

Міністерство освіти і науки, молоді та спорту України
Тернопільський національний економічний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерної інженерії

«До захисту допущено»
Завідувач кафедри
комп'ютерної інженерії
к.т.н., доц. О.М.Березький

“ _____ ” _____ 20__ р.

ДИПЛОМНИЙ ПРОЕКТ
освітньо-кваліфікаційного рівня "Спеціаліст"
зі спеціальності 6.091501 “Комп’ютерні системи та мережі”
на тему:

VHDL – модель статичної оперативної пам’яті з довільним доступом

Студент групи КСМс-51 _____ Деренюк П.І.
(підпис)

Керівник:
викладач _____ Дубчак Л.О.
(підпис)

Нормоконтроль:
к.т.н., доцент _____ Васильків Н.М.
(підпис)

Консультант
з охорони праці:
доцент _____ Сапожник Г.В.
(підпис)

2012

Міністерство освіти і науки, молоді та спорту України
Тернопільський національний економічний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерної інженерії
Спеціальність 6.091501 – “Комп'ютерні системи та мережі”

“Затверджую”
Завідувач кафедри
комп'ютерної інженерії
к.т.н., доц. О.М.Березький

“ ___ ” _____ 20__ р.

ЗАВДАННЯ НА ДИПЛОМНИЙ ПРОЕКТ СТУДЕНТА

Деренюка Павла Івановича

1. **Тема проекту:** " VHDL – модель статичної оперативної пам'яті з довільним доступом" затверджена наказом університету № ___ від “ ___ ” _____ 20__ р.
2. **Термін здачі студентом закінченого проекту** “ ___ ” _____ 20__ р.
3. **Вихідні дані для проекту:** Технічне завдання.
4. **Перелік задач, які мають бути вирішені:**
 - провести аналіз існуючих рішень для програмної реалізації;
 - визначити оптимальну структурну перевірки цілісності даних;
 - проаналізувати оперативну пам'ять та її типи;
 - визначити принципи функціонування статичної пам'яті з довільним доступом;
 - визначити типи статичної пам'яті з довільним доступом;
 - визначити основні характеристики SRAM;
 - реалізувати програмно оперативну статичну пам'ять з довільним доступом .
5. **Перелік графічного матеріалу** (з точним вказанням обов'язкових креслень)

- Оперативна пам'ять. Схема структурна
- SRAM. Схема функціональна
- Зв'язок модулів програми. Схема структурна
- Блок – діаграма SRAM. Схема структурна

6. Консультанти по проекту (із зазначенням розділів):

Розділ	Консультант	Підпис
Охорона праці	Сапожник Г.В.	

КАЛЕНДАРНИЙ ПЛАН

№	Назва розділів дипломного проекту	Термін виконання	Позначки керівника про виконання завдань
1	Аналіз інформаційної безпеки в комп'ютерних системах	15.09.2011 – 5.11.2011	
2	Побудова та визначення принципів функціонування статичної пам'яті з довільним доступом	6.11.2011 – 31.01.2012	
3	Розробка програмного комплексу	1.02.2012 – 14.04.2012	
4	Охорона праці	15.04.2012 – 23.04.2012	

Завдання прийняв до виконання _____
(підпис)

Керівник дипломного проекту _____
(підпис)

ТЕХНІЧНЕ ЗАВДАННЯ

1. НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ

1.1 Статична оперативна пам'ять з довільним доступом

1.2 Область застосування – комп'ютерні системи для зберігання коду та даних програм під час їх виконання

2. ОСНОВА ДЛЯ РОЗРОБКИ

Основою для розробки є завдання на дипломний проект, затверджене кафедрою комп'ютерної інженерії факультету комп'ютерних інформаційних технологій Тернопільського національного економічного університету

3. ПРИЗНАЧЕННЯ РОЗРОБКИ

Метою даної переддипломної практики є зробити аналіз а також визначити побудову та принципи функціонування статичної пам'яті з довільним доступом

4.ДЖЕРЕЛА РОЗРОБКИ

Джерелами даної розробки є матеріали навчальної та реферативної наукової літератури, технічна документація, існуючі програмні та програмно-апаратні системи, журнали, науково-дослідні роботи вітчизняних та закордонних вчених: Жарков, Симонович, Мюллер, Борзенко, Кравец, Шейнин, Гук тощо

5. ЗАДАЧІ РОЗРОБКИ

5.1 Аналіз предметної області з метою вибору існуючих або розробку нових (удосконалення існуючих) моделей, що описують предметну область, програмна постановка задач та вибір моделей їх розв'язку

5.2 Вибір та обґрунтування вибору архітектури, вибір програмного середовища та необхідних додаткових бібліотек для реалізації програмної системи, вибір зовнішніх програмних та апаратних засобів для забезпечення роботи програмної системи

5.3 Аналіз та уточнення вимог технічного завдання з точки зору обраних моделей, методів, алгоритмів та середовища розробки

5.4 Проектування, реалізація та тестування окремих компонент програмної системи, розробка принципів взаємозв'язку між ними.

5.5 Тестування програмного коду VHDL – моделі оперативної статичної пам'яті з довільним доступом

6. ВИМОГИ ДО ПРОГРАМНОЇ СИСТЕМИ

6.1 Функціональні вимоги

6.1.1 Оператор системи повинен мати змогу виконувати наступні функції:

6.1.1.1. Отримання даних для запису в SRAM

6.1.1.2. Проведення попередньої обробки отриманих даних

6.1.1.3. Виділення результатів роботи SRAM в окремому файлі

6.1.1.4. Формування звітів на основі проведених досліджень у графічному форматі

6.1.2 Вхідна інформація отримується шляхом:

6.1.2.1. Завантаження графічних файлів із жорсткого диска

6.1.3 Вихідна інформація:

6.1.3.1. Вихідна інформація повинна подаватись у простому та інтуїтивно зрозумілому для користувача форматі

6.1.3.2. Формування звітів повинно відбуватись у реальному часі

6.1.3.3. Вихідна інформація виводиться у текстовому, графічному (діаграми) форматах

6.2. Вимоги до надійності

6.2.1 Передбачити контроль введеної інформації

6.2.2 Розробити комплекс заходів контролю коректності дій користувача під час роботи з системою

6.2.3 Забезпечити можливість відновлення роботи системи після збоїв.

6.3 Вимоги до апаратної засобів

6.3.1 Система повинна працювати на IBM-сумісних робочих станціях

6.3.2 Мінімальні вимоги до робочих станцій:

- 6.3.2.1 Процесор від 1 ГГц
- 6.3.2.2 Оперативна пам'ять від 512 Мб
- 6.3.2.3 Відеокарта від 32 Мб
- 6.3.2.4 Об'єм пам'яті на жорсткому диску до 100 Мб
- 6.3.2.5 Клавіатура, маніпулятор «миша»
- 6.3.3 Вимоги до програмного забезпечення:
 - 6.3.3.1 Операційна система сімейства Windows
 - 6.3.3.2 Графічна бібліотека OpenGL
 - 6.3.3.3 Пакет програм Active - HDL
- 6.3.4 Сумісність з сучасними форматами даних:
 - 6.3.4.1 Вхідна інформація подається у форматах «*.bmp» та «*.jpg»
 - 6.3.4.2 Вихідна інформація подається у форматах «*.txt» та «*.doc» для текстової інформації, «*.xls» для табличної інформації та «*.bmp», «*.jpg» для графічної

6.4 Вимоги до програмної документації

- 6.4.1 Код програмних модулів повинен містити необхідні для його розуміння коментарі
- 6.4.2 Розроблене програмне забезпечення повинно включати довідкову систему

6.5 Вимоги експлуатації

- 6.5.1 Кліматичні вимоги до експлуатації, при яких забезпечується робота програми повинні відповідати кліматичним умовам експлуатації наявних технічних засобів
- 6.5.2 Вимоги до кваліфікації та численності персоналу. Мінімальна кількість персоналу, необхідного для роботи програми, може складати одну штатну одиницю – кінцевого користувача програми – оператор

6.6 Вимоги до захисту

- 6.6.1 Мінімальна довжина пароля 10 символів

7. ВИМОГИ ОХОРОНИ ПРАЦІ

В розділі “Охорона праці ” дипломного проекту повинен бути даний аналіз умов праці в приміщенні

8. ПОРЯДОК КОНТРОЛЮ І ПРИЙОМКИ

8.1 Представлення дипломного проекту на попередній захист

8.2 Представлення дипломного проекту на захист

ЗМІСТ

Вступ.....	9
1. Оперативна пам'ять та її призначення.....	10
1.1 Динамічна пам'ять.....	11
1.2 Статична пам'ять.....	22
1.3 Постановка задачі.....	29
2. Побудова та принципи функціонування статичної пам'яті з довільним доступом.....	31
2.1 Будова SRAM.....	31
2.2 Типи статичної пам'яті з довільним доступом.....	33
2.3 Основні характеристики SRAM.....	38
3. Проектування SRAM засобами Active – HDL.....	50
3.1 Вибір середовища проектування.....	50
3.2 HDL – модель статичної пам'яті з довільним доступом.....	59
3.3 Результати роботи проекрованої SRAM.....	63
4. Охорона праці.....	65
4.1 Аналіз санітарно-гігієнічних умов праці.....	65
4.2 Пожежна безпека.....	75
Висновок.....	77
Список використаних джерел	78
Додаток А.....	80

					ДП. КСМ. 07208/11. 00.00.000 ПЗ						
Змн	Арк.	№ докум.	Підпис	Дата	“Звіт про проходження переддипломної практики”			Літ.	Арк.	Акрушів	
Розроб.		Деренюк П.І								8	62
Перевір.		Дубчак Л.О.						ТНЕУ.ФКІТ.КСМс-51			
Реценз.											
Н. Контр.											
Затверд.											

ВСТУП

Метою даної переддипломної практики є зробити аналіз а також визначити побудову та принципи функціонування статичної пам'яті з довільним доступом.

Оперативна пам'ять - це робоча область для процесора комп'ютера. У ній під час роботи зберігаються програми і дані. Оперативна пам'ять часто розглядається як тимчасове сховище, тому що дані і програми в ній зберігаються тільки при включеному комп'ютері або до натискання кнопки перезагрузки (reset). Перед вимиканням, натисканням кнопки перезагрузки всі дані, піддані змінам під час роботи, необхідно зберегти на запам'ятовуючому пристрої, що може зберігати інформацію постійно (зазвичай це жорсткий диск). При новому включенні збережена інформація знову може бути завантажена в пам'ять.

Пристрої оперативної пам'яті іноді називають запам'ятовуючими пристроями з довільним доступом. Це означає, що звертання до даних, що зберігається в оперативній пам'яті, не залежить від порядку їх розташування в ній. Коли говорять про пам'ять комп'ютера, зазвичай мають на увазі оперативну пам'ять, ніж мікросхеми або пам'яті модулі, у яких зберігаються активні програми і дані, використовувані процесором.

Оперативно запам'ятовуючий пристрій, який не треба регенерувати називається статичної пам'яттю з довільним доступом або просто статичної пам'яттю. Гідність цього виду пам'яті - швидкість. Оскільки тригери зібрані на вентилях, а час затримки вентиля дуже мало, то і перемикання стану тригера відбувається дуже швидко, також вона дозволяє підтримувати стан сигналу без постійного перезапису.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дат		

1 ОПЕРАТИВНА ПАМ'ЯТЬ ТА ЇЇ ПРИЗНАЧЕННЯ

Оперативна пам'ять (англ. Random Access Memory, - пам'ять з довільним доступом, первинна пам'ять) — пам'ять ЕОМ, призначена для зберігання коду та даних програм під час їх виконання. У сучасних комп'ютерах оперативна пам'ять переважно представлена динамічною пам'яттю з довільним доступом DRAM((англ. Dynamic random access memory – динамічна пам'ять з довільним доступом)[2].

Протилежністю до пам'яті з довільним доступом є пам'ять з послідовним доступом. При довільному доступі, пам'ять організована таким чином, що в будь-яку мить можна отримати значення, записане в будь-якій комірці пам'яті, не переглядаючи інші. На противагу при застосуванні яка реалізується, наприклад, на магнітній стрічці, для доступу до певного елемента пам'яті потрібно прокрутити стрічку, зчитуючи інші елементи.

Види запам'ятовуючих пристроїв з довільним доступом:

Напівпровідникова статична пам'ять (SRAM) складається з комірок які є напівпровідниковими тригерами. Її переваги є невелике енергоспоживання, висока швидкодія. Відсутність необхідності проводити «регенерацію». Недоліки — малий обсяг, висока вартість. Нині вона широко використовується як кеш-пам'ять процесорів у комп'ютерах.

Напівпровідникова динамічна (DRAM) побудована таким чином що кожна комірка є конденсатором на основі переходу КМОН-транзистора. Переваги — низька вартість, великий обсяг. Недоліки — необхідність періодичного прочитування і перезапису кожної комірки (регенерації), і як наслідок, зниження швидкодії, велике енергоспоживання. Процес регенерації реалізується спеціальним контролером, встановленим на материнській платі або в центральному процесорі. DRAM зазвичай використовується як оперативна пам'ять (ОЗП) комп'ютерів.[2]

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дат		

Феромагнітна пам'ять - є матрицею з провідників, на перетині яких знаходяться кільця або біакси, виготовлені з феромагнітних матеріалів. Переваги — стійкість до радіації, збереження інформації при виключенні живлення; недоліки — мала ємність, велика вага, стирання інформації при кожному читанні. В наш час в такому, зібраному з дискретних компонентів вигляді, не застосовується.

У 2003 року з'явилася магнітна пам'ять MRAM (англ. Magnetoresistive random-access memory – запам'ятовуючий пристрій з довільним доступом) в інтегральному виконанні. Поєднуючи швидкість SRAM і можливість зберігання інформації при відімкненому живленні, MRAM є перспективною заміною типам ROM (англ. Read only memory – постійно запам'ятовуючий пристрій) і RAM (англ. Random access memory – пам'ять з довільним доступом). Проте вона приблизно удвічі дорожча за мікросхеми SRAM (при тій же ємності і габаритах).

1.2 Динамічна пам'ять

Динамічна оперативна пам'ять або DRAM - один із видів комп'ютерної пам'яті із випадковим доступом (RAM), найчастіше використовується в якості оперативно-запам'ятовуючого пристрою. Основна перевага пам'яті цього типу полягає в тому, що її комірки розташовані дуже щільно, тобто в невелику мікросхему можна розмістити багато бітів, тобто, на їх основі можна побудувати пам'ять великої ємності[3].

Елементи пам'яті в мікросхемі DRAM — це крихітні конденсатори, які утримують заряди. Саме так (наявністю або відсутністю зарядів) і кодуються біти. Проблеми, пов'язані з пам'яттю цього типу, викликані тим, що вона динамічна, тобто повинна постійно регенеруватися, оскільки інакше

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		11

електричні заряди в конденсаторах пам'яті «стікатимуть» і дані будуть втрачені. Регенерація відбувається, коли контролер пам'яті системи бере перерву і звертається до всіх рядків даних в мікросхемах пам'яті. Більшість систем мають контролер пам'яті (зазвичай вбудований в набір мікросхем системної плати), який налаштований на відповідну промисловим стандартам частоту регенерації, рівну 15 мкс. До всіх рядків даних звернення здійснюється після проходження 128 спеціальних циклів регенерації. Це означає, що кожні 1,92 мс прочитуються всі рядки в пам'яті для забезпечення регенерації даних.[4]

Регенерація пам'яті, на жаль, віднімає час в процесора: кожен цикл регенерації за тривалістю займає декілька циклів центрального процесора. У старих комп'ютерах цикли регенерації могли займати до 10 % (або більше) процесорного часу, але в сучасних системах, що працюють на частотах, рівних сотням мегагерц, витрати на регенерацію складають 1 % (або менше) процесорного часу. Деякі системи дозволяють змінити параметри регенерації за допомогою програми установки параметрів CMOS (англ. Complementary-symmetry/metal-oxide semiconductor – технологія побудови електронних схем), але збільшення часу між циклами регенерації може призвести до того, що в деяких елементах пам'яті заряд зникне, а це викличе збої пам'яті. В більшості випадків надійніше дотримуватися частоти регенерації, що рекомендується, або заданої за замовчуванням. Оскільки витрати на регенерацію в сучасних комп'ютерах складають менше 1 %, зміна частоти регенерації має незначний вплив на характеристики комп'ютера.

У пристроях DRAM для зберігання одного біта використовується тільки один транзистор і пара конденсаторів, тому вони місткіші, ніж мікросхеми інших типів пам'яті.

В даний час є мікросхеми динамічної оперативної пам'яті ємністю до 512 Мбіт і більше. Це означає, що подібні мікросхеми містять 512 млн (і навіть більше) транзисторів. Адаже Pentium IV має 55 млн транзисторів. Річ у

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		12

тому, що в мікросхемі пам'яті всі транзистори і конденсатори розміщуються послідовно, зазвичай у вузлах квадратних ґрат, у вигляді дуже простих структур, що періодично повторюються, на відміну від процесора, що є складнішою схемою різних структур і не має чіткої організації.

Транзистор для кожного однорозрядного регістра DRAM використовується для читання стану суміжного конденсатора. Якщо конденсатор заряджений, у комірці записана 1, якщо заряду немає - записаний 0. Заряди в крихітних конденсаторах увесь час зникають, ось чому пам'ять повинна постійно регенеруватися. Навіть миттєве переривання подачі живлення або який-небудь збій в циклах регенерації приведе до втрати заряду у комірці DRAM, а отже, і до втрати даних.

Динамічна оперативна пам'ять використовується в персональних комп'ютерах. Оскільки вона недорога, то мікросхеми можуть бути щільно укомплектовані, а це означає, що пристрій великої ємності, що запам'ятовує, може займати невеликий простір. На жаль, пам'ять цього типу не відрізняється високою швидкістю, зазвичай вона набагато «повільніша» за процесор. Тому існує безліч різних типів організації DRAM, що дозволяють покращити цю характеристику.

Конструктивно пам'ять DRAM складається із комірок розміром в 1 або 4 біти, в кожній із яких можна зберігати певний обсяг даних. Сукупність комірок такої пам'яті створюють умовний «прямокутник», який складається із певної кількості стрічок та стовпців. Один такий «прямокутник» називається сторінкою, а сукупність сторінок називається банком. Весь набір комірок умовно ділиться на кілька областей.

В сучасних комп'ютерах фізично DRAM-пам'ять представляє собою електrolітичну плату — модуль, на якому розміщуються мікросхеми пам'яті зі спеціалізованим роз'ємом для підключення до материнської плати. Роль комірок відіграють конденсатори та транзистори, які розташовані всередині мікросхем пам'яті. Конденсатори заряджаються у випадку, коли в комірку

заноситься одиничний біт, або розряжаються у випадку, якщо в комірку заноситься нульовий біт. Транзистори потрібні для утримання заряду всередині конденсатора. За відсутності подачі електроенергії до оперативної пам'яті відбувається розряження конденсаторів і пам'ять спустошується. Ця динамічна зміна заряду конденсатора і є основним принципом роботи пам'яті типу DRAM. Елементом пам'яті такого типу є чутливий підсилювач (англ. sense amp), який підключений до кожного із стовпців «прямокутника». Він реагує на слабкий потік електронів, які рухаються через відкриті транзистори із конденсаторів, і зчитує цілком всю сторінку. Саме сторінки і являються мінімальною порцією обміну із динамічною пам'яттю, тому що обмін даними із окремо взятою коміркою нереальний.

Принцип роботи DRAM читання та запису для пристрої матриці 4x4 поданий на рисунку 1.1 та 1.2, відповідно [5].

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дат		

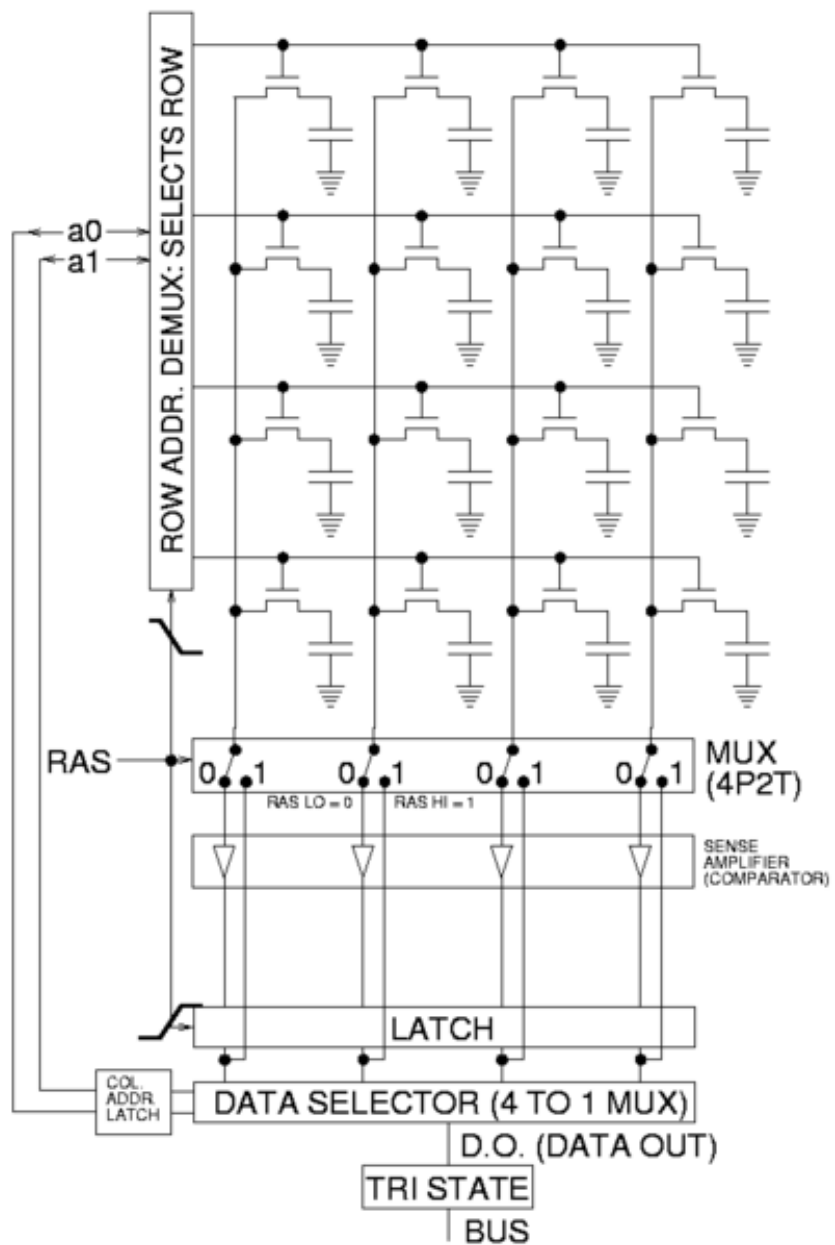


Рисунок 1.1 - Принцип роботи DRAM читання
для простої матриці 4x4

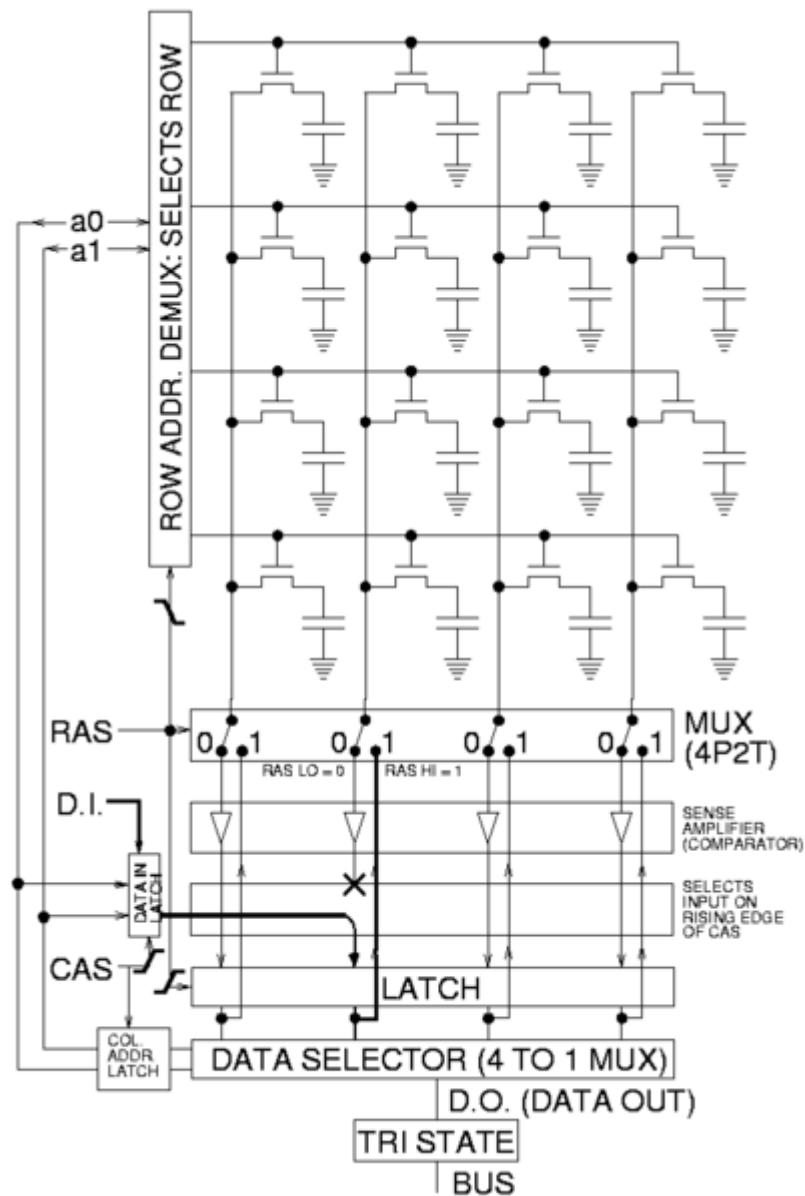


Рисунок 1.2 - Принцип роботи DRAM запису для простої матриці 4x4

Основними характеристиками DRAM являються таймінги та робоча частота. Для звернення до комірок контролер задає номер банку, номер сторінки в ньому, номер стрічки та номер стовпчика. На ці всі запити затрачаючи певний час. Крім того доволі великий період йде на відкриття та закриття самого банку після виконання операції. На кожну дію вимагається час, який називається таймінгом. Основними таймінгами DRAM являються: затримка між подачею номеру стрічки і номера стовпчика, який називається

часом повного доступу (англ. RAS to CAS delay); затримка між подачею номеру стовпчика і отримання вмісту комірки, що називається часом робочого циклу (англ. CAS delay); затримка між читанням останньої комірки та подачею номеру наступної стрічки (англ. RAS precharge). Таймінги вимірюються в наносекундах і чим менша величина цих таймінгів, тим швидше працює оперативна пам'ять. Робоча частота вимірюється в мегагерцах і збільшення робочої частоти пам'яті призводить до збільшення її швидкодії.

На відміну від статичної пам'яті типу SRAM (англ. static random access memory), яка є конструктивно складнішим і дорожчим типом пам'яті RAM та використовується в основному для кеш-пам'яті, пам'ять DRAM виготовляється на основі конденсаторів невеликої ємності, які швидко втрачають заряд, тому інформацію приходится оновлювати через певні проміжки часу, щоб уникнути втрати даних. Цей процес називається регенерацією пам'яті. Регенерація реалізовується спеціальним мікроконтролером, встановленим на материнській платі або інтегрованим в кристал центрального процесора. Протягом певного часу, який називається крок регенерації, в DRAM перезаписується ціла стрічка комірок і через кожні 8-64 мс оновлюються всі стрічки пам'яті [6].

Процес регенерації пам'яті в класичному варіанті суттєво «гальмує» роботу системи, оскільки в цей час обмін даними із пам'яттю неможливий. Регенерація, яка базується на принципі перебору стрічок не використовується в сучасних типах DRAM. Існують декілька економічніших варіантів даного процесу — розширений, пакетний, розподільчий; найекономічнішим є метод прихованої регенерації.

Із нових технологій регенерації можна виділити тип регенерації PASR (англ. Partial Array Self Refresh), що використовується компанією Samsung в чипах пам'яті SDRAM із низьким рівнем енергоспоживання. Регенерація комірок виконується тільки в період очікування тих банків пам'яті, в яких

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		17

вона заповнена. Паралельно з цією технологією реалізовується метод TCSR (англ. Temperature Compensated Self Refresh), який призначений для регулювання швидкості процесу регенерації в залежності від робочої температури.

Протягом тривалого часу розробниками створювалися різноманітні типи пам'яті. Вони характеризувалися різними параметрами і в них використовувалися різні технічні рішення. Основною рушійною силою розвитку пам'яті був розвиток комп'ютерів та центральних процесорів. Постійно вимагалось збільшення швидкодії та обсягу оперативної пам'яті.

Асинхронна DRAM є основною формою, з якої походять всі інші. Асинхронний чип DRAM має потужні зв'язки, деяку кількість вхідних адресацій (зазвичай 12) і кілька (як правило, 1 або 4) двонаправлених ліній даних. Існують чотири активні параметри контрольних сигналів:

-RAS- ряд адресного простору (англ. Row Address Strobe). Адресація розпочинається і слідує до кінця простору /RAS, потім вибирається вільний рядок для запису. Рядок залишається відкритим до тих пір, поки /RAS знаходиться на низькому рівні;

-CAS- колона адресного простору (англ. Column Address Strobe). Адресація розпочинається на кінці простору /CAS і вибирається стовпчик із всіх відкритих, який у цей час придатний для читання і запису;

-WE- доступ запису (англ. Write Enable). Цей сигнал визначається з врахуванням кінцевого статусу /CAS читання (якщо високий) або запису (якщо низький). При низькому рівні дані входять та прямують до прикінцевого краю /CAS;

-OE- доступ зчитування (англ. Output Enable). Це є додатковим сигналом, який контролює вихід до даних I/O пінів. Дата-піни спрямовуються до DRAM чипів, якщо /RAS і /CAS є низькими і /WE високим та /OE низьким. У багатьох програмах, /OE може бути постійно низьким

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дат		

(зчитування завжди доступне), але це може бути корисним при підключенні декількох чипів пам'яті паралельно.

Описаний інтерфейс забезпечує прямий контроль внутрішніх таймінгів. Коли /RAS низький, то /CAS цикл не повинен робити спроб заповнення простору до тих пір, поки чутливі підсилювачі виявлятимуть стан пам'яті, та /RAS не повинен ставати високим, поки комірка зберігання не буде регенерована. /RAS має бути високим наскільки довго, скільки потрібно для повної перезарядки.

Video RAM (VRAM) - спеціальний тип оперативної пам'яті, який був розроблений на основі пам'яті типу SDRAM для використання у відеокартках (розробки велися Ф. Діллом (F. Dill) і Р. Матіком (R. Matick) починаючи з 1980 року, а запатентували в 1985 (Патент США 4,541,075))[.....]. Він дозволяв забезпечувати неперервний потік даних в процесі оновлення зображення, що було необхідно для реалізації зображення високої якості. На основі пам'яті типу VRAM, з'явилася специфікація пам'яті типу Windows RAM (WRAM), хоча іноді її помилково пов'язують із операційними системами сімейства Windows. Її продуктивність стала на 25 % вище, ніж у оригінальної пам'яті типу SDRAM, завдяки деяким технічним змінам.

Але пізніше, вже в 1990-х стандартна пам'ять DRAM (тобто SDRAM) стала дешевшою, щільнішою та високопродуктивнішою настільки, що витіснила VRAM.

Сторінкова пам'ять (англ. page mode DRAM, PM DRAM) являлася однією із перших типів комп'ютерної пам'яті, яка випускалася на початку 90-х років. Але з ростом швидкодії центральних процесорів та зростанням системних вимог до програм та операційних систем почали підвищуватися не тільки до обсяги оперативної пам'яті, а й до швидкості її роботи.

Швидка сторінкова пам'ять (англ. fast page mode DRAM, FPM DRAM) з'явилася в 1995 році. Принципово нових змін пам'ять не набула, а збільшення швидкості роботи досягалося підвищенням навантаження на

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дат		

апаратну складову. Даний тип пам'яті в основному використовувався для комп'ютерів із процесорами Intel 80486 чи аналогічних процесорів інших фірм. Пам'ять могла працювати на частотах 25 МГц і 33 МГц, із часом повного доступу 70нс і 60нс та часом робочого циклу 40 нс і 35 нс, відповідно [7].

Пам'ять із вдосконаленим виходом (англ. extended data out DRAM, EDO DRAM) з'явилася на ринку в 1996 році і стала активно використовуватися на комп'ютерах із процесорами Intel Pentium і новіше. Її продуктивність виявилася на 10-15 % вище, в порівнянні із пам'яттю типу FPM DRAM. Її робочі частоти були 40 МГц і 50 МГц, відповідно, і час повного доступу — 25 нс і 20 нс. Ця пам'ять містить регістр-застібку (англ. data latch) вихідних даних, що забезпечувало деяку конвеєризацію роботи для підвищення продуктивності при читанні.

Синхрона пам'ять (англ. synchronous DRAM, SDRAM). Використовує тактовий генератор для синхронізації всіх сигналів та використання конвеєрної обробки інформації. Також пам'ять могла працювати при значно вищих частотах системної шини (100 МГц і вище). Недоліками даного типу пам'яті була висока ціна, а також несумісність із багатьма чипсетамі і материнськими платами в силу своїх нових конструктивних особливостей. Робочі частоти даного типу пам'яті могли дорівнювати 66 МГц, 100 МГц чи 133 МГц, час повного доступу — 40 нс і 30 нс, а час робочого циклу — 10 нс і 7,5 нс [7].

Пакетна пам'ять EDO RAM (англ. burst extended data output DRAM, BEDO DRAM) стала дешевою альтернативою пам'яті SDRAM. Заснована на пам'яті EDO DRAM, її ключовою особливістю являлася технологія по блочному читанню даних (блок даних читався за один такт), що зробило її роботу швидше, ніж у пам'яті типу SDRAM. Але неможливість працювати на частоті системної шини більше 66 МГц не дозволило даному типу пам'яті стати популярною.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дат		

DDR SDRAM (англ. double data rate SDRAM, SDRAM із подвоєною швидкістю передачі даних або SDRAM II) спочатку використовувалася у відеокартках, але потім з'явилася підтримка DDR SDRAM зі сторони чипсетів. Пам'ять працює на частотах 100 МГц и 133 МГц, її час повного доступу — 30нс і 22,5 нс, а час роботи циклу — 5 нс і 3,75 нс.

Direct RDRAM, або Direct Rambus DRAM відрізняється високою швидкістю, що досягається сукупністю особливостей, які не зустрічаються в інших типах пам'яті. Початкова дуже висока вартість пам'яті RDRAM призвела до того, що виробники потужних комп'ютерів віддали перевагу менш потужній, але дешевшій пам'яті DDR SDRAM. Робочі частоти пам'яті — 400 МГц, 600 МГц і 800 МГц, час повного доступу — до 30 нс, час робочого циклу — до 2,5 нс.

DDR2 SDRAM тип оперативної пам'яті був випущений в 2004 році. Базуючись на технології DDR SDRAM, цей тип пам'яті за рахунок технічних змін показує вищу швидкість. Пам'ять може працювати на частотах 200 МГц, 266 МГц, 333 МГц і 400 МГц. Час її повного доступу — 25 нс, 11,25 нс, 9 нс, 7,5 нс. Час робочого циклу — 5 нс, 3,75 нс, 3 нс, 3,5 нс.

DDR3 SDRAM (англ. double-data-rate three synchronous dynamic random access memory - синхронна динамічна пам'ять з довільним доступом і подвоєною швидкістю передачі даних, третє покоління) - це тип оперативної пам'яті, використовуваної в обчислювальній техніці в якості оперативної і відео-пам'яті. Прийшла на зміну пам'яті типу DDR2 SDRAM.

У DDR3 зменшено на 40% споживання енергії в порівнянні з модулями DDR2, що обумовлено пониженням (1,5 В, в порівнянні з 1,8 В для DDR2 і 2,5 в для DDR) напруги живлення осередків пам'яті. Зниження напруги живлення досягається за рахунок використання 90-нм (спочатку, надалі 65 -, 50 -, 40-нм) техпроцесса при виробництві мікросхем та застосування транзисторів з подвійним затвором Dual-gate (що сприяє зниженню струмів витоку) [8].

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дат		

На даний момент найбільше використовується DDR3 SDRAM так як в порівнянні з іншими видами пам'яті вона, має більш висока пропускну здатність (до 19200 МБ / с), знижений тепловиділення (результат зменшення напруги живлення), а також менше енергоспоживання та поліпшене енергозбереження.

1.2 Статична пам'ять

Статична оперативна пам'ять з довільним доступом (SRAM, static random access memory) - напівпровідникова оперативна пам'ять, в якій кожен двійковий або трійковий розряд зберігається в схемі з позитивним зворотнім зв'язком, що дозволяє підтримувати стан сигналу без постійного перезапису, необхідного в динамічній пам'яті (DRAM). Проте, зберігати дані без перезапису SRAM може тільки поки є живлення, тобто SRAM залишається енергозалежною типом пам'яті.

Недоліками статичної пам'яті є:

- невисока щільність запису (шість-вісім елементів на біт, замість двох у DRAM);
- дорожчі кілобайти пам'яті в порівнянні з DRAM.

Тим не менш, високе енергоспоживання не є принциповою особливістю SRAM, воно обумовлене високими швидкостями обміну з даним видом внутрішньої пам'яті процесора. Енергія споживається тільки в момент зміни інформації в комірці SRAM.

SRAM застосовується в мікроконтролерах і ПЛІС, в яких обсяг ОЗУ невеликий (одиниці кілобайт), зате потрібне низьке енергоспоживання (за рахунок відсутності складного контролера динамічної пам'яті), а також

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дат		

передбаченість з точністю до такту час роботи підпрограм і налагодження прямо на пристрої.

У пристроях з великим об'ємом ОЗУ робоча пам'ять виконується як DRAM. SRAM'ом ж роблять регістри і кеш-пам'ять .

Історія створення статичної пам'яті йде своїм корінням в глибину століть. Пам'ять перших релейних комп'ютерів за своєю природою була статичною і довгий час не зазнавала практично ніяких змін (у всякому разі - концептуальних), змінювалася лише елементарна база: на зміну реле прийшли електронні лампи, згодом витиснені спочатку транзисторами, а потім TTL-і CMOS- мікросхемами, але ідея, що лежить в основі статичної пам'яті, була і залишається незмінною.

Динамічна пам'ять, винайдена, до речі, значно пізніше, в силу фундаментальних фізичних обмежень, так і не змогла зрівнятися зі статичною пам'яттю в швидкості.

Ядро мікросхеми статичної оперативної пам'яті (SRAM - Static Random Access Memory) є сукупність тригерів - логічних пристроїв, що мають два стійких стани, одне з яких умовно відповідає логічному нулю, а інше - логічній одиниці. Іншими словами, кожен тригер зберігає один біт інформації, - рівно стільки ж, скільки і осередок динамічної пам'яті.

Між тим, тригер має мінімум дві переваги перед конденсатором: а) стани у тригера стійкі і за наявності живлення можуть зберігатися нескінченно довго, тоді як конденсатор вимагає періодичної регенерації; б) тригер, володіючи мізерною інертністю, без проблем працює на частотах аж до декількох ГГц, тоді як конденсатори "звалюються" вже на 75-100 МГц.

До недоліків тригерів слід віднести їх високу вартість і низьку щільність зберігання інформації. Якщо для створення осередку динамічної пам'яті достатньо всього одного транзистора і одного конденсатора, то комірка статичної пам'яті складається як мінімум з чотирьох, а в середньому

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дат		

- шести - восьми транзисторів, тому мегабайт статичної пам'яті виявляється щонайменше у декілька разів дорожче.

Типова комірка статичної двійкової пам'яті (двійковий тригер) на CMOS-технології складається з двох перехресно (кільцем) включених інверторів і ключових транзисторів для забезпечення доступу до комірки (рисунок 1.3). Часто для збільшення щільності розміщення елементів на кристалі в якості навантаження застосовують полікремнієві резистори. Недоліком такого рішення є зростання статичного енергоспоживання [9].

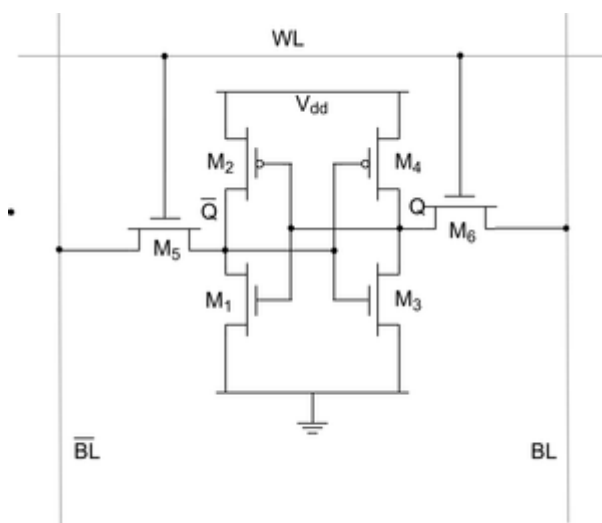


Рисунок 1.3 - Шестирезисторна ячейка статичної двійкової пам'яті SRAM

Лінія WL (Word Line) управляє двома транзисторами доступу. Лінії BL і \overline{BL} (Bit Line) - бітові лінії, які використовуються і для запису даних, і для читання.

Під час запису при подачі «0» на лінію BL або \overline{BL} паралельно включені транзисторні пари (M5 і M1) і (M6 і M3) утворюють логічні схеми 2АБО, подальша подача «1» на лінію WL відкриває транзистор M5 або M6, що призводить до відповідного перемикання тригера .

Під час читання при подачі «1» на лінію WL відкриваються транзистори M5 і M6, рівні записані в тригері виставляються на лінії BL і \overline{BL} і потрапляють на схеми читання.

Перемикання тригерів через транзистори доступу є неявною логічною функцією пріоритетного перемикання, яка в явному вигляді, для двійкових тригерів, будується на двохходових логічних елементах 2АБО-НЕ або 2І-НЕ. Схема комірки з явним перемиканням є звичайним RS-тригером. При явній схемі перемикання лінії читання і запису розділяються, відпадає потреба в транзисторах доступу (по 2 транзистори на 1 клітинку), але в самій комірці потрібні транзистори з двома затворами.

В даний час з'явилася вдосконалена схема з відключенням сигналу запису зворотнього зв'язку, яка не вимагає транзисторів навантаження і відповідно позбавлена від високого споживання енергії при записі.

В основі всіх тригерів лежить кільце з двох логічних елементів "НЕ" (інверторів), сполучених за типом "засувки" (рисунок 1.4). Якщо подати на лінію Q сигнал, відповідний одиниці, то пройшовши крізь елемент D.D1 він звернеться в нуль. Але, вступивши на вхід наступного елементу D.D2 цей нуль знов перетвориться на одиницю. Оскільки вихід елементу D.D2 підключений до входу елементу D.D1, то навіть після зникнення сигналу з лінії \overline{Q} , він підтримуватиме себе самостійно, тобто тригер перейде в стійкий стан[9].

Природно, якщо на лінію Q подати сигнал, відповідний логічному нулю, все буде відбуватися так само, але навпаки.

Подібно осередкам динамічної пам'яті, тригери об'єднуються в єдину матрицю, що складається з рядків (row) і стовпців (column), останні з яких так само називаються бітами (bit).

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дат		

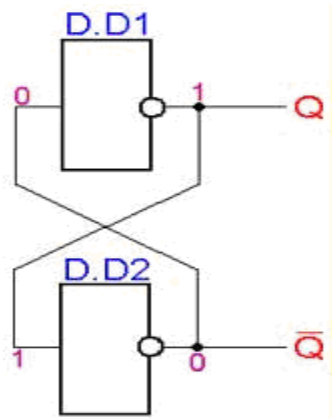


Рисунок 1.4 - Пристрій найпростішого тригера

На відміну від осередку динамічної пам'яті, для управління якої достатньо всього одного ключового транзистора, осередок статичної пам'яті управляється як мінімум двома. Це відбувається через те, що тригер, на відміну від конденсатора, має роздільні входи для запису логічного нуля і одиниці відповідно. Таким чином, на осередок статичної пам'яті витрачається цілих вісім транзисторів (рисунок 1.5) - чотири йдуть, власне, на сам тригер і ще два - на керуючі "засувки" [9].

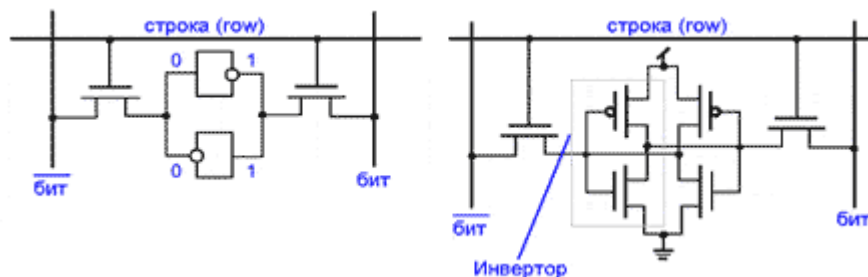


Рисунок 1.5 - Пристрій 6-транзисторної одно-портової ячейки SRAM-пам'яті

Основний недолік шести транзисторної комірки полягає в тому, що в кожен момент часу може оброблятися всього лише один рядок матриці пам'яті. Паралельне читання комірок, розташованих в різних рядках одного і

того ж банку неможливо, так само як неможливо і читання одного осередку одночасно із записом іншого.

Цього обмеження позбавлена багато-портова пам'ять. Кожна комірка багато-портової пам'яті містить один-єдиний тригер, але має кілька комплектів керуючих транзисторів, кожний з яких підключено до "своїх" ліній ROW і BIT, завдяки чому різні комірки матриці можуть оброблятися незалежно. Такий підхід набагато прогресивніший, ніж ділення пам'яті на банки. Адже, в останньому випадку паралелізм досягається лише при зверненні до осередків різних банків, що не завжди можна здійснити, а багато-портова пам'ять допускає одночасну обробку будь-яких осередків, позбавляючи програміста від необхідності вникати в особливості її архітектури [9].

Частіше зустрічається двох-портова пам'ять, пристрій комірки якої зображено на рисунку 1.6. Неважко підрахувати, що для створення одного осередку двох-портової пам'яті витрачається вісім транзисторів. Нехай ємність кеш-пам'яті становить 32 Кб, тоді тільки на одне ядро піде понад два мільйони транзисторів.

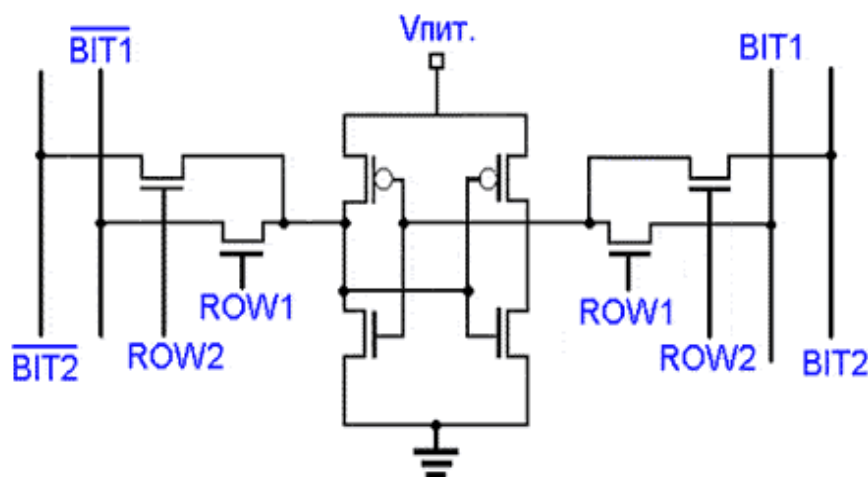


Рисунок 1.6 - Пристрій 8-транзисторної двох портової ячейки SRAM-пам'яті

Цикл читання починається зі скидання сигналу CS (Chip Select - Вибір Чіпа) в низький стан, даючи зрозуміти тим самим мікросхемі, що чіп "вибраний" і зараз з ним працюватимуть.

До того моменту, коли сигнал стабілізується, на адресних лініях повинен перебувати готова до експлуатації адреса комірок (тобто номер рядка і номер стовпця), а сигнал WE має бути переведений у високий стан (відповідно операції читання комірок). Рівень сигналу OE (Output Enable - дозвіл виводу) не грає ніякої ролі, тому що на виході поки нічого не міститься, точніше вихідні лінії знаходяться в так званому високо імпедансному стані.

Через деякий час (tAddress Access), який визначається швидкістю керуючої логіки і швидкоплинністю перехідних процесорів в інверторі, на лініях виходу з'являються довгоочікувані дані, які аж до закінчення робочого циклу (tCycle) можуть бути безпосередньо порашовані. Звичайний час доступу до осередку статичної пам'яті не перевищує 1 - 2 нс, а часто буває і менше.

Цикл запису відбувається в зворотному порядку. Спочатку виставляється на шину адрес записуваного осередку і одночасно з цим скидається сигнал WE в низький стан. Потім, дочекавшись, коли адреса декодується і надійде на відповідні бітові лінії, скидається CS в низький рівень, наказуючи мікросхемі подати сигнал високого рівня на необхідну лінію row. Засувка, що утримує тригер, відкриється і залежно від стану bit-лінії, тригер перемкнеться в той або інший стан.

На рисунку 1.7 подані тимчасові діаграми читання та запису асинхронної статичної пам'яті.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дат		

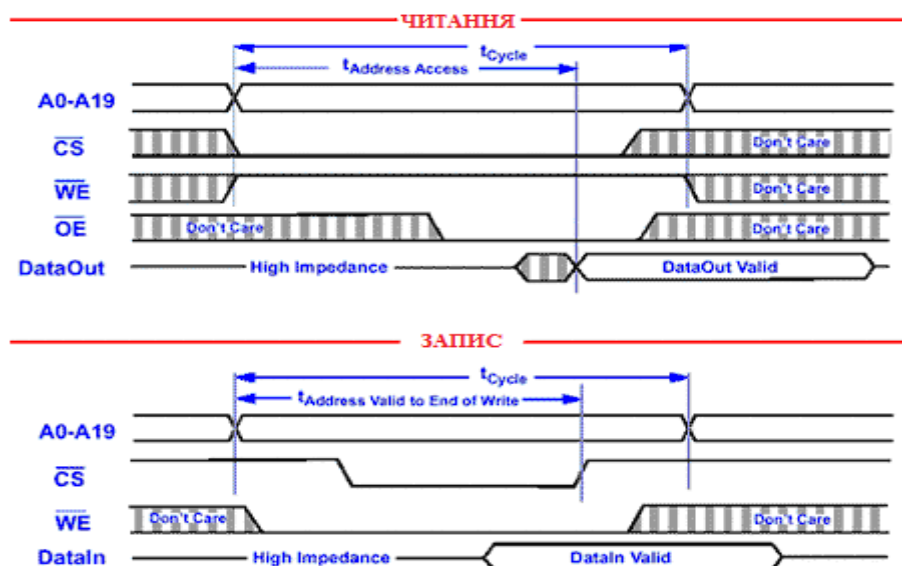


Рисунок 1.7 - Тимчасові діаграми читання / запису асинхронної статичної пам'яті

1.3 Постановка задачі

Статична оперативна пам'ять є однією з найбільш використовуваних і вона може працювати без постійного перезапису, який необхідний у динамічній пам'яті.

Тому розробка моделі статичної пам'яті є актуальною задачею.

Метою даного дипломного проекту є аналіз статичної оперативної пам'яті з довільним доступом а також проектування її HDL – моделі.

Під час здійснення проектування потрібно вирішити наступні задачі:

- 1) проаналізувати основні характеристики оперативної пам'яті та її призначення;
- 2) дослідити відмінність динамічної та статичної пам'яті;
- 3) здійснити аналіз побудови та принципів функціонування статичної пам'яті з довільним доступом;
- 4) вибрати середовище проектування досліджуваної SRAM;

- 5) побудувати HDL – модель статичної пам'яті з довільним доступом;
- 6) здійснити верифікацію проекту;
- 7) дослідити основні вимоги до охорони праці.

Програмою для моделювання статичної пам'яті варто обрати Aldec Active-HDL, в якій є повністю інтегрованим середовищем розробки цифрових пристроїв. Вона побудована на основі ядра логічного моделювання, що підтримує текстовий і графічний метод введення проектів на суміші мов VHDL, Verilog, EDIF, SystemC, SystemVerilog. Система проектування Active-HDL є сьогодні провідним середовищем проектування для створення проектів та їх моделювання для всіх програмованих інтегральних схем ПЛІС, забезпечуючи гнучкість підходу і надаючи розвинені функції підтримки найбільш складних сучасних проектів. Середовище проектування Active-HDL дає розробникам незалежність у використанні інших засобів проектування з єдиного, повністю інтегрованого оточення. Крім того, створений у Active-HDL проект можна про симулювати та здійснити його верифікацію.

Спроектована статична оперативна пам'ять може застосовуватися у запам'ятовуючих пристроях, використовуючи функцію запам'ятовування біта інформації. При цьому вони зберігають свої традиційні переваги – велику інформаційну ємність та дуже мале енергоспоживання – до долей мікрвольта на один елемент пам'яті. Ця функція використовується в RS – тригерах за схемотехнікою КМОН (комплементарний метал-оксидний-напівпровідник). Також вона широко використовується в кеш-пам'яті, яка повинна мати максимально можливу швидкодію.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дат		

2 ПОБУДОВА ТА ПРИНЦИПИ ФУНКЦІОНУВАННЯ СТАТИЧНОЇ ПАМ'ЯТІ З ДОВІЛЬНИМ ДОСТУПОМ

2.1 Основні характеристики SRAM

Основними характеристиками оперативної пам'яті є:

1) латентність (затримка при передачі даних) - у модулів SIMM вона становила близько 60 нс., у перших DIMM-модулів 8-10 нс., у сучасній оперативній пам'яті DIMM DDR, DIMM DDR2 - 1,6-2,5 нс [10].

2) висока габаритність, обсяг одного модуля оперативної пам'яті може досягати 1 Гб. Стандартний обсяг сучасної оперативної пам'яті: 64, 128, 256, 512, 1024 Мб.

Оперативна пам'ять - це робоча область для процесора комп'ютера. У ній під час роботи зберігаються програми і дані. Оперативна пам'ять часто розглядається як тимчасове сховище, тому що дані і програми в ній зберігаються тільки при включеному комп'ютері або до натискання кнопки скидання (reset). Перед вимиканням або натисканням кнопки скидання всі дані, піддані змінам під час роботи, необхідно зберегти на запам'ятовуючому пристрої, який може зберігати інформацію постійно (зазвичай це жорсткий диск). При новому включенні живлення збережена інформація знов може бути завантажена в пам'ять.

Пристрої оперативної пам'яті іноді називають запам'ятовуючими пристроями з довільним доступом. Це означає, що звернення до даних, що зберігаються в оперативній пам'яті, не залежить від порядку їх розташування в ній. Коли говорять про пам'ять комп'ютера, зазвичай мають на увазі оперативну пам'ять, перш за все мікросхеми пам'яті або модулі, в яких зберігаються активні програми і дані, що використовуються процесором.

За кілька років визначення RAM (Random Access Memory) перетворилося зі звичайної аббревіатури в термін, що позначає основний

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дат		

робочий простір пам'яті, створюване мікросхемами динамічної оперативної пам'яті (Dynamic RAM - DRAM), і використовується процесором для виконання програм. Одним із властивостей мікросхем DRAM (і, отже, оперативної пам'яті в цілому) є динамічне зберігання даних, що означає, по-перше, можливість багаторазового запису інформації в оперативну пам'ять, а по-друге, необхідність постійного оновлення даних (тобто, по суті, їх перезапис) приблизно кожні 15 мс. Також існує так звана статична оперативна пам'ять (Static RAM - SRAM), яка не вимагає постійного оновлення даних. Слід зауважити, що дані зберігаються в оперативній пам'яті тільки при включеному живленні.

Під комп'ютерною пам'яттю зазвичай мається на увазі ОЗП (RAM), тобто фізична пам'ять системи, яка складається з мікросхем або модулів пам'яті, які використовуються процесором для зберігання основних, запущених в поточний момент часу, програм і даних. При цьому термін "сховище даних" відноситься не до оперативної пам'яті, а до таких пристроїв як жорсткі диски і накопичувачі на магнітній стрічці (які, тим не менш, можна використовувати як різновид RAM, що отримала назву віртуальна пам'ять).

Термін "оперативна пам'ять" часто позначає не тільки мікросхеми, які складають пристрої пам'яті в системі, але включає і такі поняття, як логічне відображення і розміщення. Логічне відображення - це спосіб представлення адрес пам'яті на фактично встановлених мікросхемах. Розміщення - це розташування інформації (даних і команд) певного типу за конкретними адресами пам'яті системи.

Під час виконання програми в оперативній пам'яті зберігаються її дані. Мікросхеми оперативної пам'яті (RAM) іноді називають енергозалежною пам'яттю: після вимикання комп'ютера дані, що зберігаються в них, будуть втрачені, якщо вони попередньо не були збережені на диску або іншому пристрої зовнішньої пам'яті. Щоб уникнути цього, деякі програми

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дат		

автоматично роблять резервні копії даних.

Файли комп'ютерної програми при її запуску завантажуються в оперативну пам'ять, в якій зберігаються під час роботи із зазначеною програмою. Процесор виконує програмно реалізовані команди, що містяться в пам'яті, і зберігає їх результати. Оперативна пам'ять зберігає коди натиснутих клавіш при роботі з текстовим редактором, а також величини математичних операцій. При виконанні команди “зберегти” (Save) вміст оперативної пам'яті зберігається у вигляді файлу на жорсткому диску.

Фізично оперативна пам'ять в системі являє собою набір мікросхем або модулів, що містять мікросхеми, які зазвичай підключаються до системної плати. Ці мікросхеми або модулі можуть мати різні характеристики і, щоб функціонувати правильно, повинні бути сумісні із системою, в яку встановлюються.

У сучасних комп'ютерах використовуються запам'ятовуючі пристрої трьох основних типів:

- ROM (Read Only Memory) - постійний запам'ятовуючий пристрій - ПЗП, не здатний виконувати операцію запису даних;
- RAM (Dynamic Random Access Memory) - динамічний запам'ятовуючий пристрій з довільним порядком вибірки;
- SRAM (Static RAM) - статична оперативна пам'ять [11].

2.2 Типи SRAM

Async SRAM - звичайна (стандартна) асинхронна статична пам'ять (Standard або Asynchronous SRAM). Цей тип мається на увазі під терміном SRAM за умовчанням, коли тип пам'яті не вказаний (до недавніх пір йому і не було альтернативи).

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дат		

Мікросхеми цього типу мають найпростіший асинхронний інтерфейс, що включає шину адреси, шину даних і сигнали управління CS #, OE # і WE #. Мікросхема вибирається низьким рівнем сигналу CS # (Chip select), низький рівень сигналу OE # (Output Enable) відкриває вихідні буфери для зчитування даних, WE # (Write Enable) низьким рівнем дозволяє запис. Тимчасові діаграми циклів звернення наведено на рисунку 2.1. При операції запису управління вихідними буферами може вироблятися як сигналом OE # (цикл 1), так і сигналом WE # (цикл 2). Для зручності об'єднання мікросхем внутрішній сигнал CS # може збиратися за схемою "І" з декількох зовнішніх, наприклад, CS0 #, CS1 і CS2 # - в такому разі мікросхема буде вибрана при поєднанні логічних сигналів 0, 1, 0 на відповідних входах [12].

Час доступу - затримка появи дійсних даних на виході щодо моменту встановлення адреси - у стандартних мікросхем SRAM складає 12, 15 або 20 наносекунд, що дозволяє процесору виконувати пакетний цикл читання 2-1-1-1 (тобто без тактів очікування) на частоті системної шини до 33 МГц. На більш високих частотах цикл буде не краще 3-2-2-2.

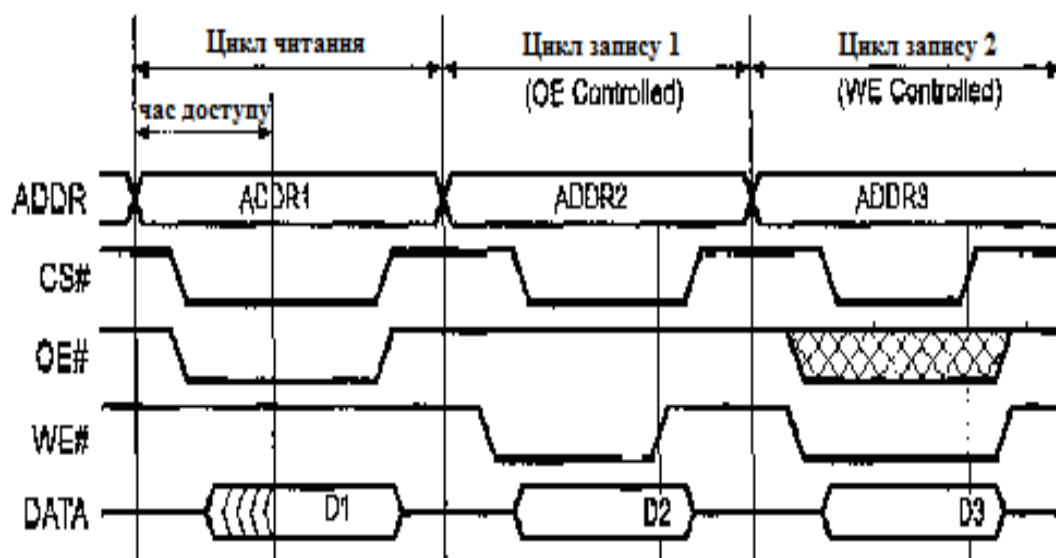


Рисунок 2.1 – Часова діаграма зчитування та запису асинхронної статичної пам'яті

Sync Burst SRAM - синхронна статична пам'ять, оптимізована під виконання пакетних (burst) операцій обміну, властивих роботі кеш-пам'яті. В її структуру введено внутрішній двобітний лічильник адреси. На додаток до сигналів, які використовуються асинхронною пам'яттю (адреса, дані, CS #, OE # і WE #), пам'ять використовує сигнал CLC (Clock) для синхронізації з системною шиною і сигнали управління пакетним циклом ADSP #, CADS # і ADV #. Сигнали CADS # (Cache Address Strobe) і ADSP # (Address Status of Processor), якими процесор або кеш-контролер зазначає фазу адреси чергового циклу, є спробою запису початкової адреси циклу у внутрішній регістр адреси. Будь-який з цих сигналів ініціює цикл обігу, одиночний (single) або пакетний (burst), а сигнал ADV # (ADVance) використовується для переходу до наступної адреси пакетного циклу. Усі сигнали, окрім управління вихідними буферами OE #, синхронізуються по позитивному перепаду сигналу CLK. Це означає, що значення вхідних сигналів має встановитися до перепаду і утримуватися після нього ще деякий час. Вихідні дані при зчитуванні будуть також дійсні під час цього перепаду. На рисунку 2.2 приведені діаграми декількох варіантів циклів читання синхронної статичної пам'яті. Звернемо увагу, що двобітний лічильник адреси не дозволяє перейти кордон чотирьохелементною пакетного циклу. Крім того, порядок рахунку адрес усередині пакетного циклу відповідає специфічному порядку (interleaved), прийнятому в процесорах i486 і старше. Мікросхеми синхронної статичної пам'яті, як і SDRAM, зазвичай мають сигнал, що вибирає режим рахунку адреси: чергування (для процесорів Intel) або послідовний рахунок (для Power PC).

Синхронний інтерфейс з таким набором сигналів дозволяє пам'яті дізнаватися про наміри процесора раніше і при затримці даних на виході SRAM щодо синхронізуючого перепаду ТКQ (Clock-to-Output Access Time) 8,5, 10 і 13,5 нс забезпечувати цикл 2-1-1 -1 на частотах 66, 60 і 50 МГц відповідно. Однак на частотах 75 МГц і вище цикл буде 3-2-2-2.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дат		

PB SRAM (Pipelined Burst SRAM) - конвеєрне вдосконалення синхронної пам'яті (слово "синхронна" з її назви для стислості вилучили, але воно обов'язково мається на увазі). Конвеєром є додатковий внутрішній регістр даних, який, вимагаючи додаткового такту в першій пересилці циклу, дозволяє інші дані отримувати без тактів очікування навіть на частотах вище 75 МГц. Затримка даних щодо синхронізуючого перепаду у сучасних мікросхем PB SRAM складає 4,5-8 нс. Але, як і у випадку Sync Burst SRAM, цей параметр не є часом доступу в чистому вигляді (не слід забувати про два-три такти в першій передачі), а відображає появу дійсних даних щодо чергового перепаду сигналу синхронізації. Інтерфейс PB SRAM аналогічний інтерфейсу Sync Burst SRAM.

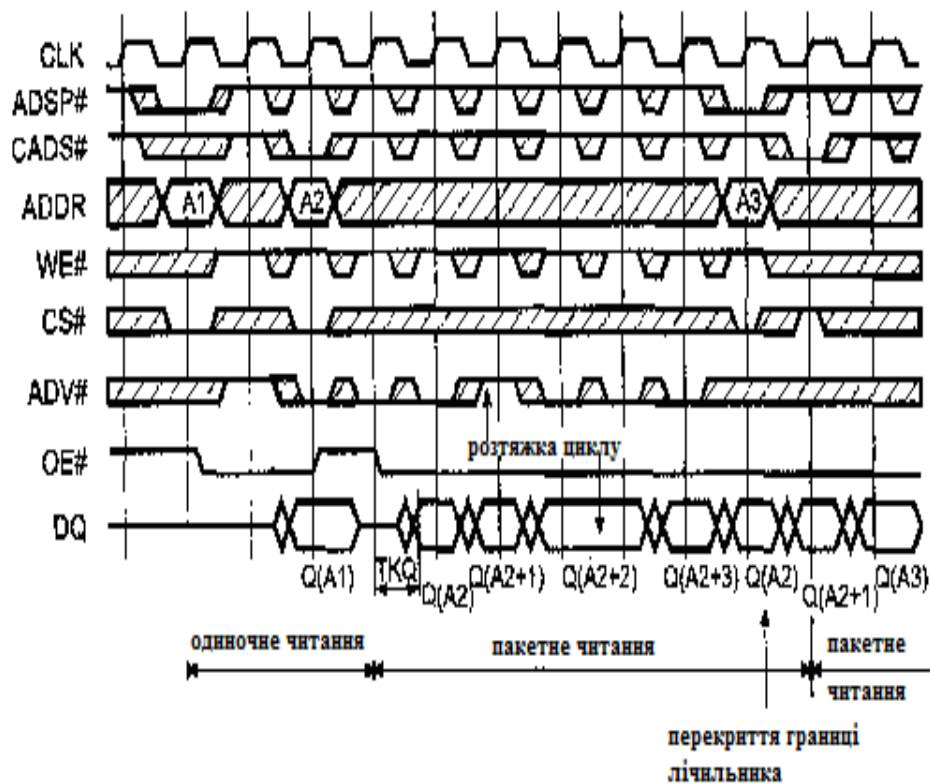


Рисунок 2.2 – Часові параметри читання статичної пам'яті

Природно, що ціна синхронної пам'яті вище, ніж асинхронної. Ефективні області застосування описаних різновидів статичної пам'яті у вторинному кеші допоможе визначити таблиця 2.1.

Таблиця 2.1. Параметри пакетних циклів SRAM

Частота шини	Async SRAM		Sync Burst SRAM		PBSRAM	
	Цикл	T_{AC} , нс	Цикл	T_{KO} , нс	Цикл	T_{RC} , нс
33	2-1-1-1	15	2-1-1-1		3-1-1-1	-
50	3-2-2-2	20	2-1-1-1	13,5	3-1-1-1	-
60	3-2-2-2	17	2-1-1-1	10	3-1-1-1	-
66	3-2-2-2	15	2-1-1-1	8,5	3-1-1-1	15
75	-	-	3-2-2-2		3-1-1-1	13
83	-	-	3-2-2-2		3-1-1-1	12
100	-		3-2-2-2			10
133	-		3-2-2-2			7,5

* Для різних типів пам'яті швидкодію вказується різними способами:

ТАС - час доступу,

ТКQ - затримка даних щодо синхронізуючого перепаду,

ТКС - мінімальний період тактових імпульсів.

2.3 Будова SRAM

Трійкова SRAM може бути:

- 1) однопровідною трирівневою
- 2) двухпровідною дворівневою (двобітною) (звичайні два двійкові тригери, четвертий стан або не використовується зовсім або використовується для інших цілей);
3. трьохпровідною дворівневою (трьохбітною)

Трійчастий дворівневий трьохрозрядний тригер на трьох двухвходових елементах 2 АБО-НЕ і на трьох двухвходових елементах 2 І-НЕ можна використовувати для побудови часткової трійкової пам'яті, подібної двійкової SRAM (рисунок 2.3) [12]

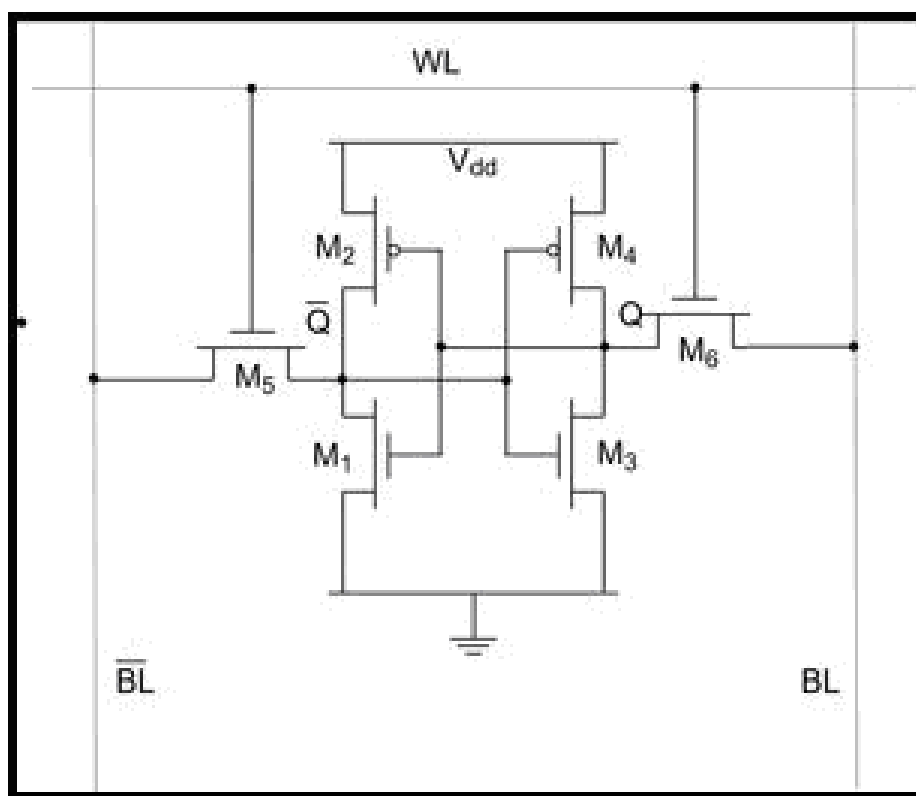


Рисунок 2.3 – Шеститранзисторний регістр статичної двійкової пам'яті (біт) SRAM

Для цього необхідно три двовходових КМОП елементи (інвертори), що виконують логічну функцію 2 АБО-НЕ або 2 І-НЕ і три транзистора доступу. Зазвичай елементи 2 АБО-НЕ або 2 І-НЕ будуються на чотирьох КМОП транзисторах (рисунок 2.4);

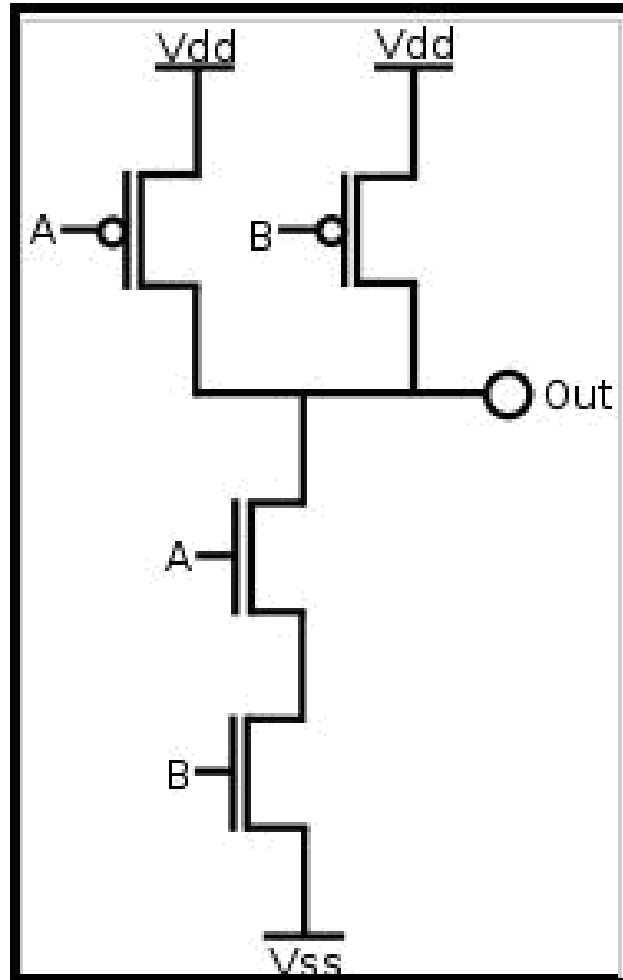


Рисунок 2.4 – Схема 2 І-НЕ

Три таких елементи складаються з 12 транзисторів, плюс 3 транзистора доступу, підключені до трьох виходів Q0, Q1, Q2. Всього 15 транзисторів на одну клітинку. Насправді елементи 2 І-НЕ і 2 АБО-НЕ будуються не на чотирьох окремих транзисторах, а на двох комплементарних парах транзисторів, одна з яких, з паралельним включенням двох транзисторів T1 і T2, виконує логічну функцію 2АБО, а друга, з послідовним увімкненням двох транзисторів T3 і T4, виконує логічну функцію 2І(рисунок 2.5)[12].

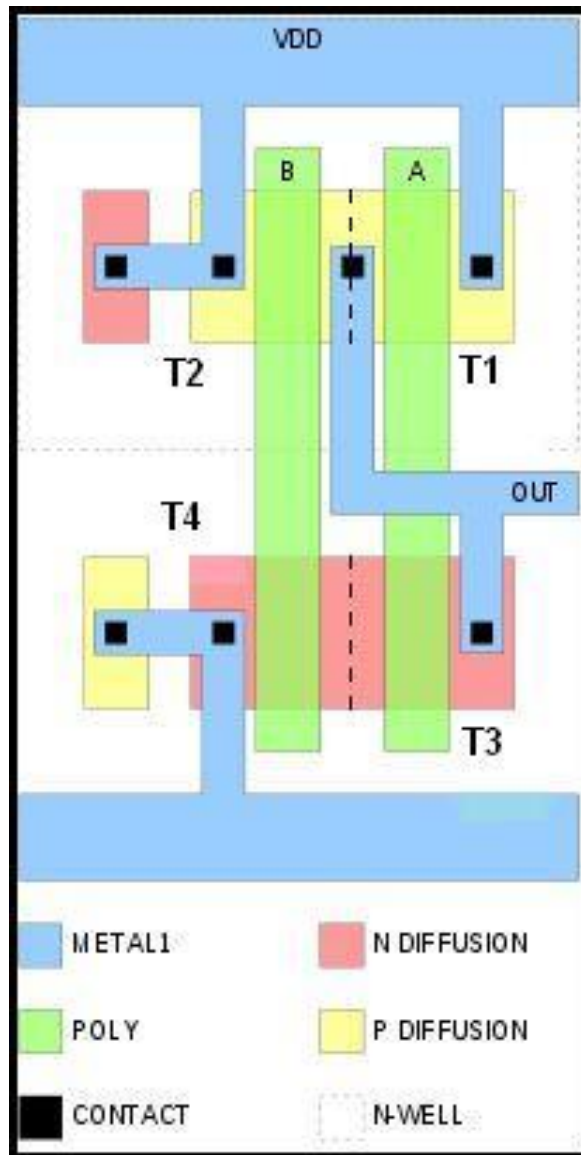


Рисунок 2.5 – Схема 2 І-НЕ та 2 АБО-НЕ

Якщо у верхній частині стоїть пара транзисторів, які виконують логічну функцію 2 АБО, а в нижній частині стоїть пара транзисторів, які виконують логічну функцію 2 І, то послідовне включення двох таких пар транзисторів виконує логічну функцію 2 І-НЕ (2 І-інвертор).

Якщо у верхній частині стоїть пара транзисторів, які виконують логічну функцію 2 І, а в нижній частині стоїть пара транзисторів, які виконують логічну функцію 2 АБО, то послідовне включення двох таких пар транзисторів виконує логічну функцію 2 І без інверсії, тобто логічну функцію 2 І (2 І-повторювач).

Таким чином, в дійсності, одна трійкова трьохбітна частина пам'яті (трійкова SRAM) складається з 6-ти двозатворних польових транзисторів і 3-х однозатворних польових транзисторів доступу, всього 9-ть транзисторів на одну клітинку (третію). Разом на 18-ти транзисторах можна побудувати або 3 двійкові частини пам'яті (SRAM), які можуть зберігати код, або 2 трійкових трьохбітних осередків трійкової трьохбітної пам'яті, які можуть зберігати код, що на 1 код більше. Зі збільшенням числа транзисторів різниця зростає ще більше.

Таким чином три двохходових елемента 2 І-НЕ і 2 АБО-НЕ складаються з 6 двозатворних транзисторів, плюс 3 транзистори доступу. Всього 9 транзисторів на одну клітинку. У другому випадку (9 транзисторів на Тріт), порівняно зі звичайною двійковою шеститранзисторною частиною статичної пам'яті SRAM, вже при трьох осередках питома ємність трійкової пам'яті перевершує питому ємність двійкової пам'яті.

При 18 транзисторах двійкова пам'ять має ємність $2^3 = 8$ кодів (чисел) на 18 транзисторів ($8/18 = 0,444 \dots$ [коду / транзистор]), а трійкова $3^2 = 9$ кодів на 18 транзисторів ($9/18 = 0,5$ [коду / транзистор]), при подальшому збільшенні числа транзисторів питома ємність трійкової пам'яті збільшується ще більше.

При 18 транзисторах, 3 двійкових або 2 трійкових розрядів, ($2^3 = 8$ кодів, $3^2 = 9$ кодів) збільшення ємності SRAM в $9/8 = 1,125$ рази (на 12,5%).

При 36 транзисторах, 6 двійкових або 4 трійкових розрядів, ($2^6 = 64$ кодів, $3^4 = 81$ кодів) збільшення ємності SRAM в $81/64 = 1,267$ рази (на 26,7%).

При 72 транзисторах, 12 двійкових або 8 трійкових розрядів, ($2^{12} = 4096$ кодів, $3^8 = 6561$ кодів) збільшення ємності SRAM в $6561/4096 = 1,6$ рази (на 60%).

При 144 транзисторах, 24 двійкових або 16 трійкових розрядів, збільшення ємності SRAM в $43046721/16777216 = 2,57$ рази (на 157%).

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		41

При 288 транзисторах, 48 двійкових або 32 трійкових розрядів, збільшення ємності SRAM в $1\ 853\ 020\ 188\ 851\ 841/281\ 474\ 976\ 710\ 656 = 6,58$ рази (на 558%) [13].

При застосуванні непозиційної унарної трійкової системи числення, апаратні витрати активних елементів (транзисторів) в трійковій трьохбітній пам'яті не збільшуються, а надійність зберігання даних зростає, так як при несправності одного з трьох бітових (BL) ліній за двома справними бітовими лініями (BL) можна повністю відновити записаний в пам'ять код. При застосуванні позиційної трійкової системи числення апаратні витрати активних елементів зменшуються в $2 * \ln 3 / (3 * \ln 2) = 1,057$ рази, тобто на 5,7%. Надійність зберігання даних збільшується так само, як і в першому випадку.

У наведених схемах трійкової SRAM(рисунку 2.6 та рисунку 2.7), як і в осередках двійкової SRAM, застосовуються схеми неявного пріоритетного перемикування. В схемі з явним пріоритетним перемикачем логічні елементи мають три входи, тобто є трьохходовими (3 АБО-НЕ, 3 І-НЕ). При цьому шини запису і читання розділяються, виникає потреба в трьох транзисторах доступу, але на одиницю збільшується число входів в одному логічному елементі, тобто в сумі, в логічних частинах всіх трьох елементів додаються три діода (в ТТЛ). В таких застосуваннях осередків SRAM-пам'яті, як зберігання числа кодів або чисел, питома ємність при переході до трійкової SRAM значно збільшується.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дат		

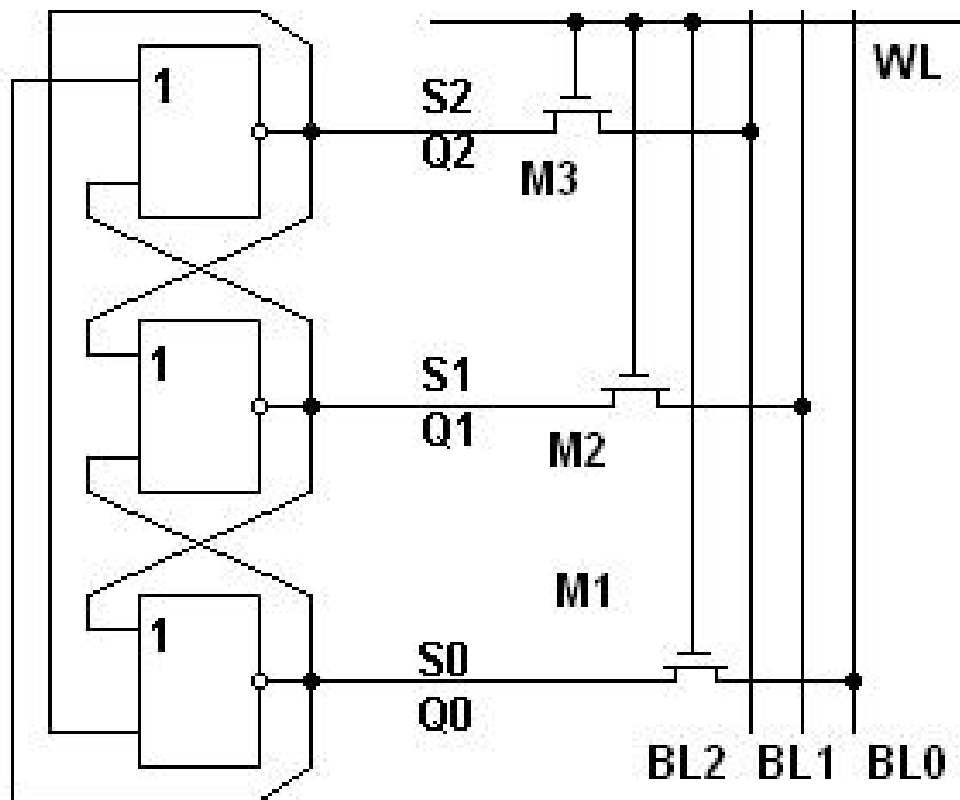


Рисунок 2.6 - Схема трійкової трьохбітної частини пам'яті (TSRAM) на трійковому трьохбітному тригері з трьох логічних елементів 2 АБО-НЕ.

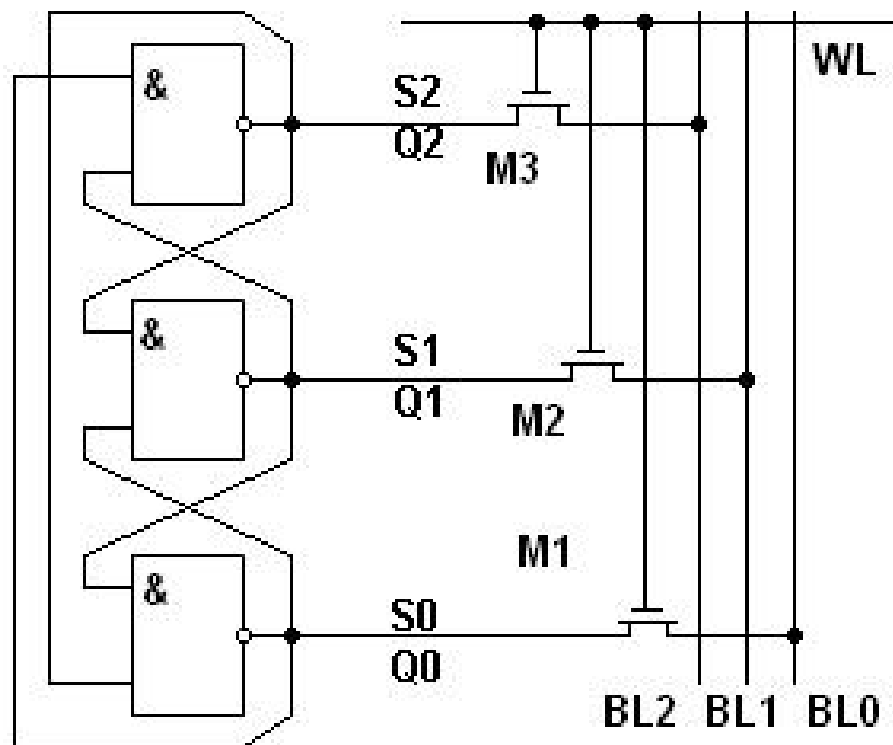


Рисунок 2.7 - Схема трійкової трьохбітної частини пам'яті на трійковому трьохбітному тригері з трьох логічних елементів 2 І-НЕ.

Змн.	Арк.	№ докум.	Підпис	Дат

В дисплейній пам'яті на 18-ти транзисторах можна побудувати три двійкових 6-ти транзисторних частин пам'яті, які при двох градаціях яскравості (чорне, біле) мають $2^3 = 8$ станів, або два трійкових 9-ти транзисторних елементів пам'яті, які при трьох градаціях яскравості (чорне, сіре, біле) мають $2^3 = 8$ станів, тобто ємність пам'яті збільшується на одну одиницю, при подальшому збільшенні числа транзисторів різниця в ємностях збільшується ще більше.

При застосуванні D-тригер або частини регістру / пам'яті варто, використовувати каскад з послідовно включених ключів, один з прямим, другий з інверсним входом управління, замкнутими на середню точку між ними для зберігання одного біта двійкової інформації. В цьому випадку живлення подається через сигнали WR'і WR. При відключенні або точніше ізоляції їх від живлячих напруг можливо короткочасне зберігання інформації, як в комірці динамічної пам'яті, при цьому можливий також перезапис. Дані знімаються під час читання або подаються для запису прямо з/на середню точку.

Для використання тригера в матриці необхідні для кожного осередку додатково чотири (рисунок 2.8б) або три (рисунок 2.8в) ключа для вибірки по координатних осях та управління записом.

Можлива реалізація подібної частини на КМДП-транзисторах з вбудованим каналом. Ключі обрамлення можуть являти собою звичайні КМДП-транзистори з індукованим каналом, реалізація яких особливо проста.

Для реалізації на КМДП-транзисторах з індукованим каналом необхідно додати ще один каскад (рисунок 2.9, на рисунок 2.9б показана схема з додатковими ключами для двокоординатної вибірки). Ключі для двокоординатної вибірки і додаткові ключі на рисунок 2.8б можуть бути при застосуванні двозатворних МДП-транзисторів з послідовно розташованими затворами (рисунок 2.10) згруповані попарно, що зменшить кількість ключів або транзисторів на рисунок 2.8б до трьох.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дат		

Як видно, каскади на рисунку 2.8 являють собою формувачі без інверсії, а на рисунку 2.9 - інвертори, в результаті отримуємо схеми на рисунку 4. На рисунку 4а та 4б нижні ключі намальовані з інверсним керуванням для можливості об'єднання ключів за допомогою двозатворних транзисторів, тобто при відсутності вони можуть бути і з прямими входами [14].

Якщо зазначені схеми використовуються в пристроях з іншим принципом дії, наприклад на біполярних транзисторах або оптичних ключах, то при відсутності ефекту ємності на входах елементів необхідно сигнали WR і WR' теж подавати через схеми управління, аналогічно лінії DIO (один дозволяє ключ для регістра або послідовні два ключі для двокоординатної вибірки).

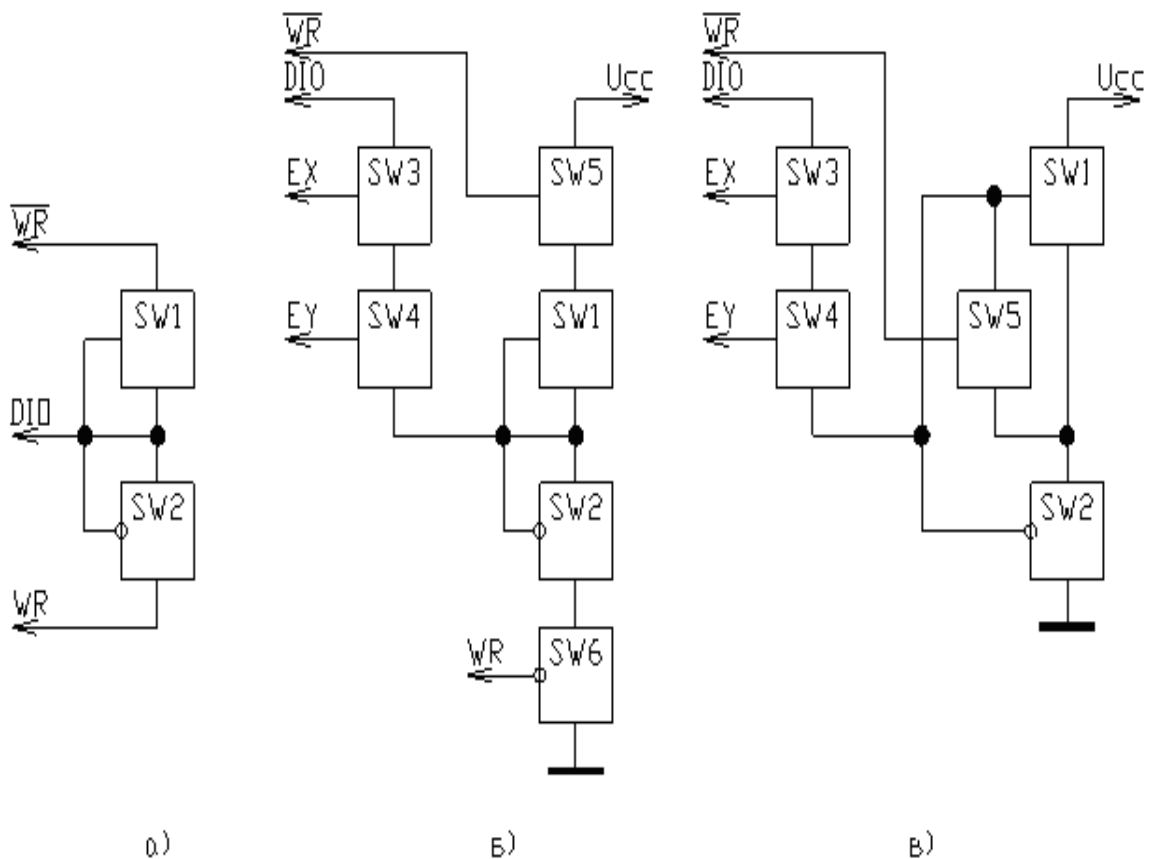


Рисунок 2.8 – Схеми осередків тригерів

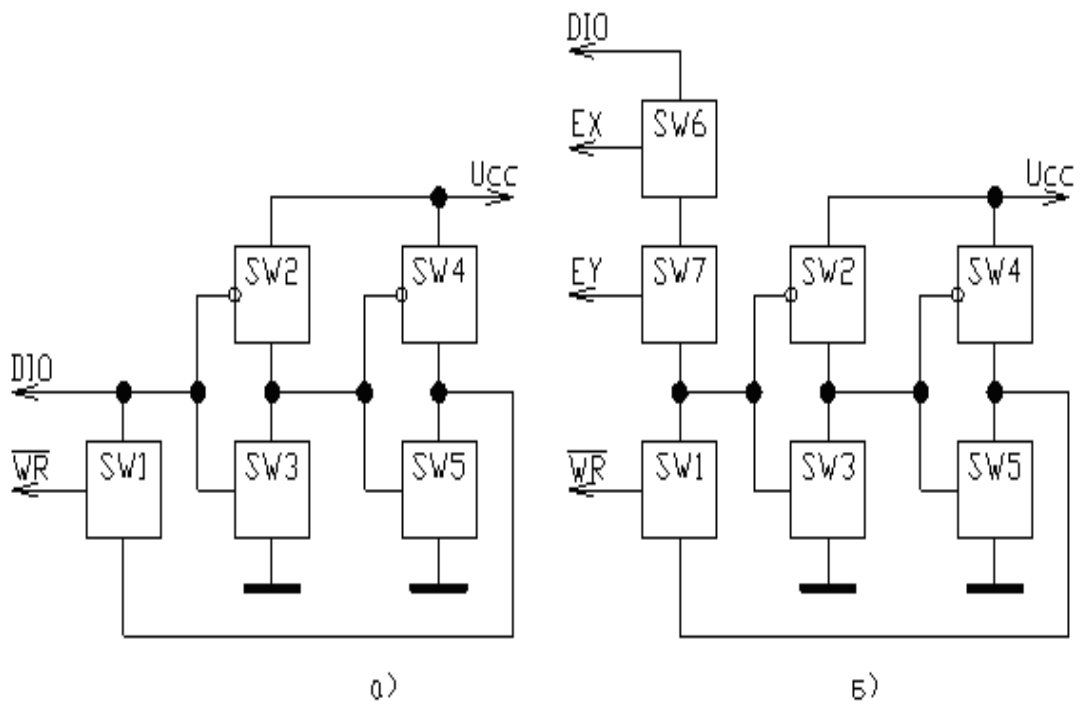


Рисунок 2.9 – Каскад транзисторів

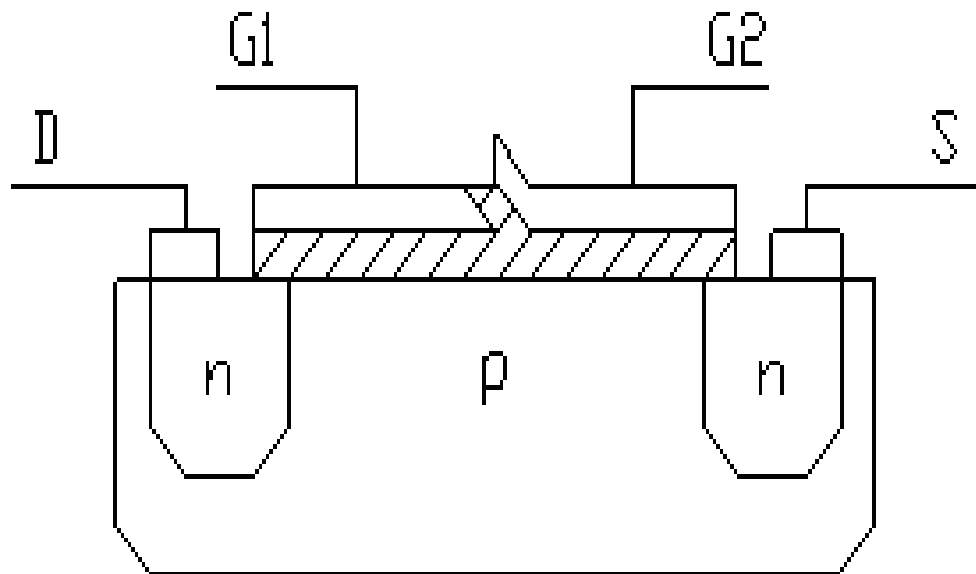


Рисунок 2.10 – Схема застосування МДП - транзисторів

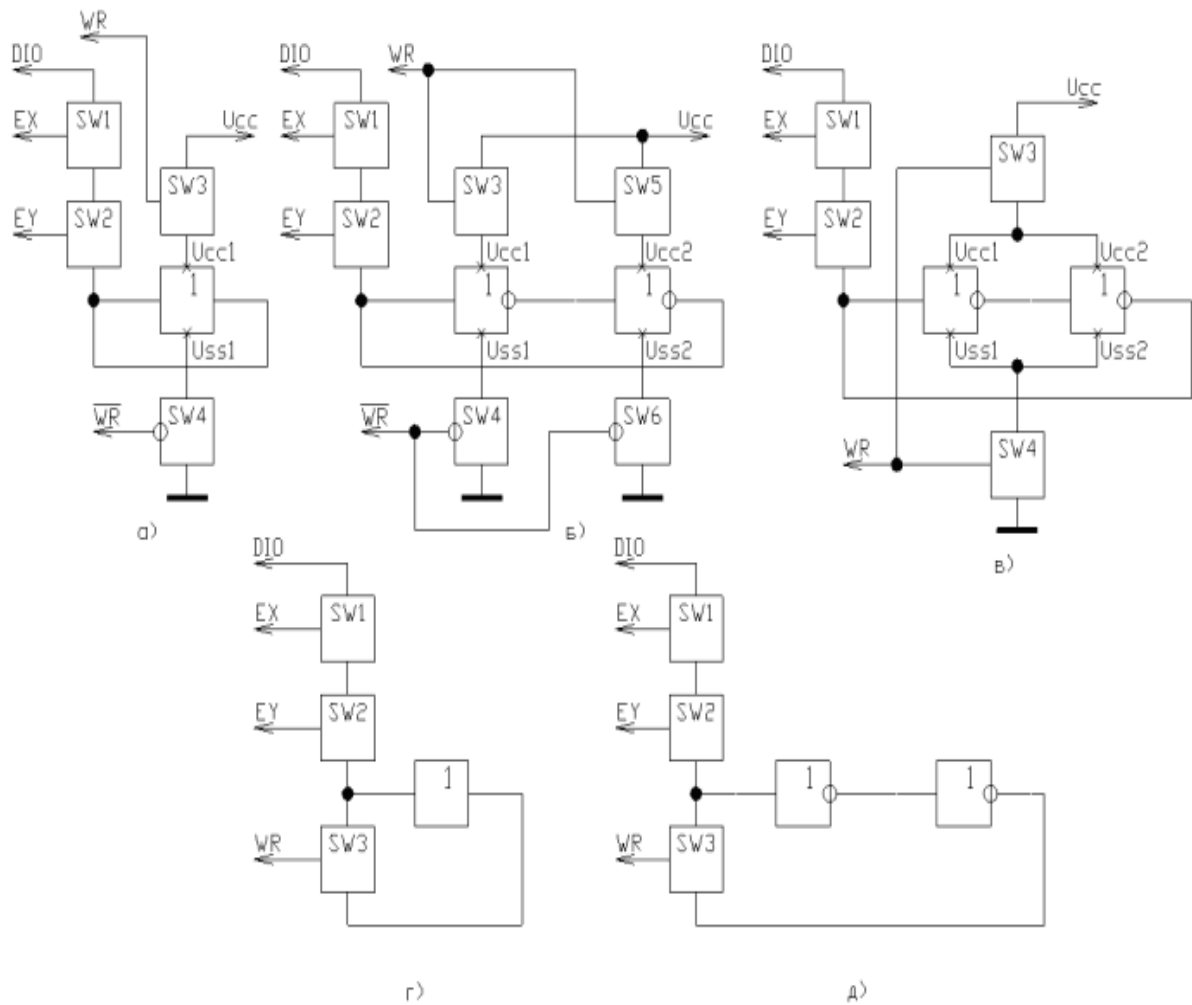


Рисунок 2.11- Частини трійкової дворівневої трьохпроводної (трьохбітної) SRAM-пам'яті (трійкові трьохбітні D-тригери)

Трійкові трьохбітні тригери на трьох двовходових елементах 2 АБО-НЕ і на трьох двовходових елементах 2 І-НЕ можна використовувати для побудови осередків трійкової пам'яті, подібних частинам двійкової SRAM(рисунок 2.12).

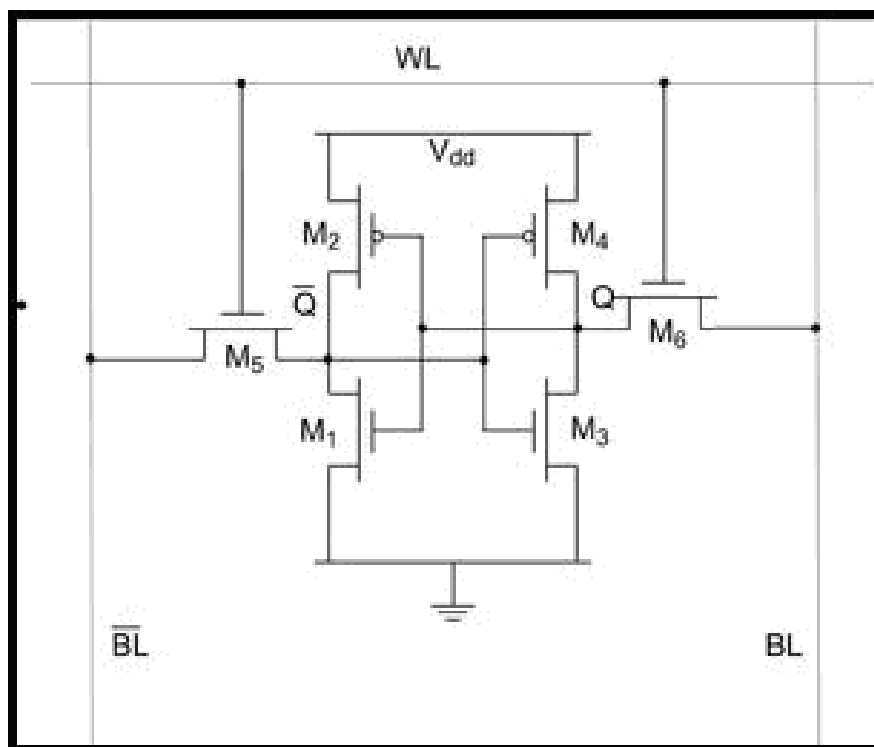


Рисунок 2.12 – Шеститранзисторна комірка статичної двійкової пам’яті
SRAM

Шеститранзисторна частина двійкової двобітної SRAM являє собою двійковий D-тригер з парафазним (двофазним) входом. В шеститранзисторній частині двійкової двобітної SRAM використовується неявна схема пріоритетного перемикавання. При подачі рівнів перемикавання на виходи інверторів відбувається конкуренція двох сигналів: рівень з інвертора схеми і рівень з шини перемикавання, тому, щоб не було конкуренції та неоднозначності, сигнали перемикавання повинні бути сильніші сигналів з інверторів: високий повинен бути вищим, а низький повинен бути нижче. Інший спосіб - перехід до більш надійних схем з явним пріоритетом перемикавання.

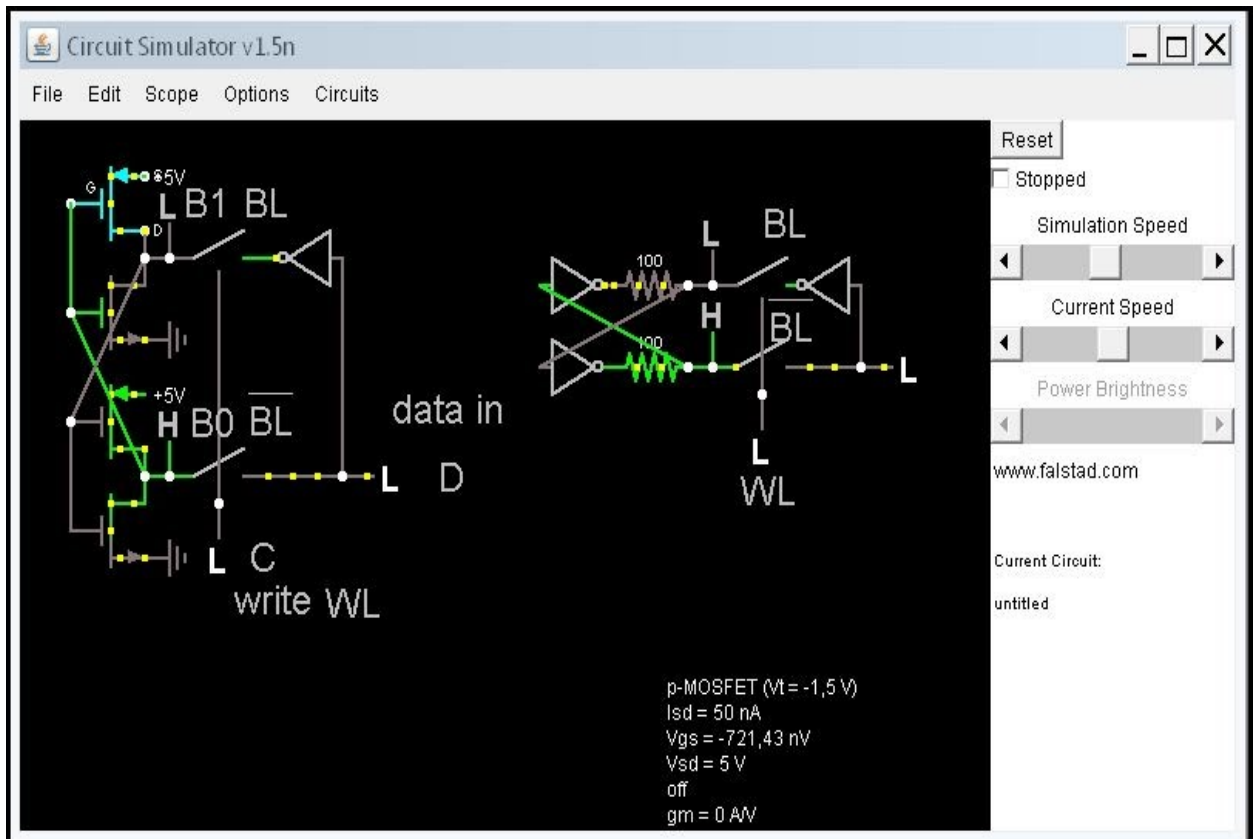


Рисунок 2.13 - Знімок моделі 6-ти транзисторної двійкової SRAM (двійкового парафазного D-тригера) в симуляторі електронних схем Circuit Simulator

3 ПРЕКТУВАННЯ SRAM ЗАСОБАМИ ACTIVE-HDL

3.1 Вибір середовища проектування

Правильний вибір середовища проектування забезпечує швидку та економічну розробку будь-якого проекту.

VHDL є мовою для опису цифрових електронних систем. Вона з'явилась в результаті виконання програми Уряду США по створенню надшвидкісних інтегральних схем (НШІС), яка була ініційована в 1980 році. В ході виконання цієї програми виникла необхідність в стандартній мові для опису структур і функцій інтегральних схем (ІС). В результаті аналізу вже існуючих мов опису апаратних засобів AHPL, CDL, CONLAN, IDL, ISPS, TEGAS, HDL, ZEUS, які використовувались провідними фірмами комп'ютерного профілю, була розроблена мова VHDL (VHSIC Hardware Description Language). В 1987 році ця мова була адаптована в якості стандарту в США інститутом інженерів-електриків та електроніків (IEEE).

VHDL спроектована для всього спектру потреб, які виникають в процесі проектування. По перше, вона дозволяє описати структуру проекту, тобто його поділ на складові частини та їх взаємозв'язок. По друге, вона дозволяє описати функцію проекту використовуючи подібні до мови програмування форми. По третє, як результат, вона дозволяє змоделювати проект перед початком виготовлення, так що проектувальники можуть швидко порівняти альтернативи та перевірити правильність функціонування без затримки та витрат на апаратне макетування.

На даний час інститут електроніків та інженерів-електротехніків (IEEE) рекомендує САПР для проектування ІС Active-HDL [15].

Пакет Active-HDL являє собою повністю інтегроване середовище розробки цифрових пристроїв на основі текстових HDL описів.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дат		

Система проектування Active-HDL на сьогодні є провідним середовищем проектування для створення проектів та їх моделювання для всіх родин програмованих логічних інтегральних схем (ПЛІС), забезпечуючи гнучкість підходу і надаючи розвинені функції підтримки найбільш складних сучасних проектів. Це середовище проектування дає розробникам незалежність у використанні інших засобів проектування з єдиного, повністю інтегрованого оточення.

Тісна інтеграція всіх додатків забезпечує абсолютний контроль над проектом від специфікацій до його фізичної реалізації на всьому шляху проектування. Active-HDL також має інтерфейси з усіма провідними програмними продуктами, надаючи розробникам свободу у використанні тих засобів проектування, які найбільш повно відповідають вимогам кожного конкретного проекту.

Система Active-HDL структурно побудована навколо свого блоку управління маршрутом проектування, який дозволяє користувачам легко отримувати доступ до засобів проектування і бібліотек, які вони використовують в проекті. Active-HDL також поставляється з усіма бібліотеками виробників ПЛІС, які вже прекомпільовані і готові до використання в проектах. Блок управління проектом дозволяє розробнику виконувати всі модифікації та операції над проектом з єдиного оточення. Така інтеграція дає кращий контроль над процесом проектування і зберігає час, усуваючи необхідність у запуску безлічі віконних інтерфейсів і процесів.

До засобів середовища проектування Active-HDL належать (рисунок 3.1):

1) Block Diagram Editor – редактор схем з можливістю генерації HDL коду після компіляції створеної схеми;

2) HDL Editor – редактор VHDL тексту з можливістю відображення заданих синтаксичних конструкції мови різними кольорами. Інтеграція

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		51

редактора з стимулятором компонент дозволяє виконувати зручне покрокове відлагодження пристрою і швидко виявляти синтаксичні помилки;

3) State Machine Editor – дозволяє проектувати керуючі автомати зручними графічними засобами з подальшою автоматичною трансляцією з графічного відображення у VHDL опис;

4) Test Bench - забезпечує входи моделювання стимулами, а також перевіряє виходи проекту.

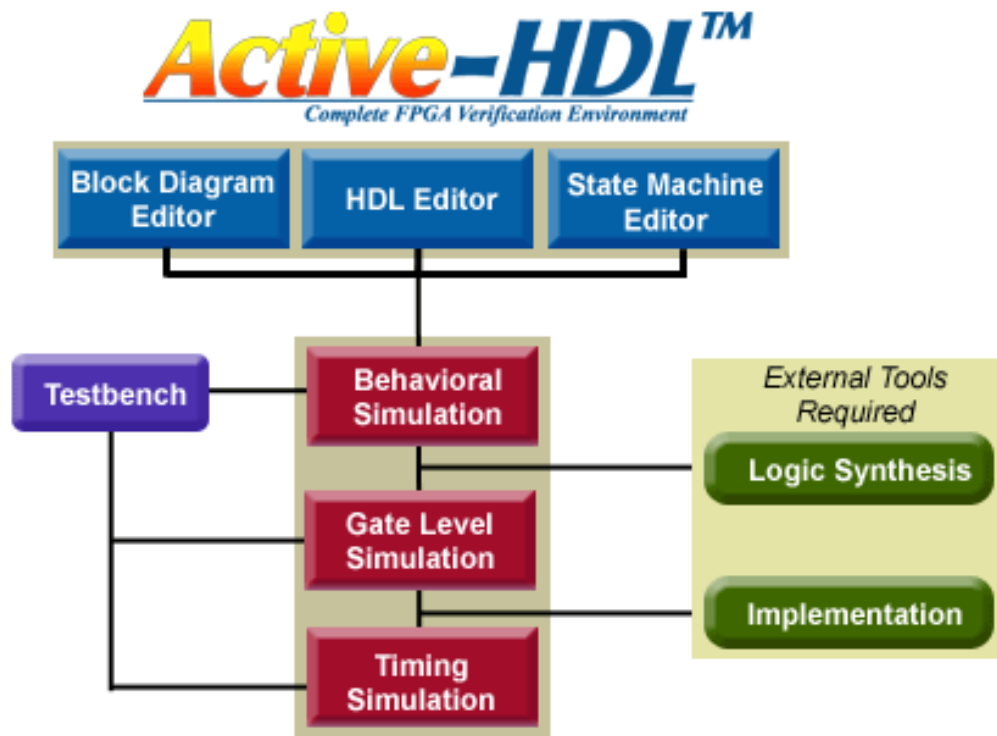


Рисунок 3.1 – Засоби середовища проектування Active-HDL

Маршрут проектування компанії Aldec не залежить від виробника ПЛІС і настраюється на будь-які комбінації засобів логічного синтезу і топологічній реалізації ПЛІС (рисунок 3.2). Розробник цифрових систем може робити проекти на ПЛІС будь-якого виробника на ринку з єдиним інтегрованим середовищем.

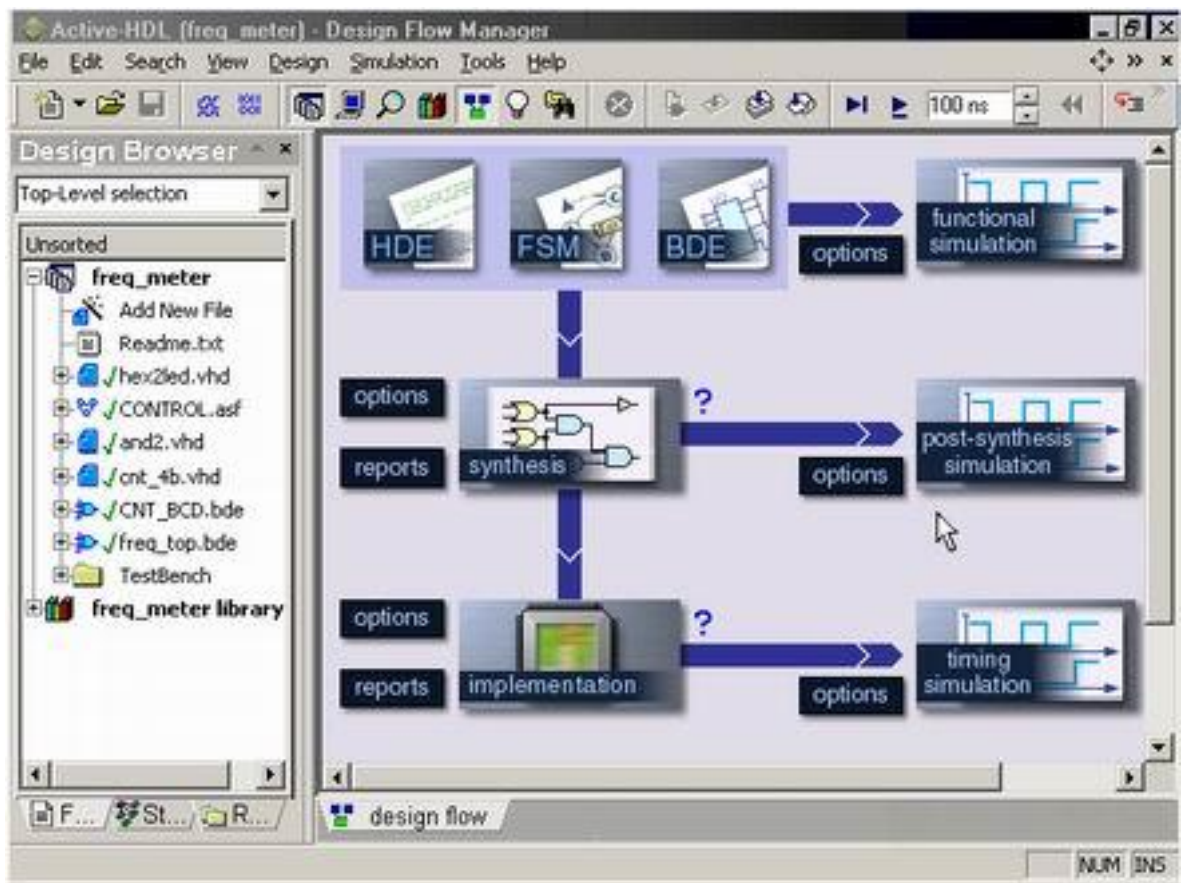


Рисунок 3.2 – Маршрут проектування в середовищі Active-HDL

Розробники можуть одночасно відкривати кілька проектів і інтегрувати їх в один супер-проект. Мультипроекторна робоча область надає середовище проектування, що дозволяє користувачам керувати всіма завантаженими проектами, перемикатися між ними, редагувати їх ресурси і конфігурувати їх незалежно один від одного. Всі модулі можуть розроблятися окремо один від одного і потім інтегруватися разом як один проект верхнього рівня [16].

Збільшення розмірів проектів змушує багатьох розробників ПЛІС використовувати методи групової розробки при проектуванні. Active-HDL надає засоби підвищення продуктивності, наприклад, такі, як засіб управління завданнями для задач, що займають великий час виконання, зокрема моделювання, логічного синтезу і топологічної реалізації. Кожна задача може бути призначена на віддалений сервер або кластер інших комп'ютерів для виконання відповідного завдання, таким чином вивільняючи

комп'ютер розробника для виконання інших завдань. Система Active-HDL також надає інтерфейс до більш, ніж 15 найпопулярніших систем управління версіями.

Система управління проектами забезпечує швидке та ефективне управління всіма ресурсами проектів. Розробники можуть використовувати її для:

- додавання, видалення, перегляду, модифікації або виконання інших операцій над файлами ресурсів проекту;
- перегляду вмісту робочої бібліотеки, бібліотеки результатів логічного синтезу і бібліотеки часових параметрів поточного проекту;
- перегляду структури модельованого проектного модуля;
- перегляду об'єктів, визначених всередині окремих областей модельованого проектного модуля.

Система управління бібліотеками надає ефективне управління всіма бібліотеками в середовищі проектування Active-HDL. Користувачі можуть виконувати такі операції над бібліотеками та їх вмістом:

- приєднання, від'єднання та видалення бібліотек;
- редагування логічних імен бібліотек;
- стиснення і звільнення бібліотек;
- перегляд вмісту бібліотек;
- перегляд вихідних файлів певних бібліотечних модулів;
- видалення певних бібліотечних модулів;
- пошук проектних модулів в бібліотеках.

Розробники цифрових систем мають доступ до найбільш повно протестованих і перевірених ІР-блоків в промисловості (рисунок 3.3). Вони можуть використовувати генератор ІР-блоків для створення необхідних моделей, таких як:

- арифметичні функції;
- послідовна логіка;

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дат		

- блоки пам'яті;
- фільтри;
- конвертори кодів;
- елементи для побудови тестів;
- промислові програми;
- програми.

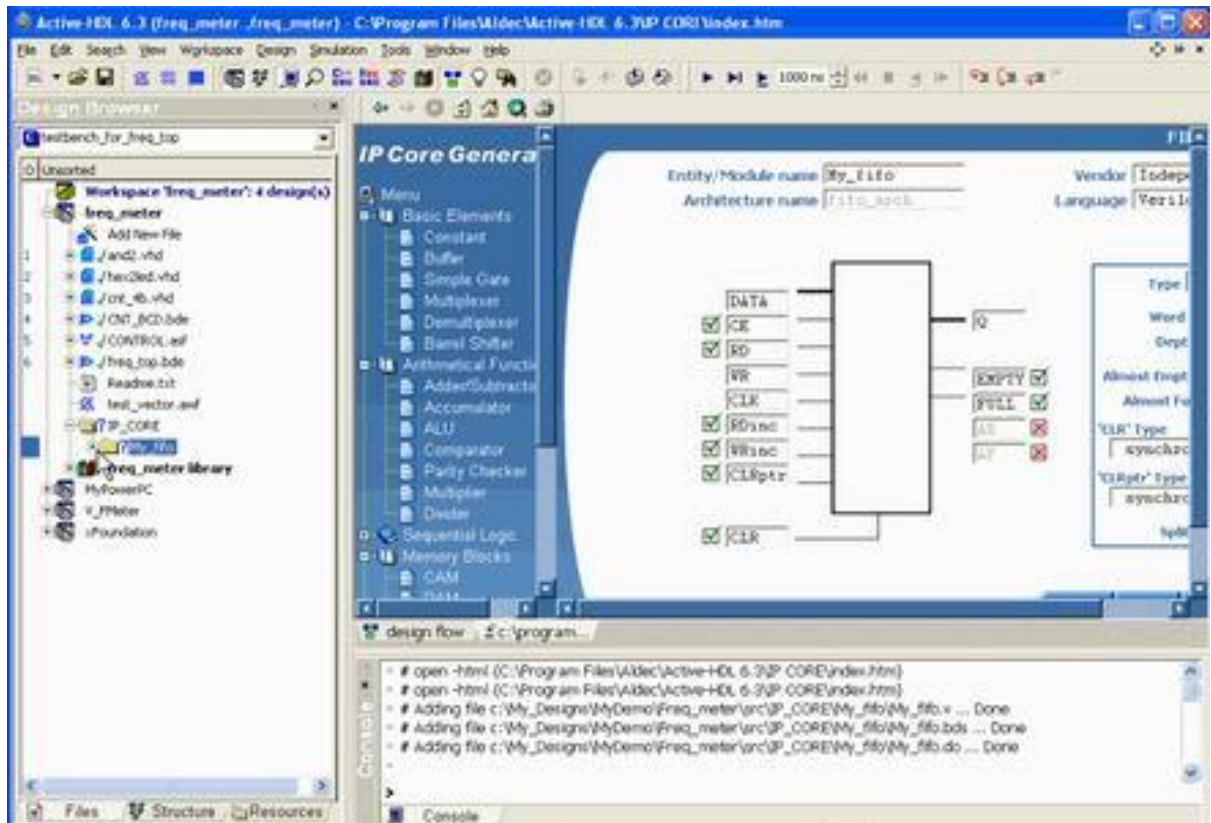


Рисунок 3.3 Генератор IP – блоків у середовищі Active-HDL

Трасування всіх сигналів дозволяє розробникам виконувати крос-налагодження (зондування) між часовими діаграмами та блок-діаграмами. Таке трасування дозволяє зробити зворотню анотацію проекту і дає прямий доступ як до текстового, так і графічного представлення проекту. Клацання мишею на будь-якому повідомленні про помилку або попередження адресує розробника безпосередньо на відповідний рядок вихідного коду.

Active-HDL дає змогу ідентифікувати ті частини проекту, які не виконувалися під час прогону тесту. Цей засіб дає можливість інженерові легко визначити, які частини тесту вимагають доопрацювання. Засоби аналізу тестового покриття компанії Aldec інтегровані в ядро моделювання і підтримують функції покриття по рядках коду, по перемиканнях та інтегральне покриття на безліч тестів (рисунків 3.4).

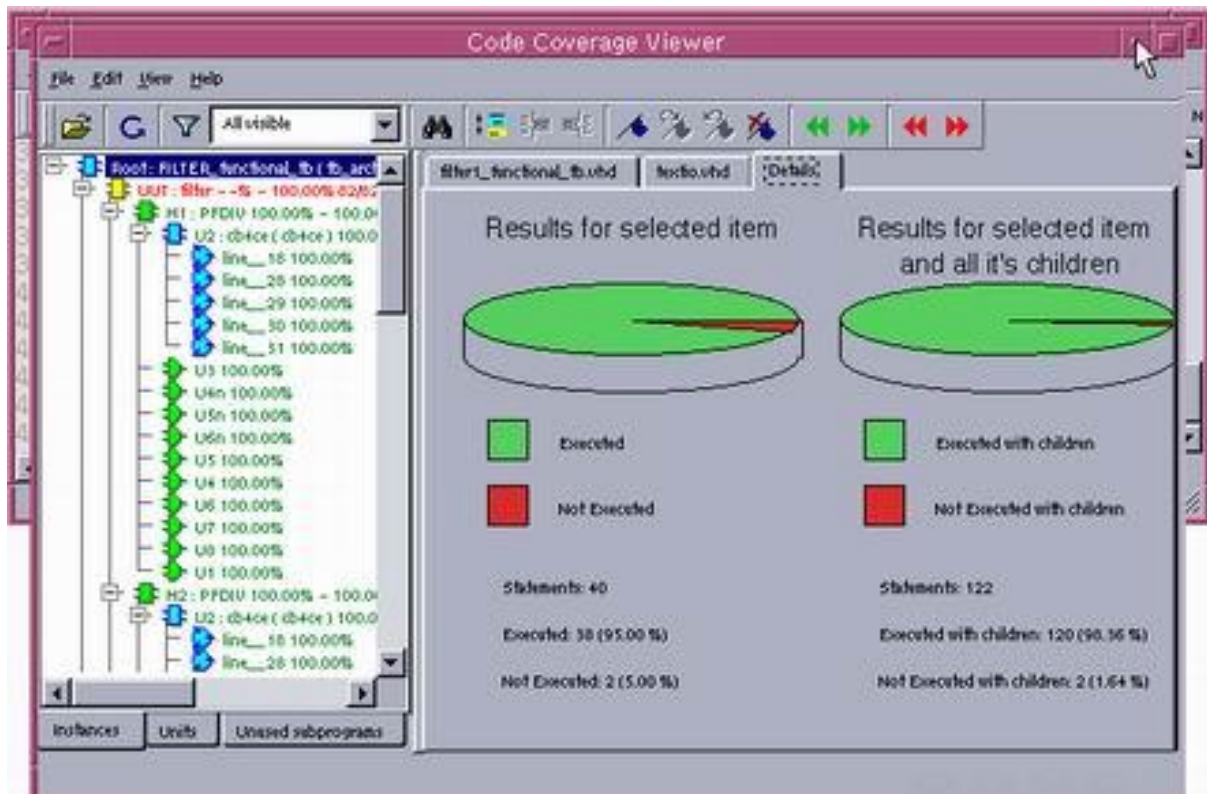


Рисунок 3.4 – Засіб аналізу створеного HDL – коду

Система Active-HDL надає інтерфейс і можливість спільного моделювання поведінкових моделей на мовах опису апаратури і блоків цифрової обробки сигналів в єдиному середовищі математичного представлення моделей високого рівня. Прямий інтерфейс із системою Simulink компанії Mathworks автоматизує процес установки для виконання спільного моделювання з Active-HDL. Перевірка роботи проекту здійснюється розробником часових діаграм (рисунк 3.5).

Проекти, зроблені в середовищі Active-HDL, можуть бути експортовані в зовнішній файл формату HTML. Такий файл HTML підтримує ту ж саму структуру та ієрархію проекту, що і Active-HDL, без необхідності роботи з самою системою моделювання. Проекти можуть розділятися серед членів робочої групи та ідеальні для документування.

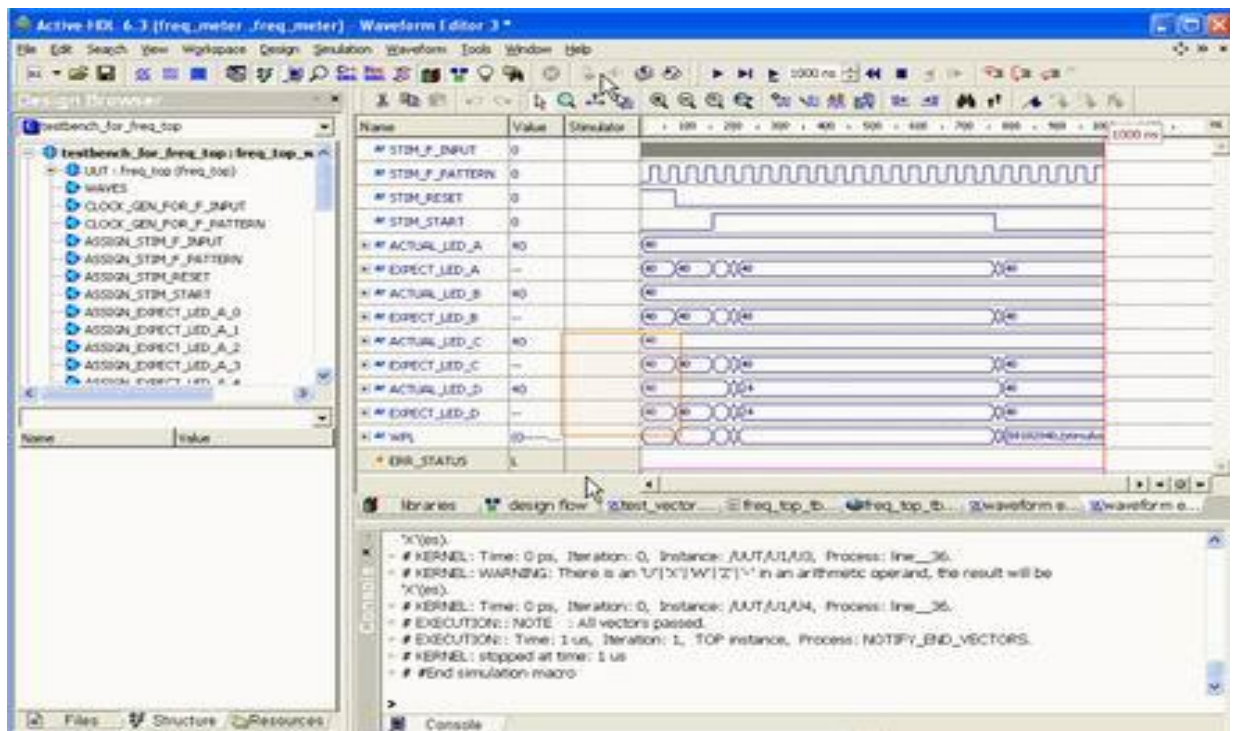


Рисунок 3.5 – Симуляція роботи проекту за допомогою часової діаграми

Функція Code2Graphics може генерувати графічне представлення проектів, зроблених на мовах VHDL або Verilog (рисунок 3.6). Ця функція дає розробникам систем ясну картину взаємозв'язків між компонентами, які використовуються в проекті. Вона аналізує вихідні файли на мовах VHDL, Verilog або в форматі EDIF і генерує один або більше файлів блок-діаграм, в залежності від кількості проектних об'єктів, модулів або елементів, знайдених в аналізованих файлах. Результуючі графічні файли (блок-діаграми або діаграми автоматів кінцевих станів) можуть бути автоматично приєднані до проекту або розміщені окремо.

Система Active-HDL може використовуватися для автоматичної генерації тестів з графічних тимчасових діаграм або діаграм автоматів кінцевих станів. Генератор тестів керує послідовністю дій інженера в діалоговому режимі і створює шаблон тесту або для окремого процесу, або на основі IEEE Waves. Один і той самий тест може використовуватися на будь-якому рівні абстракцій в процесі проектування (поведінковому, регістрових передач або тимчасовому).

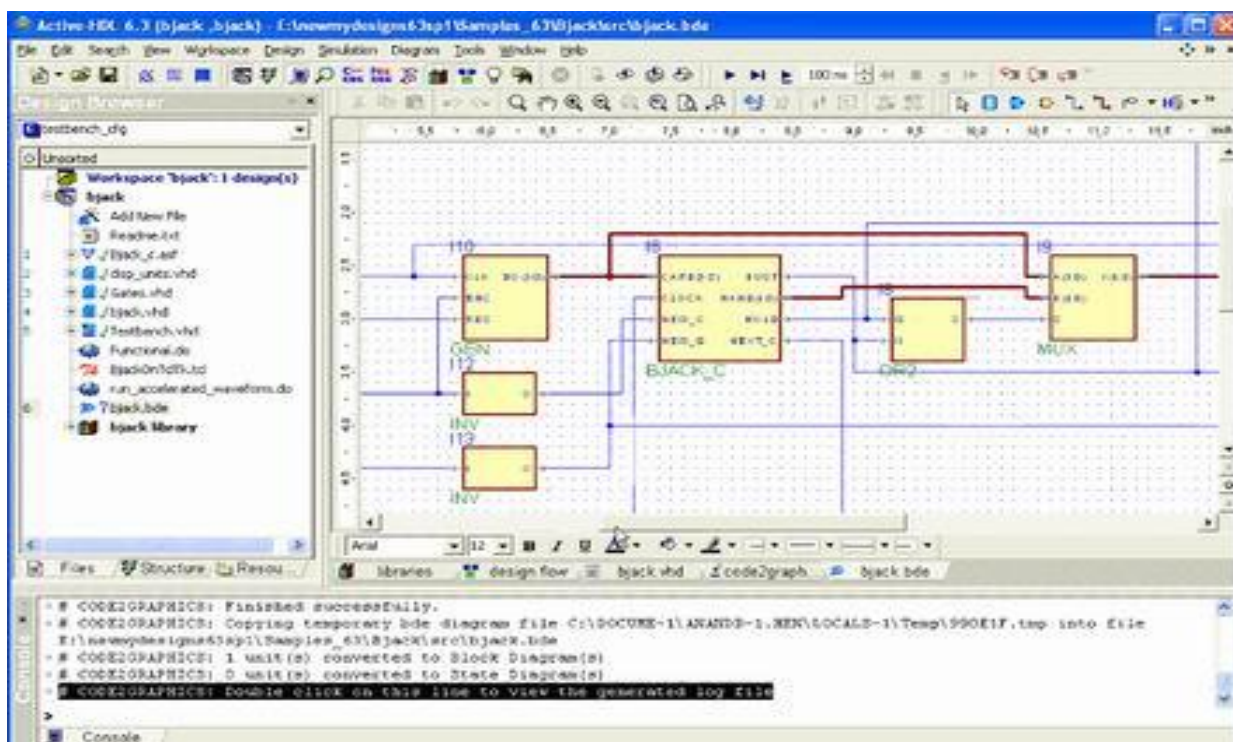


Рисунок 3.6 – Функція Code2Graphics

Проектні блоки, що займають великий час моделювання, можуть бути легко визначені за допомогою функції профілювання проекту. За рахунок ідентифікації цих блоків і оптимізації тих частин проекту, які збільшують час моделювання, загальний час моделювання проекту може бути істотно зменшений.

Системні вимоги:

- 256 Мбайт фізичної пам'яті, рекомендується 512 Мбайт;
- операційні системи Microsoft Windows NT/2000/XP;

- 198 Мбайт вільного дискового простору.

Підтримувані стандарти:

- VHDL 1076-87/93;
- Verilog 1364-95/2001 (частково);
- VITAL 1076.4-95/2000;
- SDF 1.0, 2.0 і 3.0.

Інтерфейси:

- Tcl / Tk;
- PERL;
- SWIFT;
- PLI / VPI;
- VHPI;
- CHPI;

Система Active-HDL повністю сумісна з мультиплатформовий системою моделювання Riviera компанії Aldec. Використання системи Riviera дає можливість Active-HDL робити моделювання в середовищі багатомашинних серверних систем і на інших підтримуваних платформах операційних систем, включаючи Linux і Unix.

3.2 HDL-модель статичної пам'яті з довільним доступом

Для побудови моделі необхідно визначити особливості статичної пам'яті з довільним доступом.

SRAM підтримує 18 типових часових параметрів, в ній є пари з активно низьким і активно високим Chip-Enable сигналом. В NWE-пам'ять можливий тільки контроль доступу.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дат		

Багато часу (але не весь) і контроль доступу використовується для затвердження повідомлень про порушення.

Формат даних оперативної пам'яті – це формат ASCII-файлів для того, щоб завантаження оперативної пам'яті або скидання було дуже простим. Кожен рядок файлу складається з адреси пам'яті (з урахуванням як десяткове число) і опрацьовуються відповідні дані оперативної пам'яті за цією адресою (при вигляді двійкового числа). Будь-який текст в рядку після ширини розряду двійкового числа, ігноруються. Варто зазначити, що адреса і дані повинні бути розділені одним пропуском, проте, в двійковому числі повинно бути стільки цифр, скільки вказано на загальній ширині, і що ніяких додаткових пропусків або порожніх рядків не допускається.

Приклад:

```
0 0111011010111101 – цей текст інтерпретується як коментар
1 1011101010110010
17 0010001001000100
```

Для виконання проекту необхідно застосувати наступні бібліотеки:

IEEE.std_logic_1164.all – для роботи з бінарними значеннями;
IEEE.std_logic_unsigned.all – для виконання арифметичних операцій;
IEEE.std_logic_textio.all – для роботи зі стрічковими змінними.

Конфігурація RAM в даному випадку буде наступною

- 1) кількість слів у пам'яті – 8;
- 2) кількість бітів у адресі – 3;
- 3) кількість бітів на одне слово в пам'яті – 8.

Для проектування варто ввести проміжні змінні:

- 1) clear_on_power_up – якщо її значення TRUE то RAM обновляється для запуску симуляції;
- 2) download_on_power_up - якщо її значення TRUE то у RAM загружаються вхідні значення;
- 3) trace_ram_load – відображає вхідні значення RAM на екрані;

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дат		

4) enable_nWE_only_control – дозвіл на читання/запис, який контролюється nWE.

Вхідні та вихідні порти даної моделі статичної пам'яті з довільним доступом наступні:

- nCE – малоактивний Chip-Enable приладу SRAM;
- nOE – малоактивний Output-Enable приладу SRAM;
- nWE – малоактивний Write-Enable приладу SRAM,
- A – шина адреси приладу SRAM;
- D – двонаправлена шина даних в / з пристрою SRAM;
- CE2 – високоактивний Chip-Enable приладу SRAM;
- download – FALSE або TRUE, перехід на цей сигнал загрузки даних;
- download_filename – ім'я вхідного файлу, який потрібно завантажити;
- dump – FALSE до TRUE, перехід на цей сигнал скидання;
- dump_start – письмове скидання файлу з пам'яті адресу в пам'ять слова;
- dump_end – початок скидання для його завершення;
 - dump_filename – ім'я файлу, призначеного для скидання.

Загальна структура проекту подана на рисунку 3.7.

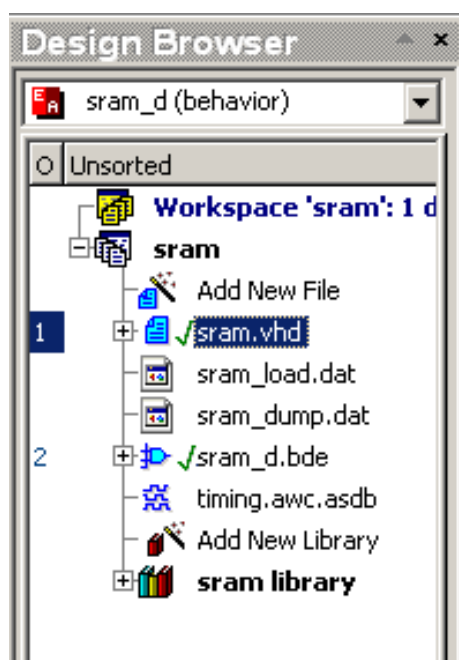



Рисунок 3.7 – Список файлів, які використовуються в проекті

На даному рисунку 3.7:

- Design Browser – вікно перегляду проекту, в якому відображається весь вміст проекту. Гаряча клавіша Alt +1 включає і вимикає вікно;
- Workspace – простий у використанні менеджер, який служить відправною точкою для запуску VHDL редактора і стимулятора;
- Sram – назва програми;
- Add new file – додати новий файл в поточну програму;
- Sram.vhdl – основна частина програми, в якій знаходиться весь HDL-код;
- Sram_load.dat – назва завантажуючого файла;
- Sram_dump.dat – назва файла результату;
- Sram_d.bde – блок-схема моделі пам'яті, такі файли позначаються спеціальним значком із зображенням символу ;
- Timing.awc.asdb – часова діаграма зовнішніх впливів;
- Add New Library – додати нову бібліотеку у програму;
- Sram library – список бібліотек, які вже використовуються у даній програмі.

Архітектура розробленої HDL-моделі складається з наступних модулів:

- 1) Check_For_Valid_Data – функція перевірки даних;
- 2) Check_For_Tristate – перевірка наявності Tristate;
- 3) Memory – процес опису характеристик пам'яті;
- 4) Power_up – збільшення потужності;
- 5) Load – загрузка даних з файлу;
- 6) Do_damp – запис даних у вихідний файл;
- 7) PROCESS (A) - перевіряє чи зміни адреси дозволено;
- 8) PROCESS (nOE, nWE, nCE, CE2) – перевіряє чи керуючі сигнали на

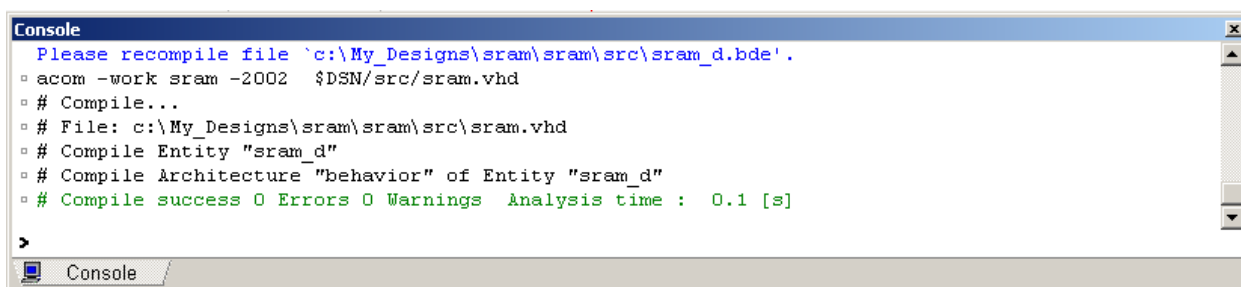
RAM дійсні весь час.

Схема взаємозв'язку модулів HDL – коду подана в додатку Б.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		62

3.3 Результати роботи спроектованої SRAM

Для перевірки правильності розробленого HDL-коду необхідно здійснити компіляцію. Компіляція – це вікно для інтерактивного введення-виведення текстової інформації. В момент компіляції система повідомляє, який був завантажений проект, як він називається і де він знаходиться. Також під час компіляції перевіряється код на вміст помилок в тексті, і якщо вона пройшла успішно, тобто без помилок, програма запускається на виконання, в противному випадку програма компіляції вказує на наявність помилки і на місце її знаходження. В даній програмі під час компіляції було показано, що програма зроблена правильно – 0 помилок, 0 попереджень (рисунок 3.8).



```
Console
Please recompile file 'c:\My_Designs\sram\sram\src\sram_d.bde'.
acom -work sram -2002 $DSN/src/sram.vhd
# Compile...
# File: c:\My_Designs\sram\sram\src\sram.vhd
# Compile Entity "sram_d"
# Compile Architecture "behavior" of Entity "sram_d"
# Compile success 0 Errors 0 Warnings Analysis time : 0.1 [s]
```

Рисунок 3.8 – Результати виконання компіляції в даному проекті

Симуляція роботи моделі статичної пам'яті здійснена засобом Wavefarm Editor. В якості вхідних даних взято файл sram_load.dat.

Результати роботи проекту зображено на рисунку 3.9.

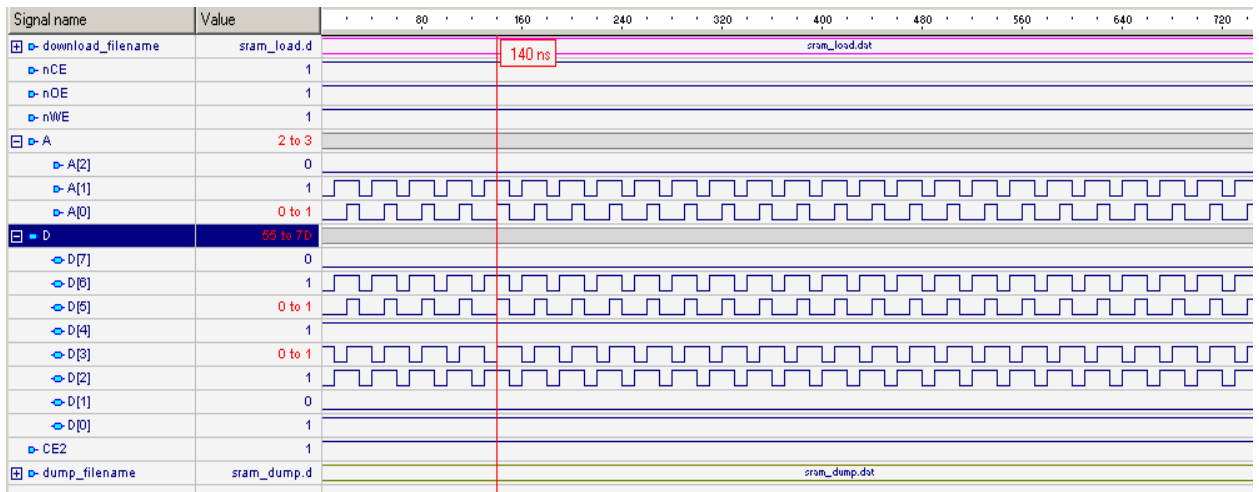


Рисунок 3.9 – Часова діаграма HDL-моделі статичної пам'яті з довільним доступом

З аналізу часової діаграми випливає, що модель статичної пам'яті з довільним доступом спроектована вірно.

Блок діаграма розробленої HDL-моделі подана на додатку В.

В результаті проведеної верифікації проекту можна зробити висновок, що проектування здійснено вірно і HDL-модель може застосовуватись до програмування ПЛІС чи ПЛІМ.

4 ОХОРОНА ПРАЦІ

Поняття “охорона праці” визначено статтею 1 Закону України “Про охорону праці”. Охорона праці – це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів і засобів, спрямованих на збереження здоров’я і працездатності людини в процесі праці.

Основною ціллю охорони праці є створення на кожному робочому місці безпечних умов праці, експлуатації обладнання, зменшення або повна нейтралізація дії шкідливих і небезпечних виробничих факторів на організм людини і зниження виробничого травматизму та професійних захворювань.

4.1 Аналіз санітарно-гігієнічних умов праці

У даному розділі розглядаються питання по охороні праці при розробці веб-сайту «Захист інформації в глобальній мережі Інтернет» засобами системи керування вмістом Joomla.

Даний програмний комплекс розроблявся на базі а.4108. У відділі постійно працює до дванадцяти чоловік. Види виконуючих робіт – вивчення матеріалів у відповідності з розвитком техніки, розробка, налагодження і запуск програм на комп’ютерах відділу. Для цих цілей у відділі використовується 12 персональних ЕОМ.

План а.4108 Тернопільського національного економічного університету подано на рисунку 3.1.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						65
Змн.	Арк.	№ докум.	Підпис	Дат		

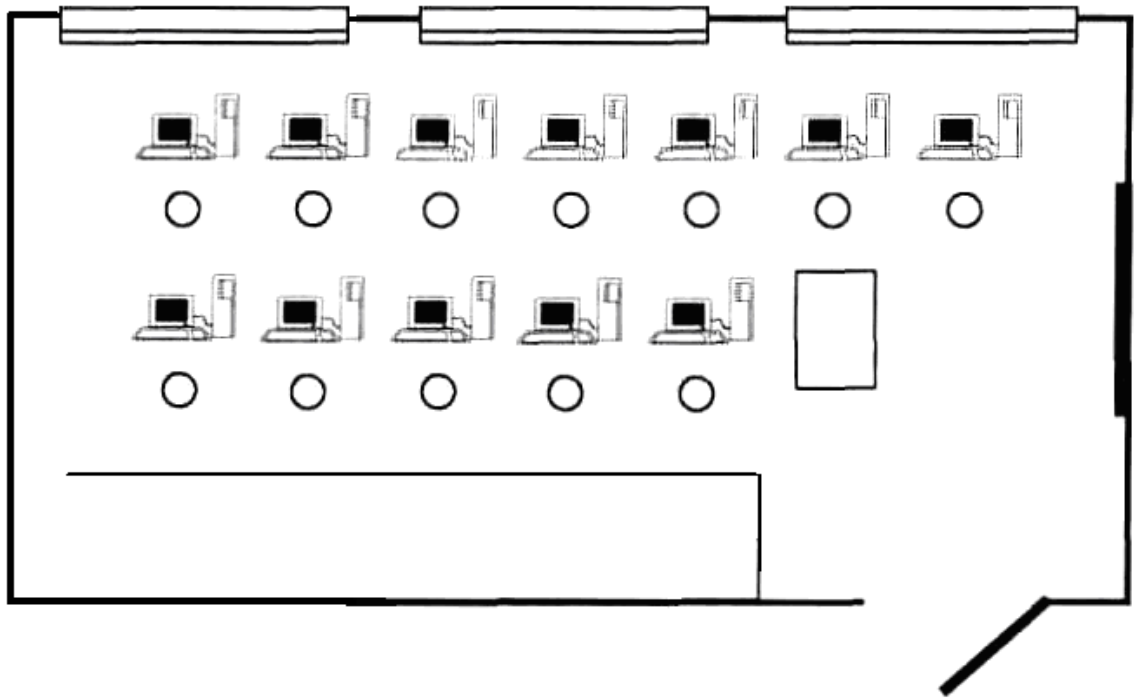


Рисунок 3.1 – План інженерно-технічного відділу

Розміри відділу описані в таблиці 3.1.

Таблиця 3.1 – Розміри відділу

Позначен	Визначення	Значення
I	Довжина	9 м
d	Ширина	7 м
h	Висота	4 м
S_0	Площа	63 м^2
V_0	Об'єм	252 м^3

Згідно СН-245-71, на одного працюючого об'єм приміщення повинен складати не менше $19,5 \text{ м}^2$, площа – не менше 6 м^2 . Число працюючих у приміщенні $N_p=9$. Таким чином, на одне робоче місце приходиться площа $S=63/9=7 \text{ (м}^2\text{)}$ і об'єм $V = 252/9=28 \text{ (м}^3\text{)}$. Ці значення відповідають вимогам.

Крім того, повинні дотримуватися норми приміщення, подані у таблиці 3.2.

Таблиця 3.2 – Норми приміщення

Параметр	Значенн
ширина основних проходів	≥ 1200
ширина допоміжних проходів	≥ 700 мм
відстань між двома столами, якщо між ними є стілець	≥ 1300

У розглянутому приміщенні а.4108: відстань між двома столами становить 1500 мм, відстань між двома столами в ряді – 1500 мм, а між рядами – 2000 мм.

Отже, норми виконуються.

У технічних умовах роботи ЕОМ вказуються робочі діапазони параметрів мікроклімату:

- температура повітря 5 – 45 °С,
- відносна вологість повітря 40 – 90 %.

Однак, вимоги точного регулювання параметрів повітряного середовища приміщення значно звужують ці діапазони.

З метою забезпечення комфортних умов для персоналу, а також максимальної безвідмовності функціонування техніки, встановлюють вимоги до повітряного середовища приміщень. Так, у відділі повинні бути:

- температура повітря 18 – 22 °С,
- відносна вологість повітря 50 – 60 %,
- атмосферний тиск 1013 – 1013,5 гПа.

При зниженні тиску погіршується відвід тепла від елементів ЕОМ, знижуються ізоляційні властивості повітря.

Як було показано вище, показники об'єму і площі приміщення на одного працюючого відповідають нормативним значенням.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		67

Роботи, що проводяться в а.4108 відносяться до легких фізичних робіт групи 1а, відповідно до ГОСТ 12.1.005-88, оскільки вони проходять сидячи і не вимагають фізичного навантаження, здійснюються при нормальних метеорологічних умовах і не викликають забруднення одягу і рук. Витрати енергії не перевищують 172 Дж/с (155 ккал/год). У таблиці 3.3 і таблиці 3.4 наведені норми температури, відносної вологості і швидкості руху повітря на робочих місцях відповідно до ГОСТ 12.1.005-88, що встановлює норми виробничого мікроклімату. Дані приведені для приміщень з незначним надлишком явного тепла (до 20 ккал/год м³) для виконання легких робіт.

Таблиця 3.3 – Норми температури, відносної вологості і швидкості руху повітря на постійних робочих місцях

Період року	Норми	Температура повітря t, °С	Відносна вологість, %	Швидкість руху повітря, м/с
Холодний	оптим.	22-24	40-60	менше, ніж 0,1
	доп.	21-25	менше 75	менше, ніж 0,1
Теплий	оптим.	23-25	40-60	0,1
	доп.	22-28	менше, ніж 55	0,1-0,2

Таблиця 3.4 – Відносна вологість повітря в теплий період року

Температура повітря, °С	2	2	2	2	2	≤
	8	7	6	5	4	23
Відносна вологість, %	≥	6	6	7	7	7
	55	0	5	0	5	5

Основними джерелами тепла у відділі є:

- сонячна радіація,
- система опалення,
- люди, що працюють у приміщенні,
- устаткування.

У таблиці 3.5 приведені дані, виміряні в інженерно-технічному відділі у лютому місяці.

Таблиця 3.5 – Результати виміру параметрів мікроклімату у відділі

Параметр	Значення
Температура повітря t, °С	17 – 20
Відносна вологість, %	50 – 60
Швидкість руху повітря, м/с	0,2

Як видно з таблиці 3.5, у розглянутому приміщенні відділу значення параметрів мікроклімату відповідають нормативним. Постійність цих параметрів підтримується загальною системою утеплення і кондиціонування повітря. При цьому використовується кондиціонер SAMSUNG AQT-24A5RE, а при необхідності здійснюється провітрювання приміщення. У таблиці 3.6 зображено параметри кондиціонера SAMSUNG AQT-24A5RE. Він забезпечує встановлені норми мікроклімату у відділі.

Таблиця 3.6 – Параметри кондиціонера SAMSUNG AQT-24A5RE

Параметр	Значення
Потужність охолодження	6.8 кВт
Продуктивність охолодження	24 000 БТЕ/год
Потужність обігріву	6,9 кВт
Продуктивність обігріву	24 000 БТЕ/год
Видалення вологи з повітря	3 л/год
Циркуляція повітря	14 м ³ /хв

Джерелами пилу в а.4108 є: книги, документація, роздруківки, а також одяг, взуття працівників і зовнішнє повітря.

Встановлений у відділі кондиціонер SAMSUNG AQT-24A5RE забезпечує встановлені норми чистоти поступаючого зі сторони приміщення повітря, що надходить ззовні. У відділі періодично проводиться вологе

прибирання. Зазначені умови забезпечують підтримку в нормі параметрів чистоти повітряного середовища.

У відділі використовується природне і штучне освітлення. Природне освітлення здійснюється з допомогою трьох вікон загальною площею $S=42 \text{ м}^2$, що забезпечує коефіцієнт природної освітленості $E=1,5\%$. Це відповідає СНіП І-4-79.

Штучне освітлення у відділі здійснюється системою загального рівномірного освітлення, що реалізована на основі люмінесцентних ламп типу ЛДЦ-40-1, що мають наступні параметри:

- висока світловіддача;
- тривалий термін служби;
- мала яскравість освітлювальної поверхні;
- близькість спеціального складу до природного освітлення.

Робота за монітором ПЕОМ по розряду зорових робіт відноситься до III типу (роботи високої точності з розміром об'єкта 0,2-0,4 мм). При загальному освітленні, освітленість робочого місця повинна складати від 200 до 400 лк.

При штучному освітленні нормуються наступні параметри:

- E (лк) – найменша припустима освітленість;
- M – показник дискомфорту;
- $K_{\text{п}}$ (%) – коефіцієнт пульсації освітлення.

Номинальний світловий потік лампи білого свічення ЛДЦ-40-1: $\Phi_{\text{л}}=3120$ лм.

У відділі застосовуються світильники, у яких встановлені дві лампи. Висота підвіски світильника визначається за формулою:

$$h = H - h_{\text{С}} - h_{\text{Р}} - h_{\text{П}}, \quad (3.1)$$

де H – висота приміщення (м),

$h_{\text{С}}$ – висота світильника (м),

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						70
Змн.	Арк.	№ докум.	Підпис	Дат		

h_{Π} – відстань від стелі до підвіски (м),

$h_{\text{р}}$ – висота робочої поверхні (м).

Для розглянутого відділу:

$$H = 4 \text{ м,}$$

$$h_{\text{с}} = 0,15 \text{ м,}$$

$$h_{\Pi} = 0 \text{ м, (підвісу немає)}$$

$$h_{\text{р}} = 0,8 \text{ м.}$$

$$\text{Звідси } h = 4 - 0,15 - 0,8 = 3,05(\text{м}).$$

Світильники розташовані в 3 ряди. Висота підвіски світильників складає 3,05 м відносно підлоги, відстань між рядами 1 м, відстань від ряду до стіни 1,5 м. Приміщення має наступні габарити:

– довжина $A = 9$ м,

– ширина $B = 7$ м.

Визначимо освітленість у робочій точці. Для розрахунку загальної рівномірної освітленості при горизонтальній робочій поверхні використовуємо метод коефіцієнта використання світлового потоку.

Розрахункова формула для світлового потоку світильника має такий вигляд:

$$\Phi_{\text{л}} = \frac{E \cdot K_{\text{з}} \cdot S \cdot Z}{N \cdot n}, \quad (3.2)$$

де N – кількість світильників у відділі ($N = 6 \cdot 3 = 18$);

n – коефіцієнт використання світлового потоку;

$\Phi_{\text{л}}$ – світловий потік ламп;

$K_{\text{з}}$ – коефіцієнт запасу ($K_{\text{з}} = 1,5$);

Z – коефіцієнт нерівномірності;

S – площа приміщення;

E – освітленість, створювана усіма світильниками.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						71
Змн.	Арк.	№ докум.	Підпис	Дат		

Звідси одержуємо формулу для розрахунку освітленості на робочому місці:

$$E = \frac{\Phi_{л} \cdot N \cdot n}{K_{з} \cdot S \cdot Z} \quad (3.3)$$

Коефіцієнт використання світлового потоку залежить від:

- КПД кривої розподілу сили світла світильника;
- коефіцієнта відбивання стелі $R_{п}$ і стін $R_{с}$;
- висоти підвісу світильників $h_{п}$;
- показника приміщення i :

$$i = \frac{A \cdot B}{h \cdot (A + B)} \quad (3.4)$$

Тобто $i = (9 \cdot 7) / (3,05 \cdot (9 + 7)) = 1,29$.

Стеля і стіни пофарбовані в білий колір.

Приймаємо $R_{п} = 50\%$, $R_{с} = 30\%$.

Звідси $n = 31\%$,

$$E = \frac{(3120 \cdot 2) \cdot 18 \cdot 0,31}{63 \cdot 1,1 \cdot 1,5} = 335_{лк}$$

Оскільки по розряду зорової роботи робота за дисплеєм ПЕОМ відноситься до III типу (високої точності, розмір об'єкта 0.2-14 мм), то при загальному висвітленні освітленість робочого місця повинна складати від 200 до 400 лк, рекомендована освітленість при роботі з дисплеєм ПЕОМ складає 200 лк, а при сполученні роботи з документами — 400 лк. Фактична освітленість на робочому місці складає 335 лк. Таким чином для роботи з дисплеєм цілком достатньо існуючих джерел світла, однак робота з документами повинна вестися при природному освітленні або за допомогою

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		72

додаткових місцевих джерел освітлення.

У відділі основними джерелами шумів є: вентилятори системи охолодження ПЕОМ, кондиціонер і друкуючий пристрій. Згідно ГОСТ 12.1.003-83, нормованою шумовою характеристикою робочих місць при постійному шумі являються рівні звукових тисків у децибелах в октавних смугах. Сукупність таких рівнів називається граничним спектром (ГС), номер якого дорівнює рівню звукового тиску в октавній смузі із середньо-геометричною частотою 1000 Гц. В таблиці 3.7 приведені значення звукового тиску у відділі при роботі принтера.

Таблиця 3.7 – Рівні звукового тиску в дБ на робочих місцях

Найменування параметрів	Нормовані значення	Фактичні значення
Розміри символів по висоті h, мм	≥ 3	4
Ширина лінії, мм	≥ 0.4	0.4
Яскравість зображення, лм	100	100
Потужність дози рентгенівського випромінювання на відстані 5 см, мкР/с	≤ 0.03	0.01
Щільність потоку ультрафіолетового випромінювання,	≤ 10	8
Шум, дБА	≤ 40	10

Електромагнітні випромінювання низької частоти (від 12 до 150 Гц) роблять найбільш шкідливий вплив на організм людини. Тривалий вплив низькочастотних полів сприяє порушенню репродуктивної функції і виникненню раку.

Для зниження рівня перемінного електромагнітного поля в сучасних моніторах, що відповідають специфікаціям Low Radiation (LR), MPRII і TCQ92, застосовуються котушки компенсації, встановлені на електронно-променевій трубці (ЕПТ), а також спеціальні матеріали в її конструкції. Застосовувані при роботі у відділі монітори Celeron 1700, 2001 року виготовлення, задовольняють встановлені норми.

В найбільшій мірі негативний вплив на зір при роботі з ПЕОМ зв'язано з нерівномірно спроектованим освітленням, прямими і відбитими від екранів відблесками, несприятливим розподілом яскравості в полі зору, пульсацією екрана, неправильним розміщенням робочого місця відносно світлових променів.

Оптимальною для робочих приміщень, призначених для роботи з відеотерміналами, вважається освітленість 200 – 400 лк. Стрибок яскравості при зміні полів зору повинен бути мінімальним, тобто інтенсивність освітлення поверхні, де знаходяться рукописи і документи, не повинні перевищувати яскравості екрана дисплея. Співвідношення яскравості екрана і безпосередньо найближчого оточення не повинне перевищувати три до одного. Фактично дані вимоги на робочому місці виконуються згідно вимог ГОСТ 27016-86.

При тривалій роботі з друкувальним пристроєм вимоги по охороні праці в області тривалих шумових впливів на оператора виконуються і відповідають встановленим нормам.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						74
Змн.	Арк.	№ докум.	Підпис	Дат		

4.2 Пожежна безпека

Розглянута а.4108 відноситься до категорії В, класу П-Па ПУЕ 76/87 по пожежній небезпеці. У відділі є горючі речовини:

- волокнисті (папір);
- тверді (дерево).

Пожежа у відділі представляє особливу небезпеку, оскільки пов'язана зі значними матеріальними втратами. Як відомо, пожежа може виникнути при взаємодії горючих речовин, окислювача і джерела запалювання. Горючими речовинами являються будівельні матеріали для акустичної обробки приміщення, перегородки, двері, підлога, папір для принтеру, корпуси ПЕОМ і принтерів, ізоляція кабелів. Особливістю сучасних ПЕОМ являється дуже висока щільність розміщення елементів електронних схем. При проходженні електричного струму по провідниках і деталях виділяється тепло, що в умовах їх високої щільності може привести до перегріву. Надійна робота окремих елементів і електричних схем в цілому забезпечується тільки у визначених інтервалах температури, вологості і при заданих електричних параметрах. При відхиленні реальних умов експлуатації від розрахункових може виникнути пожежонебезпечна ситуація.

Кабельні лінії зв'язку являються найбільш пожежонебезпечним місцем. Для зниження загоряння і здатності розповсюдження вогню кабелі покривають вогнетривким покриттям.

Для гасіння пожежі на початковій стадії її виникнення у відділі встановлені 3 вуглекислотних вогнегасники ОУ-2.

Для передбачення пожежі в відділі прийняті такі міри:

- передбачений вільний доступ до мережевих рубильників і вимикачів;
- на випадок короткого замикання передбачені запобіжники і автоматичне відключення мережі;

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						75
Змн.	Арк.	№ докум.	Підпис	Дат		

- в наявності є вогнегасники ОУ-2 для гасіння електрообладнання і ОХП-10 для гасіння об'єктів, що не знаходяться під напругою;
- вхідні двері відділу відкриваються на зовні;
- ширина дверей не менше 0,8 м, а висота проходу більше 1 м;
- у відділі є план евакуації людей;
- у спільному коридорі, поруч з відділом, знаходиться пожежний кран;
- ширина загального коридору, ширина дверей, висота дверей відповідають нормативним значенням (таблиця 3.9).

Таблиця 3.9 – Характеристики евакуаційних виходів

	Нормативні	Існуючі значення,
Ширина	> 2,0	2,5
Ширина дверей	> 0,8	1,2
Висота дверей	> 2,0	2,5

ВИСНОВОК

В даному дипломному проекті ми вирішили наступні задачі:

- 1) приведено аналіз діяльності бази практики, а також апаратного та програмного забезпечення, яке застосовується в лабораторії кафедри КІ;
- 2) була описана оперативна пам'ять та її призначення, і розглянуті різновиди оперативної пам'яті, статична та динамічна пам'ять;
- 3) були визначені побудова та принципи функціонування статичної пам'яті з довільним доступом, типи статичної пам'яті і основні характеристики SRAM;
- 4) Проведений аналіз санітарно-гігієнічних умов праці і визначені умови пожежної безпеки.

Проведені під час проходження переддипломної практики дослідження є основою для подальшого дипломного проектування.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						77
Змн.	Арк.	№ докум.	Підпис	Дат		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Методичні вказівки щодо проходження переддипломної практики студентами спеціальності 6.091501 "Комп'ютерні системи та мережі" / Березький О.М., Дубчак Л.О., Васильків Н.М.- Тернопіль: ТНЕУ, 2011р.
2. Жарков С.Д. Оперативная память./М.: Диалектика, 2000 – 125с.
3. Рорбоу Л.У Модернизация вашего ПК./М.: Диалектика, 1997 -384с.
4. Фигурнов В.Э IBM PC для пользователя Издание 7-ое./М.: Инфра-М, 1998 – 480с.
5. Симонович С.В. Вы купили компьютер./М.: АСТ-Пресс, 2002 - 432с.
6. Мюллер С.П. Модернизация и ремонт ПК./М.: Вильямс, 2000 – 1340с.
7. Гук М.Ю. Аппаратные средства IBM PC./М.: Питер, 2003 -1072с.
8. Кузнецов Е. Ю., Осман В. М. Персональные компьютеры и программируемые микрокалькуляторы: Учеб. пособие для ВТУЗов./М.: Высш. шк, 1991 – 443с.
9. Борзенко А.Е. IBM PC: устройство, ремонт, модернизация./М.: КомпьютерПресс, 1997 – 342с.
10. Кравец В.А. Зарубежные ЭВМ. Оборудование и программное обеспечение./Х.: Основа, 1991 – 216с.
11. Степаненко О.С. Информационно-вычислительные системы./К.: УСКП Кобза, 1994 – 274с.
12. Ахметов А. Н., Борзенко А. В. Современный персональный компьютер./М.:Компьютер Пресс, 2003 - 471с.
13. Астафьева Н. Е., Гаврилова С. А., Ракитина Е. А., Вязовова О. В. Информатика в схемах./М.: Бином, 2010 – 48с.
14. Симонович С.В. Общая информатика./П.: Питер, 2007 - 428с

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		78

15. Суворова Е.А., Шейнин Ю.Е. Проектирование цифровых систем на VHDL./П.: БХВ-Петербург, 2003 – 576с.
16. Армстронг Дж.Р. Моделирование цифровых систем на языке VHDL: Пер с англ./М.: Мир, 1992 - 175 с.
17. Максимов И.В., Партыка Т.Л., Попов И.И. Технические средства информатизации ./М.: ИНФРА, 2008 – 576с.
18. Донцов Д., Железо ./П.: Питер, 2005 -144с.
19. Акулов О. А., Медведьев Н. В. Информатика: базовый курс ./М.: Омега-Л, 2006 - 552с.
20. Дорот В. А., Новиков Ф. Н. Толковый словарь современной компьютерной лексики. 2-е изд ./СПб.: ВHV, 2001 – 607с.
21. Лесничая И.Г. Информатика и информационные технологии. Учебное пособие ./М.: Эксмо, 2007 – 592с.
22. Гордеев А.В. Операционные системы ./М.: Планета, 2004 – 416с.

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		79

Додаток А

Текст програми роботи контролера

```
USE std.textio.all;  
LIBRARY IEEE;  
USE IEEE.std_logic_1164.all;  
USE IEEE.std_logic_unsigned.all;  
USE IEEE.std_logic_textio.all;
```

```
ENTITY sram IS
```

```
  GENERIC (  
    clear_on_power_up: boolean := FALSE;  
    download_on_power_up: boolean := TRUE;  
    trace_ram_load: boolean := TRUE;  
    enable_nWE_only_control: boolean := TRUE;  
  
    size: INTEGER := 8;  
    adr_width: INTEGER := 3;  
    width: INTEGER := 8;  
  
    tAA_max: TIME := 20 NS;  
    tOHA_min: TIME := 3 NS;  
    tACE_max: TIME := 20 NS;  
    tDOE_max: TIME := 8 NS;  
    tLZOE_min: TIME := 0 NS;  
    tHZOE_max: TIME := 8 NS;  
    tLZCE_min: TIME := 3 NS;  
    tHZCE_max: TIME := 10 NS;  
  
    tWC_min: TIME := 20 NS;  
    tSCE_min: TIME := 18 NS;
```

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						80
Змн.	Арк.	№ докум.	Підпис	Дат		


```

tAW_min:  TIME := 15 NS;
tHA_min:  TIME := 0 NS;
tSA_min:  TIME := 0 NS;
tPWE_min: TIME := 13 NS;
tSD_min:  TIME := 10 NS;
tHD_min:  TIME := 0 NS;
tHZWE_max: TIME := 10 NS;
tLZWE_min: TIME := 0 NS
);

PORT (

nCE: IN std_logic := '1';
nOE: IN std_logic := '1'
nWE: IN std_logic := '1';

A: IN std_logic_vector(adr_width-1 downto 0);
D: INOUT std_logic_vector(width-1 downto 0);
CE2: IN std_logic := '1';

download: IN boolean := FALSE;

download_filename: IN string := "sram_load.dat

dump: IN boolean := FALSE;

dump_start: IN natural := 0

dump_end: IN natural := size-1;

dump_filename: IN string := "sram_dump.dat
);
END sram;

```

ARCHITECTURE behavior OF sram IS

```

FUNCTION Check_For_Valid_Data (a: std_logic_vector) RETURN BOOLEAN
IS
VARIABLE result: BOOLEAN;
BEGIN
result := TRUE;
FOR i IN a'RANGE LOOP
result := (a(i) = '0') OR (a(i) = '1');

```

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		81

```

    IF NOT result THEN EXIT;
    END IF;
END LOOP;
RETURN result;
END Check_For_Valid_Data;

```

```

FUNCTION Check_For_Tristate (a: std_logic_vector) RETURN BOOLEAN IS
    VARIABLE result: BOOLEAN;
BEGIN
    result := TRUE;
    FOR i IN a'RANGE LOOP
        result := (a(i) = 'Z');
        IF NOT result THEN EXIT;
        END IF;
    END LOOP;
    RETURN result;
END Check_For_Tristate;

```

```

SIGNAL tristate_vec: std_logic_vector(D'RANGE);

```

```

SIGNAL undef_vec: std_logic_vector(D'RANGE

```

```

SIGNAL undef_adr_vec: std_logic_vector(A'RANGE);

```

```

SIGNAL read_active: BOOLEAN := FALSE

```

```

SIGNAL read_valid: BOOLEAN := FALSE

```

```

SIGNAL read_data: std_logic_vector(D'RANGE);

```

```

SIGNAL do_write: std_logic := '0'

```

```

SIGNAL adr_setup: std_logic_vector(A'RANGE);

```

```

SIGNAL adr_hold: std_logic_vector(A'RANGE);

```

```

SIGNAL valid_adr: std_logic_vector(A'RANGE);

```

```

BEGIN

```

```

    PROCESS BEGIN

```

```

        FOR i IN D'RANGE LOOP

```

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		82

```

    tristate_vec(i) <= 'Z';
    undef_vec(i) <= 'X';
END LOOP;
FOR i IN A'RANGE LOOP
    undef_adr_vec(i) <= 'X';
END LOOP;
WAIT;
END PROCESS;

```

memory: PROCESS

```

    CONSTANT low_address: natural := 0;
    CONSTANT high_address: natural := size - 1;

```

```

    TYPE memory_array IS
        ARRAY (natural RANGE low_address TO high_address) OF
std_logic_vector(width-1 DOWNTO 0);

```

```

    VARIABLE mem: memory_array;
    VARIABLE address : natural;

```

```

    VARIABLE write_data: std_logic_vector(width-1 DOWNTO 0);

```

PROCEDURE power_up (mem: inout memory_array; clear: boolean) IS

```

    VARIABLE init_value: std_logic;

```

BEGIN

```

    IF clear THEN
        init_value := '0';
        write(output, string("Initializing SRAM with zero ...") );
    ELSE
        init_value := 'X';
    END IF;
    FOR add IN low_address TO high_address LOOP
        FOR j IN (width-1) DOWNTO 0 LOOP
            mem(add)(j) := init_value;
        END LOOP;
    END LOOP;

```

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		83

END power_up;

PROCEDURE load (mem: INOUT memory_array; download_filename: IN string) IS

FILE source : text IS IN download_filename;
VARIABLE inline, outline : line;
VARIABLE add: natural;
VARIABLE c : character;
VARIABLE source_line_nr: integer := 1;
VARIABLE init_value: std_logic := 'U';

BEGIN

write(output, string("Loading SRAM from file ") & download_filename & string(" ... "));

WHILE NOT endfile(source) LOOP

readline(source, inline);

read(inline, add);

read(inline, c);

IF (c /= ' ') THEN

write(outline, string("Syntax error in file "));

write(outline, download_filename);

write(outline, string(", line "));

write(outline, source_line_nr);

writeline(output, outline);

ASSERT FALSE

REPORT "RAM loader aborted."

SEVERITY FAILURE;

END IF;

FOR i IN (width -1) DOWNTO 0 LOOP

read(inline, c);

IF (c = '1') THEN

mem(add)(i) := '1';

ELSE

IF (c /= '0') THEN

write(outline, string("-W- Invalid character "));

write(outline, c);

write(outline, string(" in Bitstring in "));

write(outline, download_filename);

write(outline, '(');

write(outline, source_line_nr);

write(outline, string(") is set to '0'"));

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		84

```

        writeline(output, outline);
    END IF;
    mem(add)(i) := '0';
    END IF;
END LOOP;
IF (trace_ram_load) THEN
    write(outline, string("RAM("));
    write(outline, add);
    write(outline, string(") := "));
    write(outline, mem(add));
    writeline(output, outline );
END IF;
source_line_nr := source_line_nr + 1;

END LOOP;

END load;

PROCEDURE do_dump (mem: INOUT memory_array;
    dump_start, dump_end: IN natural;
    dump_filename: IN string) IS

    FILE dest : text IS OUT dump_filename;
    VARIABLE l : line;
    VARIABLE c : character;

BEGIN

    IF (dump_start > dump_end) OR (dump_end >= size) THEN
        ASSERT FALSE
        REPORT "Invalid addresses for memory dump. Cancelled."
        SEVERITY ERROR;
    ELSE
        FOR add IN dump_start TO dump_end LOOP
            write(l, add);
            write(l, ' ');
            FOR i IN (width-1) downto 0 LOOP
                write(l, mem(add)(i));
            END LOOP;
            writeline(dest, l);
        END LOOP;
    END IF;
END;

```

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		85

```

END do_dump;

BEGIN
power_up(mem, clear_on_power_up);
IF download_on_power_up THEN
  load(mem, download_filename);
END IF;
LOOP
  IF do_write'EVENT and (do_write = '1') then
    IF NOT Check_For_Valid_Data(D) THEN
      IF D'EVENT AND Check_For_Valid_Data(D'DELAYED) THEN
        write(output, "-W- Data changes exactly at end-of-write to SRAM.");
        write_data := D'delayed;
      ELSE
        write(output, "-E- Data not valid at end-of-write to SRAM.");
        write_data := undef_vec;
      END IF;
    ELSIF NOT D'DELAYED(tHD_min)'STABLE(tSD_min) THEN
      write(output, "-E- tSD violation: Data input changes within setup-time at
end-of-write to SRAM.");
      write_data := undef_vec;
    ELSIF NOT D'STABLE(tHD_min) THEN
      write(output, "-E- tHD violation: Data input changes within hold-time at
end-of-write to SRAM.");
      write_data := undef_vec;
    ELSIF nWE'DELAYED(tHD_min)'STABLE(tPWE_min) THEN
      write(output, "-E- tPWE violation: Pulse width of nWE too short at
SRAM.");
      write_data := undef_vec;
    ELSE write_data := D;
    END IF;
    mem(CONV_INTEGER(valid_adr)) := write_data;
  END IF;
  IF Check_For_Valid_Data(valid_adr) THEN
    read_data <= mem(CONV_INTEGER(valid_adr));
  ELSE
    read_data <= undef_vec;
  END IF;
  IF dump AND dump'EVENT THEN do_dump(mem, dump_start, dump_end,
dump_filename);
  END IF;

```

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		86

```

IF download AND download'EVENT THEN load(mem, download_filename);
END IF;
WAIT ON do_write, valid_adr, dump, download;
END LOOP;

```

```

END PROCESS memory;

```

```

adr_setup <= TRANSPORT A AFTER tAA_max;
adr_hold <= TRANSPORT A AFTER tOHA_min;

```

```

valid_adr <= adr_setup WHEN Check_For_Valid_Data(adr_setup)
AND (adr_setup = adr_hold)
AND adr_hold'STABLE(tAA_max - tOHA_min) ELSE
undef_adr_vec;

```

```

read_active <= ( (nOE = '0') AND (nOE'DELAYED(tLZOE_min) = '0')
AND nOE'STABLE(tLZOE_min)
AND ((nWE = '1') OR (nWE'DELAYED(tHZWE_max) = '0'))
AND (nCE = '0') AND (CE2 = '1') AND nCE'STABLE(tLZCE_min)
AND CE2'STABLE(tLZCE_min))
OR (read_active AND (nOE'DELAYED(tHZOE_max) = '0')
AND (nWE'DELAYED(tHZWE_max) = '1')
AND (nCE'DELAYED(tHZCE_max) = '0') AND
(CE2'DELAYED(tHZCE_max) = '1'));

```

```

read_valid <= ( (nOE = '0') AND nOE'STABLE(tDOE_max)
AND (nWE = '1') AND (nWE'DELAYED(tHZWE_max) = '1')
AND (nCE = '0') AND (CE2 = '1') AND nCE'STABLE(tACE_max)
AND CE2'STABLE(tACE_max))
OR (read_valid AND read_active);

```

```

D <= read_data WHEN read_valid and read_active ELSE
undef_vec WHEN not read_valid and read_active ELSE
tristate_vec;

```

```

PROCESS (nWE, nCE, CE2)
BEGIN
IF ((nCE = '1') OR (nWE = '1') OR (CE2 = '0'))
AND (nCE'DELAYED = '0') AND (CE2'DELAYED = '1') AND
(nWE'DELAYED = '0') -- End of Write
THEN

```

						ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат			87

```

do_write <= '1' AFTER tHD_min;
ELSE
  IF (Now > 10 NS) AND (nCE = '0') AND (CE2 = '1') AND (nWE = '0') -- Start
of Write
  THEN
    ASSERT Check_For_Valid_Data(A)
    REPORT "Address not valid at start-of-write to RAM."
    SEVERITY FAILURE;

    ASSERT A'STABLE(tSA_min)
    REPORT "tSA violation: Address changed within setup-time at start-of-
write to SRAM."
    SEVERITY ERROR;

    ASSERT enable_nWE_only_control OR ((nOE = '1') AND
nOE'STABLE(tSA_min))
    REPORT "tSA violation: nOE not inactive at start-of-write to RAM."
    SEVERITY ERROR;
  END IF;
do_write <= '0';
END IF;
END PROCESS;

```

```

PROCESS (A)

```

```

BEGIN

```

```

  IF (Now > 0 NS) THEN

```

```

    ASSERT (nCE = '1') OR (CE2 = '0') OR (nWE = '1')

```

```

    REPORT "Address not stable while write-to-SRAM active"

```

```

    SEVERITY FAILURE;

```

```

    ASSERT (nCE = '1') OR (CE2 = '0') OR (nWE = '1')

```

```

      OR (nCE'DELAYED(tHA_min) = '1') OR (CE2'DELAYED(tHA_min) =
'0')

```

```

      OR (nWE'DELAYED(tHA_min) = '1')

```

```

    REPORT "tHA violation: Address changed within hold-time at end-of-write to
SRAM."

```

```

    SEVERITY FAILURE;

```

```

  END IF;

```

```

END PROCESS;

```

```

PROCESS (nOE, nWE, nCE, CE2)

```

```

BEGIN

```

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
						88
Змн.	Арк.	№ докум.	Підпис	Дат		


```

IF (Now > 0 NS) AND (nCE /= '1') AND (CE2 /= '0') THEN
  IF (nCE = '0') AND (CE2 = '1') THEN
    ASSERT (nWE = '0') OR (nWE = '1')
    REPORT "Invalid nWE-signal at SRAM while nCE is active"
    SEVERITY WARNING;
  ELSE
    IF (nCE /= '0') THEN
      ASSERT (nOE = '1')
      REPORT "Invalid nCE-signal at SRAM while nOE not inactive"
      SEVERITY WARNING;

      ASSERT (nWE = '1')
      REPORT "Invalid nCE-signal at SRAM while nWE not inactive"
      SEVERITY ERROR;
    END IF;
    IF (CE2 /= '1') THEN
      ASSERT (nOE = '1')
      REPORT "Invalid CE2-signal at SRAM while nOE not inactive"
      SEVERITY WARNING;

      ASSERT (nWE = '1')
      REPORT "Invalid CE2-signal at SRAM while nWE not inactive"
      SEVERITY ERROR;
    END IF;
  END IF;
END IF;
END PROCESS;

END behavior;

```

					ДП. КСМ. 07208/11. 00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		89