Vol. 27, No. 5. P. 1079–1090.

[56] Grudanov O. Built-in Self Test for Register File Memory Compiler. *Flight. Modern Problems of Science: Abstracts of the XXI International Scientific-Practical Conference of Applicants for Higher Education and Young Scientists*. National Aviation University. Kyiv, 2024. P. 244–245.

[57] Ghale S. B., Ghale S. B. Design and Implementation of Memory BIST for Hybrid Cache Architecture. In: *Proceedings of the 6th International Conference on Communication and Electronics Systems (ICCES)*. 2021. P. 26–31.

[58] Cai Z., Wang Y., Liu S., Lv K. A Novel BIST Algorithm for Low-Voltage SRAM. In: *Proceedings of the IEEE International Test Conference in Asia*. 2019. P. 133–138.

[59] Casu M. R., De Micheli G. Open Source Hardware Design: A Register File Case Study. Master's Thesis. Politecnico di Torino, Italy, 2022.

ДОДАТОК А. Фрагмент програмного коду Lua-модуля генерації timing/power LIB-файлів

У Додатку А Наведено вибрані фрагменти вихідних файлів main.lua, timings1.lua та lib_bst.lua, які реалізують налаштування генерації, вибір даних характеризації, формування таблиць і запис бібліотечного опису Liberty. Імена, числові значення та послідовність операторів відтворено без змін; довгі рядки переносяться засобами текстового редактора.

Це витяги з модулів, а не самостійна програма. Вони використовують змінні та функції, визначені поза наведеними діапазонами. Представлена редакція LIB-модуля містить окремі шаблони для конфігурацій 20×42 і 56×24 та формує один файл для вибраного набору умов.

А.1 Налаштування та виклик генератора

```
local lib_bst = {
    name = "TS152RFSRAM";
    syn = 1; --// 0 = RF1 20x42, 1 = RF1 56x24
    temp = -40;
    voltage_type = 5;
    voltage_value = 11;
    --[[/*   0.8  = 1  */--]]
    --[[/*   0.9  = 2  */--]]
    --[[/*   1.0  = 3  */--]]
    --[[/*   1.1  = 4  */--]]
    --[[/*   1.2  = 5  */--]]
    --[[/*   1.3  = 6  */--]]
    --[[/*   1.55 = 7  */--]]
    --[[/*   1.62 = 8  */--]]
    --[[/*   1.7  = 9  */--]]
```

```
      --[[/*   1.8  = 10 */--]]
      --[[/*   1.98 = 11 */--]]

     case = "fast"; --// fast or slow or typical
      output_dir = options.out_dir;
  }
  lib_main_bst(lib_bst);
```

A.2. Вибір даних характеризації

Функція mg_timings формує таблицю результатів для вибраного технологічного випадку. Допоміжні функції access_time, tsu_tho_time, tro_tfo_time та dcc_power звертаються до масивів, оголошених раніше у timings1.lua. Тут voltage_type є індексом у наборі часових даних, а voltage_value — індексом у таблиці струмів.

```
local function mg_timings( voltage_type, voltage_value, temp, case, asyn )
 --dbg.ensure(ncolumns);
 --dbg.ensure(nrows);
 local r = {};
 if case == "slow" then
 local tacc = access_time(voltage_type, temp, 'slow', asyn)
 r["bad"] = tacc
 local tsu_tho = tsu_tho_time(voltage_type, temp, 'slow', asyn)
 r["bad1"] = tsu_tho
  local tro_tfo = tro_tfo_time(voltage_type, temp, 'slow')
 r["bad2"] = tro_tfo
 elseif case == "typ" then

local tacc = access_time(voltage_type, temp, 'typ', asyn)
 r["typ"] = tacc
```

```
  local tsu_tho = tsu_tho_time(voltage_type, temp, 'typ', asyn)
 r["typ1"] = tsu_tho
  local tro_tfo = tro_tfo_time(voltage_type, temp, 'typ')
 r["typ2"] = tro_tfo
 elseif case == "fast" then
 local tacc = access_time(voltage_type, temp, 'fast', asyn)
 r["fast"] = tacc
 local tsu_tho = tsu_tho_time(voltage_type, temp, 'fast', asyn)
 r["fast1"] = tsu_tho
 local tro_tfo = tro_tfo_time(voltage_type, temp, 'fast')
 r["fast2"] = tro_tfo
 else
 end
  local dcc = dcc_power(voltage_value, asyn)
 r["dcc_pwr"] = dcc
 return r;
end
return mg_timings
```

А.3. Формування таблиць обмежень setup/hold

Наведений фрагмент функції lib_file_a формує по 25 значень для адресних, інформаційних та керувальних сигналів. Масиви ta, tc, tw, tm і td обчислюються у попередній частині цієї функції.

```
local tableADDRsu = {}
local tableADDRho = {}
local tableWENRENsu = {}
local tableWENRENho = {}
local tableDINsu = {}
```

```
local tableDINho = {}
for i = 1, 5, 1 do

    for j = 1, 5, 1 do
     local trf = (j == 1) and 0.1 or (j == 2) and 0.25 or (j == 3) and 0.5 or (j == 4) and 1.0 or (j == 5)
and 2

     local elem_adr_su = ta[j] - tc[i]
     local elem_adr_ho = tc[i]
     local elem_din_su = td[j] - tc[i]
     local elem_din_ho = tc[i]
     local elem_wm_su = tm[j] - tw[i]
     print ("elem_wm_su =",  tm[j], tw[i]);
     local elem_wm_ho = tw[i]

      if elem_adr_su > 0 then elem_adr_su = elem_adr_su
      else elem_adr_su = 0.01
      end

      if (elem_adr_su + elem_adr_ho) < (2*trf)/2   then
      elem_adr_ho =(2*trf)/2  + 0.1
      end

      if elem_din_su > 0 then elem_din_su = elem_din_su
       else elem_din_su = 0.01
       end

       if (elem_din_su + elem_din_ho) < (2*trf)/2   then
       elem_din_ho =(2*trf)/2  + 0.1
      end

       if elem_wm_su > 0 then elem_wm_su = elem_wm_su
```

```
        else elem_wm_su = 0.01
        end

        if (elem_wm_su + elem_wm_ho) < (2*trf)/2   then
        elem_wm_ho =(2*trf)/2  + 0.1
       end


   table.insert(tableADDRsu, string.format( "%2.4f", elem_adr_su))
   table.insert(tableADDRho, string.format( "%2.4f", elem_adr_ho))
   table.insert(tableDINsu, string.format( "%2.4f", elem_din_su))
   table.insert(tableDINho, string.format( "%2.4f", elem_din_ho))
   table.insert(tableWENRENsu, string.format( "%2.4f", elem_wm_su))
   table.insert(tableWENRENho, string.format( "%2.4f", elem_wm_ho))

    end
end
```

А.4. Шаблони таблиць і запис значень

```
  lu_table_template(CELL_UP) {
        variable_1 : total_output_net_capacitance;
         variable_2 : input_net_transition;
        index_1 (" 0.010000, 0.100000, 0.200000, 0.350000, 0.500000");
        index_2 (" 0.100000, 0.250000, 0.500000, 1.000000, 2.000000");

    }
    lu_table_template(CELL_DN) {
        variable_1 : total_output_net_capacitance;
        variable_2 : input_net_transition;
        index_1 (" 0.010000, 0.100000, 0.200000, 0.350000, 0.500000");
        index_2 (" 0.100000, 0.250000, 0.500000, 1.000000, 2.000000");
    }
    lu_table_template(CONSTRAINT_HIGH_POS) {
```

```
    variable_1 : related_pin_transition;
    variable_2 : constrained_pin_transition;
    index_1 (" 0.100000, 0.250000, 0.500000, 1.000000, 2.000000");
    index_2 (" 0.100000, 0.250000, 0.500000, 1.000000, 2.000000");
  }
  lu_table_template(CLK_TRAN) {
    variable_1 : constrained_pin_transition;
    index_1 (" 0.100000, 0.250000, 0.500000, 1.000000, 2.000000");
  }
  lu_table_template(TRAN) {
    variable_1 : total_output_net_capacitance;
    index_1 (" 0.010000, 0.100000, 0.200000, 0.350000, 0.500000");
  }
  power_lut_template(INPUT_BY_TRANS) {
    variable_1 : input_transition_time;
    index_1 (" 0.100000, 0.250000, 0.500000, 1.000000, 2.000000");
 }
```

A.5. Запис таблиці для адресної шини

```
io.write([[
                    timing() {
              timing_type      : setup_rising;
                  related_pin      : "CLK";
                  rise_constraint(CONSTRAINT_HIGH_POS) {
                     values("          ]]..tableADDRsu[1]..[[,                  ]]..tableADDRsu[2]..[[,
]]..tableADDRsu[3]..[[,  ]]..tableADDRsu[4]..[[,  ]]..tableADDRsu[5]..[["",\
                         " ]]..tableADDRsu[6]..[[,   ]]..tableADDRsu[7]..[[,   ]]..tableADDRsu[8]..[[,
]]..tableADDRsu[9]..[[,  ]]..tableADDRsu[10]..[["",\
                         "          ]]..tableADDRsu[11]..[[,                  ]]..tableADDRsu[12]..[[,
]]..tableADDRsu[13]..[[,  ]]..tableADDRsu[14]..[[,  ]]..tableADDRsu[15]..[["",\
                         "          ]]..tableADDRsu[16]..[[,                  ]]..tableADDRsu[17]..[[,
]]..tableADDRsu[18]..[[,  ]]..tableADDRsu[19]..[[,  ]]..tableADDRsu[20]..[["",\
                         "          ]]..tableADDRsu[21]..[[,                  ]]..tableADDRsu[22]..[[,
```

]]..tableADDRsu[23]..[[,]]..tableADDRsu[24]..[[,]]..tableADDRsu[25]..[[");

}

]]);

ДОДАТОК Б. АКТ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЙНОГО ДОСЛІДЖЕННЯ

SILVACO

ЄДРПОУ юридичної особи
39949495

Найменування юридичної особи
ТОВАРИСТВО З ОБМЕЖЕНОЮ ВІДПОВІДАЛЬНІСТЮ "СІЛВАКО ЮКРЕЙН"

Місцезнаходження юридичної особи
Україна, 04136, місто Київ, вул. Некрасова Віктора, будинок 1-3

АКТ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЙНОГО ДОСЛІДЖЕННЯ № 150626/1

м. Київ «15» червня 2026 р.

Товариство з обмеженою відповідальністю «СІЛВАКО ЮКРЕЙН» засвідчує, що результати дисертаційного дослідження аспіранта кафедри ЕРМІТ Київського авіаційного інституту **Груданова Олександра Миколайовича** на тему «**Методи проєктування енергоефективної двопортової RFSRAM-пам'яті для телекомунікаційних систем**», поданої на здобуття ступеня доктора філософії за спеціальністю 172 — електронні комунікації та радіотехніка, впроваджені та використані у науково-дослідній, інженерно-проєктній та програмно-технічній діяльності ТОВ «СІЛВАКО ЮКРЕЙН».

Результати дисертаційного дослідження використано під час розроблення, удосконалення та практичного застосування програмного забезпечення RFSRAM Memory Compiler, призначеного для параметричної генерації макросів двопортової пам'яті регістрового типу RFSRAM з пріоритетом низького енергоспоживання.

У межах впровадження використано такі результати дисертаційної роботи:

- методи проєктування енергоефективної двопортової RFSRAM-пам'яті на основі 8T SRAM-комірки;
- SNM/WNM-орієнтований підхід до вибору геометричних параметрів транзисторів комірки пам'яті з урахуванням знижених напруг живлення, PVT-умов та статистичного розкиду параметрів MOS-транзисторів;
- параметризовану архітектуру RFSRAM Memory Compiler та алгоритми формування матриці комірок пам'яті, збірки верхнього рівня макросу, підключення адресних формувачів, дешифраторів рядків, блоків керування, вхідних та вихідних буферів;
- алгоритми формування схемного LVS-нетліста та вихідних даних для подальшого схемотехнічного і післятопологічного моделювання;

– методику SPICE-характеризації згенерованих макросів RFSRAM для оцінювання часу доступу, затримок внутрішніх сигналів, середнього динамічного струму та енергоспоживання;
– підхід до генерації timing/power LIB-файлів на основі результатів SPICE-моделювання та 5×5 look-up tables;
– методику врахування паразитних RC-параметрів топології під час post-layout SPICE-моделювання;
– підхід до інтеграції вбудованого самотестування BIST для функціональної перевірки операцій читання та запису у згенерованих макросах RFSRAM.

Практичне використання зазначених результатів дозволило:

– формалізувати маршрут проєктування параметризованих макросів двопортової RFSRAM-пам’яті;
– скоротити час підготовки нових інстансів пам’яті з різною кількістю слів, розрядністю та режимами роботи;
– забезпечити узгодженість layout-, netlist- та timing/power-представлень згенерованого макросу пам’яті;
– підвищити достовірність оцінювання часу доступу, динамічного струму та енергоспоживання до етапу виготовлення мікросхеми у кремнії;
– забезпечити можливість використання згенерованих RFSRAM-макросів як embedded memory IP-блоків у складі систем на кристалі.

Впровадження результатів дисертаційного дослідження підтверджує їх практичну цінність для розроблення програмних засобів і методів проєктування енергоефективних IP-блоків вбудованої пам’яті для інтегральних схем, зокрема для телекомунікаційних, радіотехнічних, сенсорних, автономних та оборонних систем.

Акт складено для подання до Київського авіаційного інституту як підтвердження впровадження результатів дисертаційного дослідження.

Директор ТОВ «СІЛВАКО ЮКРЕЙН» Тетяна ЯЩЕНКО

М.П.