

**К. В. Огородник, Б. П. Книш, П. М. Ратушний, О. О. Лазарев**

# **МОДЕЛЮВАННЯ В ЕЛЕКТРОНІЦІ**

Міністерство освіти і науки України  
Вінницький національний технічний університет

**К. В. Огородник, Б. П. Книш, П. М. Ратушний, О. О. Лазарев**

# **МОДЕЛЮВАННЯ В ЕЛЕКТРОНІЦІ**

**Навчальний посібник**

Вінниця  
ВНТУ  
2017

УДК [621.38-047.58+004.94](074)

М74

Рекомендовано до друку Вченою радою Вінницького національного технічного університету Міністерства освіти і науки України (протокол № 12 від 30.03 2017 р.).

Рецензенти:

**В. С. Осадчук**, доктор технічних наук, професор

**Л. Б. Ліщинська**, доктор технічних наук, професор

**О. С. Городецька**, кандидат технічних наук, доцент

**М74 Моделювання в електроніці : навчальний посібник /**  
**К. В. Огородник, Б. П. Книш, П. М. Ратушний, О. О. Лазарев. –**  
**Вінниця : ВНТУ, 2017. – 118 с.**

У навчальному посібнику розглядаються основи теорії моделювання в електроніці. Зокрема розглядаються питання: моделювання за допомогою PSpice OrCAD, моделі сигналів і компонентів аналогової та цифрової електроніки.

Метою навчального посібника є ознайомлення читача з сучасними принципами моделювання електронних систем. Значна увага приділяється опису елементів електроніки.

Посібник розроблено відповідно до навчальної програми з дисципліни «Моделювання в електроніці» та розраховано для студентів спеціальності 171 – «Електроніка».

УДК [621.38-047.58+004.94](074)

## ЗМІСТ

ВСТУП.....	5
1 ЗАГАЛЬНІ ВІДОМОСТІ ПРО МОДЕЛЮВАННЯ В ЕЛЕКТРОНІЦІ.....	7
1.1 Поняття моделювання.....	7
1.2 Класифікація параметрів при моделюванні.....	9
1.3 Види аналізу та розрахунку електронних схем.....	11
1.4 Класифікація моделей.....	13
2 МОДЕЛЮВАННЯ ЗА ДОПОМОГОЮ PSpICE ORCAD.....	16
2.1 Структура текстового завдання на моделювання.....	16
2.2 Директиви моделювання OrCAD.....	22
2.2.1 AC Sweep – розрахунок частотних характеристик та рівня шуму.....	25
2.2.2 Load Bias Point – завантаження даних режиму за постійним струмом.....	30
2.2.3 Save Bias Point – збереження даних режиму за постійним струмом.....	30
2.2.4 DC Sweep – варіація параметрів при розрахунку режиму за постійним струмом.....	32
3 МОДЕЛІ КОМПОНЕНТІВ АНАЛОГОВОЇ ЕЛЕКТРОНІКИ.....	34
3.1 Базовий набір моделей елементів.....	34
3.2 Опис моделей резисторів.....	35
3.3 Опис моделей індуктивностей.....	38
3.4 Опис моделей конденсаторів.....	40
3.5 Опис моделей трансформаторів.....	43
3.6 Опис моделей лінії передачі.....	52
3.7 Опис моделей керованих ключів.....	54
3.8 Моделі напівпровідникових приладів.....	54
3.8.1 Опис моделі напівпровідникового діода.....	55
3.8.2 Опис моделі біполярного транзистора.....	59
3.8.3 Опис моделі польового транзистора.....	62
3.9 Моделювання схем засобами PSpice OrCAD.....	64
4 МОДЕЛІ КОМПОНЕНТІВ ЦИФРОВОЇ ЕЛЕКТРОНІКИ.....	70
4.1 Основні поняття опису цифрових компонентів.....	70
4.2 Загальний опис цифрових компонентів PSpice OrCAD.....	74
4.3 Опис багаторозрядних АЦП та ЦАП.....	76
4.4 Вентилі.....	80

4.5 Пристрої живлення.....	82
4.6 Генератори цифрових сигналів.....	83
4.6.1 Визначення форми цифрового сигналу за форматом (пристрої STIM) .....	83
4.6.2 Зчитування тимчасової діаграми цифрового сигналу з файлу (пристрої FSTREAM) .....	85
4.7 Програмовані логічні матриці.....	88
4.8 Запам'ятовувальні пристрої.....	93
4.8.1 Постійні запам'ятовувальні пристрої (ПЗП) .....	93
4.8.2 Оперативні запам'ятовувальні пристрої (ОЗП) .....	96
4.9 Джерела постійних логічних сигналів.....	100
4.10 Цифрові лінії затримки.....	101
5 МОДЕЛІ СИГНАЛІВ.....	103
5.1 Незалежні джерела сигналів.....	103
5.1.1 Стандартні сигнали.....	103
5.1.2 Експоненціальна функція.....	104
5.1.3 Кусково-лінійна функція.....	106
5.1.4 Синусоподібна функція з частотною модуляцією.....	107
5.1.5 Сигнали довільної форми.....	107
5.2 Залежні джерела сигналів.....	108
5.2.1 Джерело напруги, кероване напругою.....	108
5.2.2 Джерело струму, кероване струмом.....	109
5.2.3 Джерело струму, кероване напругою.....	109
5.2.4 Джерело напруги, кероване струмом.....	110
5.3. Приклад обрахунку залежного джерела сигналу.....	113
СПИСОК ЛІТЕРАТУРИ.....	117

## ВСТУП

Проектування в електроніці зводиться до вирішення групи задач синтезу і задач аналізу. При цьому під структурним синтезом розуміють створення (інтуїтивне або формалізоване) якогось варіанта схеми, не обов'язково остаточного. В процесі проектування синтез як задача може виконуватися багато разів, чергуючись з вирішенням задач аналізу. До задач аналізу входить вивчення властивостей схеми за заданою в результаті синтезу її структури, характеру вхідних компонентів і їх параметрів [1, 2].

Методи аналізу і розрахунку електронних схем постійно розвиваються та удосконалюються. Причин цьому декілька.

По-перше, стрімко ускладнюється сам предмет аналізу за рахунок:

- якісного переродження елементної бази (від ламп до транзисторів, мікросхем, мікропроцесорів, приладів функціональної електроніки);
- виникнення нових принципів побудови пристроїв підсилення, обробки електричних сигналів, перетворення електричної енергії;
- розширення асортименту приладів і схем з істотно нелінійними характеристиками (тиристори, диністори, одноперехідні транзистори, оптрони, лямбда-транзистори, тунельні діоди, магнітотранзисторні елементи тощо);
- впровадження нових дискретно-імпульсних режимів роботи електронних схем перетворення інформації та електричної енергії.

По-друге, якісний стрибок відбувається в технічних засобах аналізу і розрахунку електронних схем (від логарифмічної лінійки до мікрокалькуляторів, мікрокомп'ютерів, персональних та універсальних ЕОМ), які можуть тепер проводити не тільки числові розрахунки, але й вирішувати складні логічні задачі.

По-третє, підвищуються вимоги до точності, масштабності і глибини аналізу та розрахунку електронних схем, оскільки сучасна технологія виробництва (наприклад, мікросхем) унеможлиблює їх експериментальне доведення, а вимоги до технічних і метрологічних параметрів електронних пристроїв постійно підвищуються.

По-четверте, ускладнюється вигляд сигналів, що впливають на схему за рахунок масової появи в їх складі так званих розривних функцій [3].

Тому метою навчального посібника є ознайомлення читача з сучасними принципами моделювання електронних систем. Значна увага приділяється опису елементів електроніки.

В основу посібника покладено курс лекцій з дисципліни «Моделювання в електроніці».

Посібник складається з п'яти розділів, що містять загальні поняття моделювання, класифікацію параметрів при моделюванні, моделі аналогових та цифрових приладів, моделі сигналів.

Посібник адресований широкому колу читачів, що займаються розробкою електронних систем та розрахований на студентів спеціальності 171 – «Електроніка» й може бути корисним студентам інших спеціальностей.



## 1 ЗАГАЛЬНІ ВІДОМОСТІ ПРО МОДЕЛЮВАННЯ В ЕЛЕКТРОНІЦІ

### 1.1 Поняття моделювання

Математичне моделювання полягає в переведенні явищ навколишнього світу на мову математики. Поняття «модель» відображає поведінку об'єкта або процесу. Крайніми випадками поняття «моделювання» є, з одного боку, натурний експеримент (повне повторення поведінки об'єкта або процесу), з іншого – розрахунково-аналітичне моделювання (чисто математичний опис). Між цими крайніми випадками розташовуються проміжні методи дослідження. Вони дозволяють скоротити або ж спростити натурний експеримент [1].

До таких методів відносяться:

1. Фізичне моделювання, при якому натура та модель мають однакову фізичну природу, але різні, наприклад, геометричні розміри;

2. Моделі прямої аналогії. В цьому випадку натура і модель мають різну фізичну природу, але однаковий математичний опис, наприклад, теплові процеси моделюються електричними;

3. Розрахунково-аналітичний метод або математичне моделювання, яке використовується для створення математичного опису роботи об'єкта та дослідження його роботи за допомогою ЕОМ [2, 3].

Математичні моделі загалом можна поділити на:

- аналітичні, що побудовані за фізичними законами, які відображають зміст явища;

- експериментально-статистичні, що відповідають набору статистичних даних [3].

Експериментально-статистичні моделі служать для опису об'єкта як «чорного ящика», тому обґрунтування верифікації (достовірності) такої математичної моделі є необхідною умовою моделювання.

Створена модель зазвичай уточнюється шляхом постановки натурального експерименту для підтвердження точності та достовірності математичної моделі.



Похибка математичного моделювання при цифровій реалізації є похибкою методичною, яка пояснюється неточністю математичної моделі внаслідок спрощення реальної системи при її математичному описі (оскільки похибка обчислень може бути зведена до достатньо малої величини).

Підвищення складності математичної моделі задля забезпечення вищої точності результатів має природне обмеження зі сторони швидкості обробки інформації за допомогою персонального комп'ютера. Також, якщо вихідні дані відомі з похибкою до 5%, то втрачає зміст розробка моделі, що забезпечує похибку до 1%. Зазвичай при моделюванні однією із найважливіших задач є знаходження компромісу між достатньою складністю моделі для забезпечення її адекватної достовірності та достатньою спрощеністю моделі для спрощення та пришвидшення відповідних розрахунків її параметрів [4 – 7].

Для моделювання електричних схем використовуються різні мови їх опису. Найбільш поширеними і загальноприйнятими є дві мови.

1. Spice.
2. VHDL [8].

Мова Spice (Simulation Program with Integrated Circuit Emphasis) розроблена наприкінці 70-х років в університеті Берклі (США, Каліфорнія). На основі цієї мови працюють такі програмні пакети, як MicroCap, OrCad, MultiSim та багато інших.

Програми на основі мови Spice мають набір математичних моделей, типових компонентів (діоди, резистори, біполярні транзистори, лінії затримки тощо). Параметри математичних моделей компонентів записуються в бібліотечні файли або складаються файли окремих моделей. Наприклад, програма OrCAD 9.2 має у своєму складі математичні моделі десятків тисяч компонентів, а ліцензовані користувачі отримують доступ до розширеної бази даних, у якій наявні відомості про понад 200 тисяч компонентів різноманітних фірм.

Під час електричного моделювання програмні пакети надають можливість створення власних моделей на електричному рівні, також передбачається можливість створення макромоделей (ієрархічних блоків будь-якого ступеня вкладеності). Використання макромоделей не тільки зменшує складність відображення схеми, але й пришвидшує обрахунки. Під час першого запуску програми на обчислення вихідних характеристик

створюється індексний файл \*.IND, який надалі використовується для обчислень, пришвидшуючи роботу системи.

Програмні пакети на основі мови Spice використовуються для моделювання аналогових, цифрових та змішаних цифро-аналогових схем [3, 6].

VHDL (Very-High-Speed Hardware Description Language) – алгоритмічна мова опису апаратури високого рівня. Описує алгоритми роботи пристроїв через послідовність операторів присвоювання та прийняття рішень. Використовується для проектування та моделювання роботи цифрових пристроїв. З'явилась на початку 80-х років за запитом Міністерства оборони США. В 1987 році прийнята як стандарт IEEE (Міжнародний інститут з електроніки) VHDL-87. Згодом було прийнято її розширений стандарт VHDL-93 [8].

## **1.2 Класифікація параметрів при моделюванні**

При проектуванні та моделюванні схем часто вживається термін «параметр». Параметром є деяка величина, що характеризує елемент, схему або пристрій. Параметри поділяються на параметри схеми і параметри технічного завдання як проілюстровано в класифікації параметрів при проектуванні та моделюванні електронних схем (рис. 1.1).

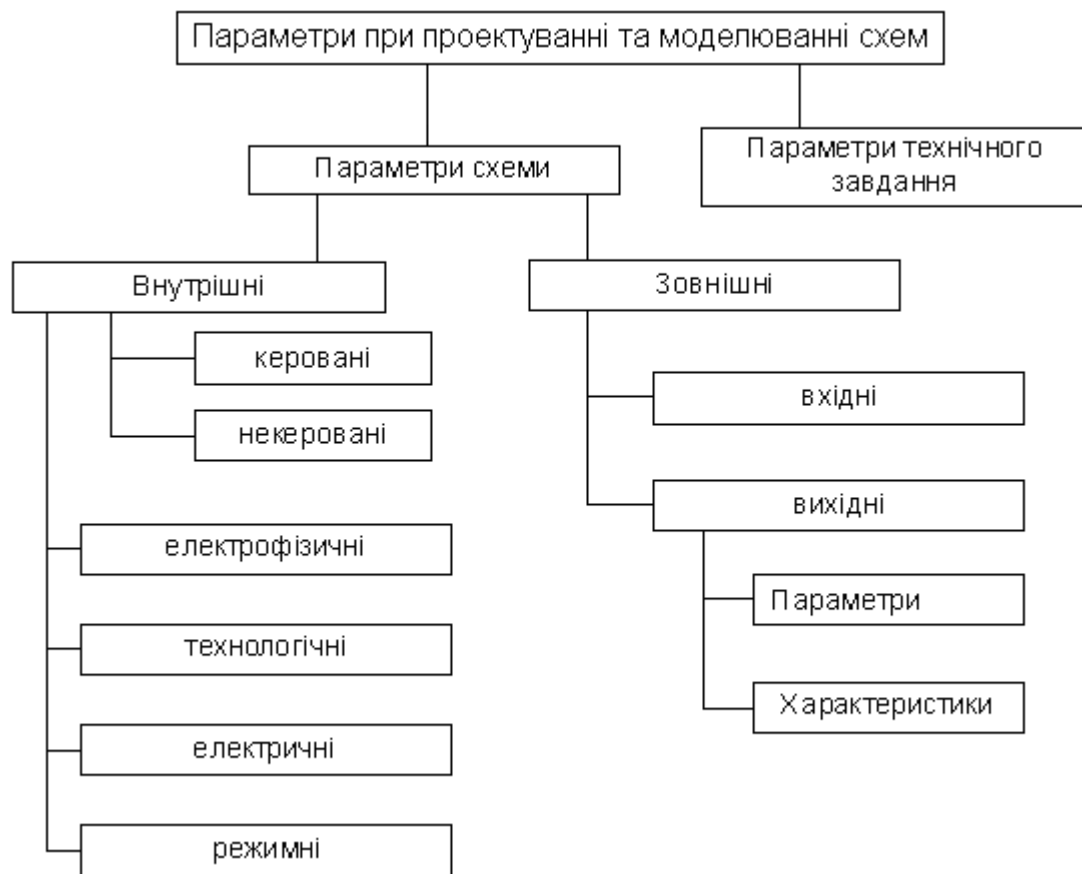


Рисунок 1.1 – Класифікація параметрів при проектуванні та моделюванні електронних схем

Параметри схеми також поділяються на внутрішні й зовнішні.

До внутрішніх параметрів відносять параметри елементів (компонентів) схем: діодів, резисторів, конденсаторів, транзисторів, мікросхем тощо. Вони поділяються на топологічні, електрофізичні, електричні й режимні. Наприклад, у польовому транзисторі концентрація домішки в напівпровіднику є електрофізичним параметром, довжина й ширина каналу – топологічними параметрами, крутизна передатної характеристики – електричним параметром, граничні значення струмів і напруг – режимними параметрами.

Серед внутрішніх параметрів окремо виділяють керовані й некеровані параметри. Керованими параметрами є ті, зміну яких фізично легко здійснити. До некерованих параметрів відносять ті, зміна яких у рамках розглянутого завдання є або нездійсненною, або недоцільною.

Зовнішні параметри схеми поділяють на вхідні та вихідні.

Вхідні параметри, як правило, визначаються технічним завданням на розроблювальний пристрій, наприклад, чутливість підсилувача, напруга

живлення, динамічний діапазон сигналів тощо.

Вихідні параметри – це параметри, за якими здійснюється оцінювання робоздатності схеми: наприклад, максимальна амплітуда вихідної напруги підсилувача на заданій частоті, вихідний опір і коефіцієнт стабілізації джерела вторинного електроживлення тощо. Крім вихідних параметрів, що є числовими характеристиками схеми, вводять поняття вихідних характеристик, тобто функціональних залежностей вихідних параметрів від деяких величин, наприклад, частоти вхідного сигналу, зміни напруги живлення, струму в навантаженні тощо.

Під час оптимізації електронних схем використовують термін **варійовані параметри**, тобто зовнішні й внутрішні параметри, зміна яких здатна привести до покращення показників системи.

Параметри технічного завдання – це параметри зовнішнього середовища та граничні режимні параметри, наприклад, напруга живлення та допуски на неї, діапазон робочих температур і тисків, вологості, граничні значення напруг і потужностей в окремих точках схеми. Параметри технічного завдання використовуються як система одно- або двосторонніх обмежень, що накладаються на зовнішні параметри проектованої схеми.

Термін «розрахунок схеми» означає визначення її параметрів і характеристик за незмінних значень внутрішніх параметрів схеми і її структури, наприклад, розрахунок статичного режиму, перехідного процесу, АЧХ і ФЧХ схеми.

Для розрахунку схеми важливим є правильний вибір базисних змінних, тобто тих змінних, які використовуються для математичного опису схеми (струми елементів, вузлові потенціали, гібридний опис).

Аналіз схеми – це процес визначення зміни вихідних і режимних параметрів залежно від зміни варійованих параметрів [5].

### **1.3 Види аналізу та розрахунку електронних схем**

Найбільш поширеними видами розрахунку електронних схем є:

- розрахунок статичного режиму;
- розрахунок малосигнальних параметрів;
- розрахунок перехідних процесів;
- розрахунок вихідних параметрів.

Розрахунок статичного режиму схеми виконується для побудови карти

режимів за постійним струмом, визначення різних статичних параметрів, наприклад, потужності, що виділяється на елементах. Даний вид розрахунку використовується для визначення початкових умов для розрахунку перехідних процесів і визначення робочої точки для розрахунку малосигнальних параметрів.

Розрахунок малосигнальних параметрів схеми й розрахунок перехідних процесів є основою для визначення динамічних параметрів схеми. Це дозволяє зробити висновок про роботоздатність схеми при різних вхідних впливах.

Розрахунок вихідних параметрів схеми роблять після розрахунків статичного режиму й перехідних процесів. Він дозволяє одержати дані про відповідність або невідповідність зовнішніх параметрів схеми параметрам, зазначеним у технічному завданні.

Найбільш поширені види аналізу схем.

1. Аналіз параметричної чутливості, тобто визначення ступеня впливу зміни внутрішніх параметрів схеми на вихідні параметри.

2. Статистичний аналіз дозволяє знайти характеристики схеми при статистичному відхиленні параметрів елементів схем (внутрішніх параметрів). Як результат можуть бути побудовані гістограми вихідних параметрів, за якими можна визначити границі відбраковування придатних виробів.

3. Аналіз на найгірший випадок – передбачає розрахунок вихідних параметрів схеми при найгіршому співвідношенні її внутрішніх параметрів.

4. Аналіз впливу зовнішнього середовища складається з розрахунку впливу температури, тиску, вологості тощо на вихідні параметри. Фактично аналіз виконується у два етапи:

- спочатку з'ясовуються залежності параметрів компонентів електронної схеми від зміни зовнішнього середовища;

- здійснюється розрахунок залежності зовнішніх параметрів від внутрішніх.

5. Багатоваріантний аналіз являє собою розрахунок схеми при різних комбінаціях параметрів елементів, які задаються проектувальником, що проводиться за один сеанс розрахунку. У ряді випадків багатоваріантний аналіз схеми дозволяє обійтися без оптимізації параметрів елементів схеми, оскільки розробник одержує результати розрахунку сукупності схем і може вибрати найкращий варіант.

Аналіз і розрахунок електронних схем здійснюється на основі моделювання реальних компонентів, тобто заміни їх деякими моделями, які з тим або іншим ступенем точності описують поведінку елементів схеми в різних режимах, тобто на основі створення математичних моделей компонентів. Математичну модель елемента можна розглядати як деякий оператор  $F$ , що ставить у відповідність системі внутрішніх параметрів компонента або схеми  $X=x_1, x_2, \dots, x_n$  сукупність пов'язаних з ними зовнішніх параметрів  $Y=y_1, y_2, \dots, y_m$ . Склад внутрішніх і зовнішніх параметрів, їх фізичний зміст змінюється залежно від призначення моделі.

Для моделей компонентів зовнішніми параметрами є, звичайно, струми й напруги, якщо модель призначена для розрахунку схем. Для розрахунку самого компонента зовнішніми параметрами виступають, звичайно, густина струму, розподіл заряду тощо. Внутрішніми параметрами в першому випадку є електричні параметри, а в другому випадку – електрофізичні й конструктивно-технологічні. Зв'язок між зовнішніми й внутрішніми параметрами при розрахунку схем здійснюється на основі законів Ома та Кірхгофа, а при розрахунку самого компонента – з використанням рівнянь переносу, безперервності, Пуассона.

Для моделей схем внутрішніми параметрами є електричні параметри їх компонентів, а зовнішніми параметрами – струми, напруги, потужності, частотні та інші характеристики. Зв'язок між зовнішніми й внутрішніми параметрами здійснюється з використанням законів Кірхгофа (для консервативних систем) і законів Максвелла (для систем відкритих, де можливі втрати на випромінювання, що не враховані законами Кірхгофа) [6].

#### **1.4 Класифікація моделей**

Математичні моделі технічних пристроїв можуть бути класифіковані за рядом ознак.

1. За характером процесів, які вони відображають:
  - статичні моделі;
  - динамічні моделі.
2. За способом наведення моделі поділяють на:
  - аналітичні;
  - графічні;
  - табличні.

Аналітичні моделі визначають прилад або компонент у вигляді рівнянь, що описують його вольт-амперні характеристики (ВАХ), або у формі диференціальних рівнянь, що описують перехідні процеси у моделювальній схемі й характеризують інерційність елемента.

Графічні моделі дозволяють подати компоненти у вигляді графіків ВАХ або у вигляді еквівалентних схем заміщення.

Табличні моделі дозволяють подати схему або елемент у вигляді цифрових таблиць, отриманих у ході експериментального дослідження об'єкта моделювання й відповідних графіків експериментальних ВАХ. Табличні моделі використовують зазвичай в тому випадку, коли аналітичну модель побудувати важко внаслідок складної залежності. Іноді при складних функціональних аналітичних залежностях для ВАХ їх свідомо табулюють і створюють, таким чином, можливо, менш точну, але більш зручну модель.

Перераховані вище моделі можуть бути виконані у вигляді підпрограм. При такому поданні вони перетворюються на цифрову модель.

Аналітичні й графічні моделі можуть бути також задані у вигляді алгоритму обчислення зовнішніх параметрів моделі, при цьому модель називається алгоритмічною моделлю. Після оформлення підпрограми відповідно до наведеного алгоритму модель стає цифровою.

Цифрові моделі можуть бути досить точними, оскільки ступінь їхньої складності визначається, в основному, складністю програми й допустимими для розрахунку витратами машинного часу. У цей час цифрові моделі використовуються усе ширше, що пов'язано з розвитком САПР РЕА.

3. За характером залежностей моделі поділяються на:

- лінійні;
- нелінійні.

Є особливий клас кусково-лінійних моделей, нелінійність яких проявляється в обмеженій кількості точок стику лінійних ділянок.

Нелінійні моделі, природно, виявляються більш точними, але й більш складними.

4. За діапазоном робочих сигналів моделі класифікуються на:

- моделі великого сигналу;
- малосигнальні.

Малосигнальні моделі, як правило, являють собою лінійні моделі. Моделі ж для великого сигналу враховують нелінійність характеристик

активних і пасивних елементів схеми (біполярних і польових підсилювальних приладів).

5. За діапазоном робочих частот виділяють:

- низькочастотні;
- високочастотні;
- надвисокочастотні моделі.

Низькочастотні моделі не враховують інерційність компонентів моделі, а тому використовуються для розрахунку схем за постійним струмом (у статичному режимі).

Високочастотні моделі – моделі більш високого рівня, вони враховують, крім особливостей статичного режиму, інерційність компонентів. Тому такі моделі доповнюють системою диференціальних рівнянь, яка враховує інерційність компонентів, або еквівалентними схемами реальних приладів на високих частотах – індуктивностями і ємностями виводів, інерційностями, які визначають фізичні процеси в компонентах (наприклад, накопичення заряду), ємностями областей структур тощо.

Особливість низькочастотних і високочастотних моделей полягає в тому, що вони створюються на зосереджених елементах і тому для цих моделей справедливі закони Кірхгофа.

Надвисокочастотні моделі відрізняються від високочастотних моделей врахуванням просторових і тимчасових координат, тому для аналізу й розрахунку НВЧ-схем необхідно використовувати рівняння Максвелла. Застосування законів Кірхгофа виправдано лише в діапазоні частот до 10 ГГц, де розміри компонентів (особливо компонентів інтегральних схем)



## КОНТРОЛЬНІ ЗАПИТАННЯ

---

залишаються меншими довжини хвилі  $\lambda = 3$  см [7].

1. Які види моделей існують?
2. Які мови для моделювання електричних схем є найбільш поширеними і загальноприйнятими? Коротко охарактеризуйте кожну з них.
3. Наведіть коротку класифікацію параметрів та характеристик, які використовуються для проектування та моделювання електронних схем.
4. У чому різниця між аналізом та розрахунком схеми?



5. Що являє собою багатоваріантний аналіз схеми?
6. За якими ознаками класифікуються моделі?
7. У чому переваги й недоліки нелінійних моделей?
8. Які особливості мають надвисокочастотні моделі?



## ***2 МОДЕЛЮВАННЯ ЗА ДОПОМОГОЮ PSpICE ORCAD***

### **2.1 Структура текстового завдання на моделювання**

Завдання на моделювання для програми PSpice заноситься в текстові файли. Знання їх форматів при графічному вводі схеми не є обов'язковим, але бажаним, оскільки це:

- значно полегшує пошук помилок при налагоджуванні схеми;
- дозволяє створювати шаблони нових символів компонентів;
- дозволяє складати текстові описи макромоделей (іноді це простіше, ніж рисувати їх схеми заміщення).

При графічному вводі схем як за допомогою програми PSpice Schematics, так і за допомогою OrCAD Capture, створюються три файли завдання з одним і тим же ім'ям і різними розширеннями: \*.NET (таблиця з'єднань), \*.ALS (список підключення кіл до виводів компонентів) та \*.CIR (список директив моделювання). При моделюванні в PSpice безпосередньо завантажується файл \*.CIR, в якому є посилання на інші файли. Для його складання вручну на папері спочатку рисується принципова схема пристрою, що моделюється, та присвоюються імена всім її вузлам (при графічному введенні цього робити необов'язково). Імена вузлів можуть бути цілими числами від 0 до 999 або алфавітно-цифровими символами довжиною не більше 131 символу. Як такі символи використовуються літери латинського алфавіту від A до Z, цифри 0, 1, ..., 9 і знаки «\$», «\_», «\*», «/», «%». Глобальні вузли наведено в табл. 2.1.

Таблиця 2.1 – Глобальні вузли

Ім'я вузла	Напруга/рівень	Опис
0	0 В	Аналогова «земля»
\$ G_CD4000_VDD	5 В	Джерело живлення МОН ІС
\$ G_CD4000_VSS	0 В	Джерело живлення МОН ІС
\$G_DPWR	5 В	Джерело живлення ТТЛ ІС
\$ G_DGND	0 В	Джерело живлення ТТЛ ІС

Продовження таблиці 2.1

Ім'я вузла	Напруга/рівень	Опис
\$ G_ECL_10K_VEE	-5.2 В	Джерело живлення ЕСЛ 10К ІС
\$ G_ECL_10K_VCC1	0 В	Джерело живлення ЕСЛ 10К ІС
\$ G_ECL_10K_VCC2	0 В	Джерело живлення ЕСЛ 10К ІС
\$ G_ECL_100K_VEE	-4,5 В	Джерело живлення ЕСЛ 100К ІС
\$ G_ECL_100K_VCC1	0 В	Джерело живлення ЕСЛ 100К ІС

		IC
$\$ G\_ECL\_100K\_VCC2$	0 В	Джерело живлення ЕСЛ 100К IC
$\$ D\_HI$	«1»	Логічна «1»
$\$ D\_LO$	«0»	Логічний «0»
$\$ D\_X$	«X»	Невизначений логічний стан X
$\$ D\_NC$	–	Не підключений до схеми вивід цифрового компонента

Логічні стани цифрових вузлів  $\$ D\_HI$ ,  $\$ D\_LO$ ,  $\$ D\_X$  підтримуються постійними незалежно від того, з якими елементами вони з'єднані. Вузол  $\$ D\_NC$  застосовується для позначення виводів, які не використовуються (аббревіатура NC означає Not Connected – немає підключення), щоб при перевірці схеми вони не вносилися до списку помилок.

При посиланні на цифрові імена вузлів вони виділяються круглими дужками, наприклад  $V(6)$  – потенціал вузла 6. Імена вузлів у вигляді алфавітно-цифрових символів при посиланнях на них виділяють квадратними дужками [], щоб відрізнити їх від імен компонентів. Наприклад, потенціал вузла IN позначається як  $V([IN])$ , а  $V(12)$  – напруга на джерелі струму 12. Зауважимо, що в програмі Probe, при посиланні на алфавітно-цифрові імена вузлів квадратні дужки не потрібні, вони відсутні і в вихідному файлі з розширенням \*.OUT.

В програмі існує домовленість, що всі вузли, імена яких починаються з символів  $\$ G\_$ , є глобальними, наприклад, вузол  $\$ G\_POS$ . Глобальні вузли використовуються в схемах, які мають макромоделі. Глобальні вузли з однаковими іменами автоматично з'єднуються в основному колі і в усіх макромоделях. З їх допомогою зручно прокладати кола живлення, синхронізації тощо (вузол «земля» 0 завжди є глобальним). Глобальні вузли з довільними іменами призначаються за допомогою директиви .GLOBAL.

Після іменування вузлів складають завдання на моделювання, яке заноситься в файл. Ім'я файлу довільне. Як розширення імені рекомендується використовувати \*.CIR (сприймається програмою PSpice за замовчуванням).

Перший рядок файлу – рядок заголовка, який потім виводиться у вигляді заголовка у вихідному файлі.

Рядки коментарів містять символ «\*» в першій позиції. Кінець будь-якого рядка після знака «;» також сприймається як коментар.

Останній рядок файлу *.END*. Порядок введення проміжних рядків значення не має, за невеликим винятком:

- рядок опису функції *.FUNC* повинен бути поміщений до посилання на неї;
- директива *.OPTIONS NOECHO* забороняє запис у вихідний файл розташованої після неї частини опису схеми.

Рядок продовження починається з символа «+» в першій позиції, максимальна довжина рядка 132 символи. Число пробілів між операторами в рядку довільне.

Програма PSpice не розрізняє великі і малі літери.

Речення вхідної мови програми PSpice поділяються на описи компонентів і директиви.

Описом компонента вважається будь-який рядок, що не починається з символа «.» (крім першого рядка, рядків коментарів і рядків продовжень).

Опис компонента має таку структуру: *<ім'я компонента> <номери двох або більше вузлів> [*<ім'я моделі>*] <числові дані>*.

Ім'я компонента складається з послідовності символів латинського алфавіту і цифр, загальна довжина імені не повинна перевершувати 131 символа (рекомендується не більше 8 символів). Перший символ – одна з букв латинського алфавіту від A до Z, далі в будь-якому порядку – алфавітно-цифрові символи і знаки «\$», «\_», «\*», «/», «%».

Перші символи імен компонентів (табл. 2.2) визначають їх тип, наприклад, *R1*, *ROUT*, *Q12*.

Таблиця 2.2 – Перші символи імен компонентів

Перший символ імені	Тип компонента
<i>B</i>	Арсенід-галієвий польовий транзистор (GaAsFET) з каналом n-типу
<i>C</i>	Конденсатор
<i>D</i>	Діод
<i>E</i>	Джерело напруги, кероване напругою (ДНКН)
<i>F</i>	Джерело струму, кероване струмом (ДСКС)
<i>G</i>	Джерело струму, кероване напругою (ДСКН)
<i>H</i>	Джерело напруги, кероване струмом (ДНКС)
<i>I</i>	Незалежне джерело струму

<i>J</i>	Польовий транзистор з керуючим р-n-переходом (JFET)
<i>K</i>	Зв'язані індуктивності і лінії передачі, феромагнітні осердя
<i>L</i>	Індуктивність
<i>M</i>	МОН-транзистор (MOSFET)
<i>N</i>	Аналого-цифровий перетворювач на вході цифрового пристрою
<i>O</i>	Цифро-аналоговий перетворювач на виході цифрового пристрою
<i>Q</i>	Біполярний транзистор
<i>R</i>	Резистор
<i>S</i>	Ключ, керований напругою
<i>T</i>	Лінія передачі
<i>V</i>	Незалежне джерело напруги
<i>W</i>	Ключ, керований струмом
<i>Y</i>	Цифровий пристрій
<i>X</i>	Макромодель (операційний підсилювач, компаратор напруги, регулятор напруги, стабілізатор напруги тощо)
<i>Z</i>	Статично індукований біполярний транзистор (IGBT)

При графічному вводі схем користувач може вводити перший символ імені компонента на свій розсуд, оскільки графічні редактори при складанні текстового опису схеми для передачі його в PSpice до імен всіх компонентів відповідно до табл. 2.2 автоматично додадуть префікси – це виконується відповідно до так званих шаблонів символів компонентів *TEMPLATE*. Тому на схемах компоненти можна іменувати, не дотримуючись наведених в табл. 2.2 правил (вони повинні бути враховані при складанні шаблонів символів). Наприклад, транзистори всіх типів можна згідно з ЄСКД іменувати як *V1*, *V2*, *V3* ..., а при складанні текстового опису схеми біполярний транзистор отримає ім'я *Q\_V1*, польовий – *J\_V2*, МОН-транзистор – *M\_V3* тощо.

Окрім перерахованих вище компонентів схем до складу модельованих пристроїв можуть входити і неелектронні елементи (електричні машини, системи автоматичного управління тощо), оформлені у вигляді макромоделей.

Номери вузлів підключення компонента до схеми перераховуються в певному порядку, встановленому для кожного компонента. Ім'я моделі компонента не є обов'язковим параметром. В кінці вказуються числові значення параметрів компонента.

В програмі PSpice здійснюється масштабування чисел за допомогою масштабних коефіцієнтів, наведених в табл. 2.3.

Таблиця 2.3 – Масштабні коефіцієнти

Суфікс	Масштабний коефіцієнт	Назва
<i>f</i>	$10^{-15}$	Фемто
<i>P</i>	$10^{-12}$	Піко
<i>n</i>	$10^{-9}$	Нано
<i>U</i>	$10^{-6}$	Мікро
<i>mil</i>	$25,4 \cdot 10^6$	Міль (0,001 дюйма)
<i>m</i>	$10^{-3}$	Мілі
<i>k</i> або <i>K</i>	$10^3$	Кіло
<i>MEG</i>	$10^6$	Мега
<i>G</i>	$10^9$	Гіга
<i>T</i>	$10^{12}$	Тера
<i>C</i>		Ознака цілого числа тактів цифрових сигналів

Допускається до масштабних суфіксів дописувати літерні символи для поліпшення наочності позначень. Так, опір 5,1 кОм може бути записаний кількома способами: *5100*, *5.1K*, *5.1КОМ*, *5.1КОНМ*, *5.1e3*, *0.0051e3K*. При цьому допускається приєднувати суфікс і до чисел, наведених у формі з плаваючою точкою, як в останньому прикладі. Суфікс *mil* використовується при заданні геометричних розмірів МОН-транзисторів.

Опори, ємності й індуктивності повинні бути постійними величинами (залежність від часу або від струмів і напруг, як у програмі Micro-Cap, на жаль, не допускається). Вони можуть бути як додатними, так і від'ємними величинами. Виняток становить аналіз перехідних процесів, де від'ємні значення ємностей і індуктивностей можуть призвести до помилок у розрахунку. У будь-якому випадку нульові значення параметрів компонентів не допускаються.

За наявності помилок, виявлених при трансляції схеми і при виконанні моделювання, на екран і у вихідний файл \*.OUT виводяться повідомлення про них.

Крім числових значень в реченнях вхідної мови програми можуть використовуватися ідентифікатори параметрів, які поміщають у фігурні дужки {}. Наприклад, постійна напруга джерела живлення може бути задана за допомогою параметра *VPOWER: VP 6 0 DC {VPOWER}*.

Усередині фігурних дужок допускаються символи арифметичних виразів, наприклад:  $C2\ 5\ 4\ \{CLOAD * 2.5\}$ .

Це не допускається при заданні:

- параметрів  $NL, F, LEN, R, L, G$  і  $C$  лінії передачі;
- температурних коефіцієнтів резистора  $TC1, TC2$  в його описі (у моделі резистора ці вирази допускаються);
- параметрів кусково-лінійного сигналу  $PWL$ ;
- поліноміальних коефіцієнтів залежних джерел типу  $E, F, G, H$ ;
- в іменах вузлів та параметрах директив  $.AC, .DC$  тощо.

Вираз повинен поміщатися на одному рядку довжиною не більше 131 символа (громіздкі вирази доцільно оформляти у вигляді функцій за допомогою директиви  $.FUNCTION$ ). Крім арифметичних виразів допускається використання стандартних функцій, наведених в табл. 2.4.

Згідно з табл. 2.4 стандартними функціями є абсолютне значення  $x$  –  $ABS(x)$ , арккосинус  $x$  –  $ACOS(x)$ , арктангенс  $x$  –  $ARCTAN(x)$ , арксинус  $x$  –  $ASIN(x)$ , арктангенс  $x$  –  $ATAN(x)$ , арктангенс  $y/x$  –  $ATAN2(y, x)$ , косинус  $x$  –  $COS(x)$ , косинус гіперболічний  $x$  –  $COSH(x)$ , похідна  $dx/dt$  –  $DDT(x)$ , експонента числа  $x$  –  $EXPM$ , оператор відношення –  $W(t, x, y)$ , уявна частина  $x$  –  $IMGU$ , обмежувач з лінійною областю –  $LIMIT(x, min, max)$  [9].

Таблиця 2.4 – Стандартні функції

Функція	Коментар	Наявність в Probe
$ABS(x)$		Так
$ACOS(x)$	Результат в радіанах	Ні
$ARCTAN(x)$	Результат в радіанах	Так
$ASIN(x)$	Результат в радіанах	Ні
$ATAN(x)$	Результат в радіанах	Так
$ATAN2(y, x)$	Результат в радіанах	Ні
$COS(x)$	$x$ в радіанах	Так
$COSH(x)$	$x$ в радіанах	Ні
$DDT(x)$	Застосовується тільки при аналізі перехідних процесів	Так

<i>EXPM</i>		Так
$W(t, x, y)$	$x$ , якщо $t$ – істинна, $y$ , якщо $t$ – неістинна	Ні
<i>IMGU</i>		Так
$LIMIT(x, min, max)$	$min$ , якщо $x < min$ , $max$ , якщо $x > max$ , $x$ , якщо $min < x < max$	

## 2.2 Директиви моделювання OrCAD

Програма PSpice розраховує такі характеристики електронних кіл:

- режим кола за постійним струмом в «робочій точці» (*Bias Point*);
- режим за постійним струмом при варіації джерел постійної напруги або струму, температури та інших параметрів кола (*DC Sweep*);
- чутливість характеристик кола до варіації параметрів компонентів в режимі за постійним струмом (*Sensitivity*);
- малосигнальні передавальні функції в режимі за постійним струмом (*Transfer Function*);
- характеристики лінеаризованого кола в частотній області при впливі одного або декількох сигналів (*AC Sweep*);
- спектральну щільність внутрішнього шуму (*Noise Analysis*);
- перехідні процеси при впливі сигналів різної форми (*Transient Analysis*);
- спектральний аналіз (*Fourier Analysis*);
- статистичні випробування за методом Монте-Карло і розрахунок найгіршого випадку (*Monte Carlo / Worst Case*);
- багатоваріантний аналіз при варіації температури (*Temperature*) та інших параметрів (*Parametric*).

За допомогою модуля PSpice Optimizer виконується параметрична оптимізація.

Кожному виду розрахунку відповідає певна директива моделювання (табл. 2.5).

Таблиця 2.5 – Директиви моделювання



Ім'я	Призначення
<i>Розрахунок стандартних характеристик</i>	
.AC	Розрахунок частотних характеристик
.DC	Розрахунок режиму за постійним струмом
.FOUR	Спектральний аналіз
.NOISE	Розрахунок рівня внутрішнього шуму
.OP	Передача в вихідний файл параметрів схеми, лінеаризованої в околі робочої точки
.SENS	Розрахунок малосигнальних чутливостей в режимі за постійним струмом
.TF	Розрахунок малосигнальних передавальних функцій в режимі за постійним струмом
.TRAN	Розрахунок перехідних процесів
<i>Управління видачею результатів</i>	
.PLOT	Наведення результатів розрахунку в вихідному файлі у вигляді графіків, побудованих в текстовому режимі
.PRINT	Наведення результатів розрахунку в вихідному файлі у вигляді таблиць

Продовження таблиці 2.5

Ім'я	Призначення
.PROBE	Передача даних в графічний постпроцесор Probe
.VECTOR	Створення файлу з результатами моделювання цифрових пристроїв
.WATCH	Видача проміжних результатів аналізу на екран програми PSpice в текстовому вигляді
.WIDTH	Призначення довжини рядків вихідного файлу
<i>Багатоваріантний аналіз</i>	
.STEP	Варіація параметрів
.TEMP	Призначення температури навколишнього середовища
<i>Допоміжні файли визначення функцій і параметрів</i>	
.END	Кінець завдання
.FUNC	Визначення функції
.INC	Включення у вхідний файл іншого файлу
.LIB	Підключення бібліотеки моделей компонентів
.PARAM	Визначення глобальних параметрів
<i>Статистичний аналіз</i>	
.MC	Статистичний аналіз за методом Монте-Карло
.WCASE	Розрахунок найгіршого випадку
<i>Моделі пристроїв</i>	
.ENDS	Кінець опису макромоделі
.DISTRIBUTION	Табличне визначення закону розподілу випадкових

	величин
<i>.MODEL</i>	Опис моделей компонентів
<i>.SUBCKT</i>	Початок опису макромоделі
<i>Задання початкових умов</i>	
<i>.IC</i>	Задання початкових умов
<i>.LOADBIAS</i>	Зчитування з файлу вузлових потенціалів схеми
<i>.NODESET</i>	Завдання вузлових потенціалів за постійним струмом на початковій ітерації
<i>.SAVEBIAS</i>	Запис в файл вузлових потенціалів схеми
<i>Інші директиви</i>	
<i>.ALIASES</i>	Початок списку відповідностей імен виводів графічних позначень компонентів іменам кіл схеми, до яких вони підключені
<i>.ENDALIASES</i>	Кінець списку відповідностей

Продовження таблиці 2.5

Ім'я	Призначення
<i>.EXTERNAL</i>	Специфікація зовнішніх портів
<i>.OPTIONS</i>	Установлення параметрів і режимів роботи програми
<i>.STIMLIB</i>	Задання імені файлу з описом зовнішніх впливів
<i>.STIMULUS</i>	Задання зовнішніх впливів

Директиви моделювання задаються в схемному редакторі PSpice Schematics за командою *Analysis > Setup* або вибором відповідної піктограми. В меню, що відкривається, вибирають потрібний вид аналізу (позначаючи його хрестиком в графі *Enabled*) і клацанням миші по панелі з ім'ям виду аналізу розкривають меню задання параметрів моделювання.

При роботі з OrCAD Capture аналогічні директиви моделювання задаються/редагуються за командами *PSpice > New / Edit Simulation Profile*.

Слід звернути увагу, що в меню схемних редакторів задаються не всі можливі директиви програми PSpice. Директиви, які не ввійшли в меню (*.LIB*, *.INC* та ін.), задаються за допомогою атрибутів, що присвоюються на схемі спеціальним символам з бібліотеки *Special.sib*, кожній директиві – окремий символ [9].

### 2.2.1 AC Sweep – розрахунок частотних характеристик та рівня шуму

Частотні характеристики розраховуються за директивою: *.AC [LIN] [OCT] [DEC] <n> <початкова частота> <кінцева частота>*.

Ця директива задає діапазон частот в межах *<початкова частота>* ... *<кінцева частота>*. Параметр *LIN* встановлює лінійний крок за частотою, при цьому *n* – загальна кількість точок за частотою. Параметри *OCT* і *DEC* встановлюють логарифмічний характер зміни частоти октавами і декадами, відповідно. Параметр *n* визначає в такому випадку кількість точок за частотою на одній октаві або декаді.

Аналіз спектральної густини внутрішнього шуму проводиться за директивою: *.NOISE Y (<вузол> [, <вузол>]) <ім'я> <n>*.

Директива *.NOISE* вказується спільно з директивою *.AC*, в якій задається діапазон частот аналізу. Джерелами шуму слугують резистори, ключі та напівпровідникові прилади. На кожній частоті розраховується спектральна густина вихідної напруги  $S_{U_{ВИХ}}(f)$ , В<sup>2</sup>/Гц, обумовлена наявністю статистичних незалежних джерел внутрішнього шуму. Точки знімання вихідної напруги вказуються за специфікацією *Y (<вузол> [ <вузол>])*. До вхідних затискачів кола підключається незалежне джерело напруги або струму, *<ім'я>* якого наводиться в списку параметрів директиви *.NOISE*. Це джерело не є джерелом реального сигналу, воно слугує лише для позначення вхідних затискачів кола, для яких перераховується вихідний шум. Якщо до входу підключається джерело напруги, то на виході розраховується еквівалентна спектральна густина напруги  $S_{U_{ВХ ЕК}}(f)$ , В<sup>2</sup>/ Гц; якщо до входу підключене джерело струму, то розраховується еквівалентна спектральна густина струму  $S_{I_{ВХ ЕК}}(f)$ , А<sup>2</sup>/Гц. Рівень шуму перераховується з виходу на вхід діленням спектральної густини вихідної напруги  $S_{U_{ВИХ}}$  на квадрат модуля відповідної передавальної функції. Слід зауважити, що внутрішній опір реального генератора сигналу  $R_G$  повинен бути внесений в опис кола як окремий резистор. Якщо вказано цілочисловий параметр *<n>*, то на кожній *n*-й частоті в діапазоні аналізу буде розраховуватися не тільки спектральна густина сумарного шуму, але і внесок в неї кожного шумового джерела. Якщо параметр *<n>* не зазначений, то цей розрахунок не проводиться.

У діалоговому вікні задання параметрів режиму AC Sweep є два розділи (рис. 2.1).

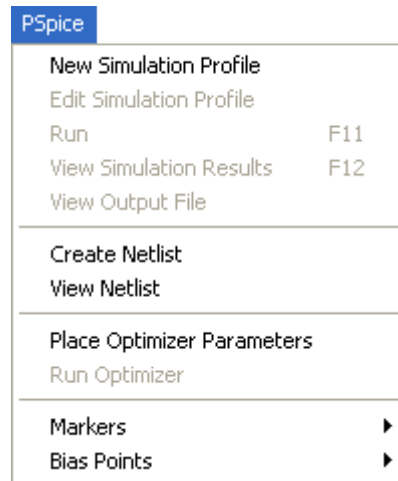


Рисунок 2.1 – Діалогове вікно задання параметрів режиму AC Sweep

У першому задаються параметри директиви зміни частоти.

У діалоговому вікні в розділі AC Sweep Type визначається характер зміни частоти:

- *Linear* – лінійна шкала;
- *Octave* – зміна частоти октавами;
- *Decade* – зміна частоти декадами.

В розділі *Sweep Parameters* задаються параметри діапазону частот:

- *Total Pts*, *Pts / Decade*, *Pts / Octave* – загальна кількість точок при виборі лінійного масштабу або кількість точок за частотою на одну декаду або октаву;

- *Start Freq* – початкова частота;
- *End Freq* – кінцева частота.

В розділі *Noise Analysis* встановлюються параметри розрахунку спектральної густини внутрішнього шуму:

- *Noise Enabled* – включення режиму розрахунку рівня шуму;
- *Output Voltage* – вихідна напруга;
- *I / V Source* – ім'я вхідного джерела напруги або струму;
- *Interval* – інтервал п розрахунку парціальних рівнів шуму.

Розрахунок характеристик в частотній області виконується після визначення режиму за постійним струмом і лінеаризації нелінійних компонентів (це робиться автоматично, ніяких додаткових директив не потрібно). Всі незалежні джерела напруги  $V$  і струму  $I$ , для яких задаються параметри АС-сигналів (амплітуди і фази), є вхідними впливними факторами. При проведенні АС-аналізу решта специфікації цих джерел,

зокрема й параметри синусоподібного сигналу  $SIN$ , не беруться до уваги, вони враховуються при аналізі перехідних процесів. Результати розрахунку комплексних амплітуд вузлових напруг і струмів віток виводяться за директивами *.PRINT*, *.PLOT* або *.PROBE*.

Наведемо приклади текстового задання директив:

```
.AC DEC 20 10k 100MEG
```

```
.NOISE V (5) VIN
```

```
.NOISE V (101) VSRC 20
```

```
.NOISE V (4,5) ISRC
```

Результати розрахунку рівнів шуму виводяться у вихідний файл *\*.OUT* за директивою *.PRINT* або *.PLOT*: *.PRINT NOISE <вихідна змінна> .PLOT NOISE <вихідна змінна>*.

Як вихідні змінні при розрахунку рівнів шуму використовуються такі імена:

- *INOISE, DB (INOISE)* – корінь ( $S_{ВХ}(f)$ ) у відносних одиницях і децибелах;

- *ONoise, DB (ONoise)* – корінь ( $S_{УВих}(f)$ ) у відносних одиницях і децибелах.

Наприклад:

```
.PRINT NOISE INoise ONoise DB (INOISE)
```

```
.PRINT NOISE INoise ONoise
```

```
.PRINT NOISE ONoise DB (ONoise)
```

Графіки спектральних густин можна побудувати за допомогою програми Probe.

Причому крім сумарних спектральних густин *INOISE* і *ONoise* доступні і парціальні спектральні густини напруги вихідного шуму, зумовлені окремими джерелами шуму.

Для їх побудови в програмі Probe використовуються спеціальні позначення.

За результатами розрахунку спектральної густини внутрішнього шуму легко обчислюється диференціальний коефіцієнт шуму лінійного чотиріполюсника при підключенні до входу джерела напруги, зображеного на рис. 2.2, а).

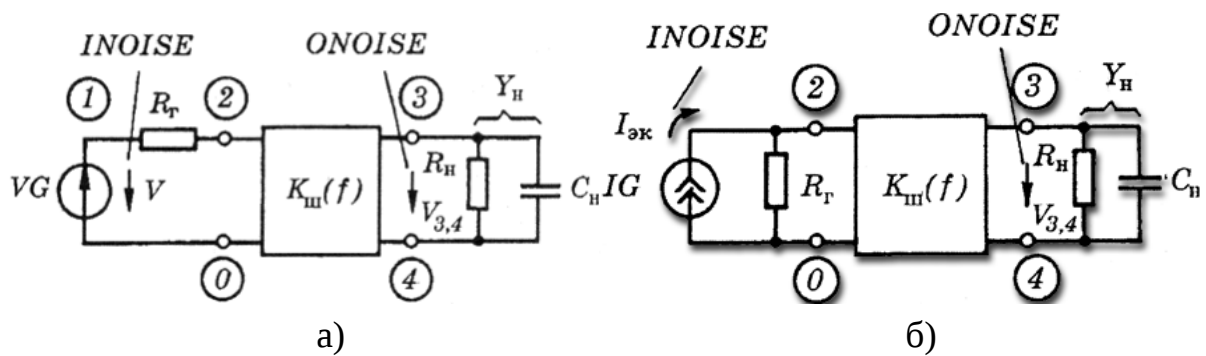


Рисунок 2.2 – Вимірювання диференціального коефіцієнта шуму лінійного чотириполіусника при підключенні до входу джерела напруги (а) або струму (б)

Як відомо, диференціальний коефіцієнт шуму дорівнює

$$K_{ш} = S_{U_{вв.ел}}(f) / S_{UГ} ,$$

де  $S_{U_{вв.ел}}(f)$  – спектральна густина напруги, обумовленої шумом опору генератора  $R_G$  і внутрішнім шумом чотириполіусника та прикладеної до його входу;  $S_{U_{вв.ел}}(f) = INOISE^2$ ;  $S_{UГ} = 4kTR_G$  – спектральна густина напруги шуму опору генератора;  $k = 1,38 \cdot 10^{-23}$  Дж/°С – стала Больцмана;  $T_0 = 300$  К – номінальна абсолютна температура.

Формула для розрахунку диференціального коефіцієнта шуму набуває, таким чином, вигляду:

$$K_{ш}(f) = INOISE^2 / 1,656 \cdot 10^{-20} R_G ,$$

де опір  $R_G$  вказується в омах.

Наведемо фрагмент розрахунку коефіцієнта шуму чотириполіусника, зображеного на рис. 2.2, а:

```
.TEMP 80
VG 1 0 .AC 1
RG 1 2 75
RLOAD 3 4 500
CLOAD 3 4 30pF
{опис чотириполіусника}
.AC LIN 21 0 1000HZ
.NOISE V(3,4) VG
```

### *.PRINT INOISE ONOISE*

Аналогічно вимірюється диференціальний коефіцієнт шуму лінійного чотириполюсника при підключенні до входу джерела струму (рис. 2.2, б).

При вимірюванні коефіцієнта шуму опору генератора  $R_G$  повинна бути прописана номінальна температура  $T_0$  чотириполюсника – його фізична температура, а опору навантаження  $R_H$  – температура абсолютного нуля, оскільки його шуми зазвичай беруться до уваги при розрахунку коефіцієнта шуму подальшого каскаду. Призначення резисторам індивідуальних значень температури виконується за допомогою параметра  $T\_ABS$ .

В деяких задачах можуть знадобитися незалежні джерела шуму. Вони можуть бути наведені у вигляді залежних джерел, керованих струмом шумового резистора (у зв'язку з особливостями програми PSpice допускається керувати струмом джерела ЕРС, увімкненого послідовно з шумовим резистором) [6].

### **2.2.2 Load Bias Point – завантаження даних режиму за постійним струмом**

Зчитування з файлу вузлових потенціалів відбувається за директивою *.LOADBIAS* <«Ім'я файлу»>.

З файлу, створеного в попередньому сеансі роботи з програмою PSpice, зчитуються вузлові потенціали за постійним струмом. Для виконання розрахунку перехідних процесів за директивою *.TRAN* із заданими початковими умовами цей файл можна попередньо відредагувати і замінити директиву *.NODESET* на *.IC*. Для передачі змісту файлу вузлових потенціалів, зазначеного в директиві *.LOADBIAS*, у вихідний файл результатів (з розширенням імені *\*.OUT*) необхідно в директиві *.OPTIONS* вказати параметр *EXPAND*.

Наведемо приклади:

```
.LOADBIAS «DC_POINT.NOD»
```

```
.LOADBIAS «D: \ PSPICE \ JOB \ TR_DATA.TRN»
```

### **2.2.3 Save Bias Point – збереження даних режиму за постійним струмом**

Запис в файл вузлових потенціалів здійснюється за директивою *.SAVEBIAS* <«Ім'я файлу»> <[*OP*] [*DC*] [*TRAN*]> [*NOSUBCKT*] + [*Time* = <значення>] [*REPEAT*] [*TEMP* = <значення>] [*STEP* = <значення>] + [*MCRUN* = <значення>] [*DC* = <значення>] [*DC1* = <значення>] + [*DC2* = <значення>].

В файл з вказаним ім'ям записуються значення вузлових потенціалів схеми для аналізу зазначеного виду *OP*, *DC* або *TRAN*. Для кожного виду аналізу в одному завданні повинна бути окрема директива.

Параметр *NOSUBCKT* забороняє запис в файл вузлових потенціалів внутрішніх вузлів макромоделей.

Параметри *TIME* = <значення> [*REPEAT*] визначають моменти часу, в які запам'ятовуються вузлові потенціали схеми при розрахунку перехідних процесів (режим *TRAN*). Якщо параметр *REPEAT* не вказаний, то вузлові потенціали запам'ятовуються в перший момент часу, більший за вказане значення (з точністю до кроку інтегрування). За наявності параметра *REPEAT* параметр <значення> дорівнює інтервалу часу, з яким вузлові потенціали періодично запам'ятовуються, при цьому в файлі зберігається тільки останній запис.

Параметр *TEMP* = <значення> задає температуру, для якої зберігаються вузлові потенціали при варіації температури, а необов'язковий параметр *STEP* = <значення> задає інтервал температур, через який оновлюється запис в файл.

Параметр *MCRUN* = <значення> визначає кількість варіантів розрахунків, через які оновлюється запис в файл при статистичному аналізі або розрахунку за методом найгіршого випадку.

Параметри *DC* = <значення>, *DC1* = <значення> і *DC2* = <значення> задають значення параметрів, що варіюються за директивою *.DC*, для яких виконується запис в файл. Параметр *DC* = <значення> використовується тільки за варіації однієї змінної, а *DC1* і *DC2* – за варіації двох змінних (відповідно першої та другої).

Утворений текстовий файл має такий формат. Спочатку на одному або більше рядках у вигляді коментарів зазначаються імена схеми, заголовки, дата створення, витрати часу на розрахунок, тип аналізу, температура тощо. Далі поміщається єдина директива *.NODESET*, що містить повну інформацію про вузлові потенціали схеми. Тому після завантаження файлу за директивою *.LOADBIAS* ці потенціали будуть встановлені автоматично.



Одне з типових застосувань директив *.SAVEBIAS*, *.LOADBIAS* – дослідження великих схем, для яких розрахунок режиму за постійним струмом займає значний час. Тоді після розрахунку режиму за постійним струмом він може бути збережений за директивою *.SAVEBIAS* і надалі перед виконанням інших видів аналізу (перехідних процесів, частотних характеристик, статистичного аналізу тощо) вузлові потенціали в цьому режимі можуть бути завантажені за директивою *.LOADBIAS*. Єдине обмеження: директиви *.SAVEBIAS* і *.LOADBIAS* повинні поміщатися в різних завданнях на моделювання. Слід зауважити, що при модифікації схеми користувач може для поліпшення збіжності вручну відредагувати файл режиму, додаючи або виключаючи з директиви *.NODESET* ряд вузлів схеми і коригуючи їх потенціали.

Наведемо приклади:

```
.SAVEBIAS «DC_POINT.NOD» OP
```

```
.SAVEBIAS «TR_DATA.TRN» TRAN NOSUBCKT TIME = 10us
```

```
.SAVEBIAS «DC.BSP» DC MCRUN = 3 DC1 = 2.5 DC2 = 10
```

При роботі з керуючою оболонкою Schematics ім'я файлу вузлових потенціалів і інші параметри вказується в діалоговому вікні, що відкривається після натискання на кнопку Save Bias Point в меню вибору директив моделювання [6].

## **2.2.4 DC Sweep – варіація параметрів при розрахунку режиму за постійним струмом**

Розрахунок режиму за постійним струмом здійснюється при варіації одного або декількох джерел постійної напруги або струму, температури, параметрів моделей компонентів схеми і глобальних параметрів за директивами:

- *.DC [LIN] <ім'я 1-ї змінної> <початкове значення> + <кінцеве значення> <приріст> + [<ім'я 2-ї змінної> <початкове значення> + <кінцеве значення> <приріст>];*

- *.DC [OCT] [DEC] <ім'я 1-ї змінної> <початкове значення> + <кінцеве значення> <кількість точок> + [<ім'я 2-ї змінної> <початкове значення> + <кінцеве значення> <кількість точок>];*

- *.DC <ім'я 1-ї змінної> LIST <значення> + [<ім'я 2-ї змінної> LIST <значення>].*

Режим за постійним струмом розраховується для декількох значень варійованих змінних, як такі можуть вважатися:

- імена незалежних джерел напруги або струму;
- параметри моделей компонентів (вказується тип компонента, ім'я моделі і в круглих дужках ім'я варійованого параметра);
- температура (як її ім'я вказується *TEMP*);
- глобальні параметри (вказується ключове слово *PARAM* і далі за ним ім'я варійованого глобального параметра, визначеного раніше).

Характер зміни змінних задається ключовими словами:

- *LIN* – лінійний масштаб (ключове слово *LIN* можна не вказувати);
- *DEC*, *OCT* – логарифмічний масштаб декадами або октавами;
- *LIST* – список значень.

Якщо вказані специфікації двох варійованих параметрів, то перший параметр змінюється в заданих межах для кожного значення другого параметра.

Такий вкладений цикл зручний, зокрема для побудови статичних характеристик напівпровідникових приладів.

Максимальна кількість ітерацій при переході до подальшого варіанта за замовчуванням встановлюється рівною досить малій величині *ITL2 = 20*. Тому в схемах, чутливих до варіацій параметрів, можуть виникнути проблеми зі збіжністю. У цих випадках рекомендується за директивою *.OPTIONS* збільшити значення *ITL2*.

Наведемо приклади:

```
.DC VIN 0.5 5.0 0.25
```

```
.DC LIN VDS 0 10 .5 VGS 0 5 1
```



## КОНТРОЛЬНІ ЗАПИТАННЯ

---

1. Яка структура текстового завдання на моделювання?
2. Які директиви моделювання OrCAD існують?
3. За що відповідає директива AC Sweep?
4. За що відповідає директива Load Bias Point?
5. За що відповідає директива Save Bias Point?
6. За що відповідає директива DC Sweep?



---

## **3 МОДЕЛІ КОМПОНЕНТІВ АНАЛОГОВОЇ ЕЛЕКТРОНІКИ**

### **3.1 Базовий набір моделей елементів**

Існує ряд моделей, на основі яких може бути побудована будь-яка електрична модель приладу. Ці моделі отримали назву базового набору елементів моделей [10].

Ідеальні активні елементи – ідеальні джерела напруги або струму вводяться для характеристики реальних джерел електричної енергії, у яких неелектричні види енергії (хімічна, механічна) перетворюються на електричну. Залежно від вигляду ВАХ і можливості керування вихідними параметрами розрізняють залежні й незалежні джерела напруги й струму.

Незалежне ідеальне джерело напруги являє собою двополюсний елемент, напруга на якому, можливо, змінюється в часі за деяким законом  $e(t)$ , але не залежить від величини струму, що проходить через джерело. Еквівалентна схема й ВАХ джерела напруги  $u(i)$  та його еквівалентна схема

заміщення зображені на рис. 3.1, а).

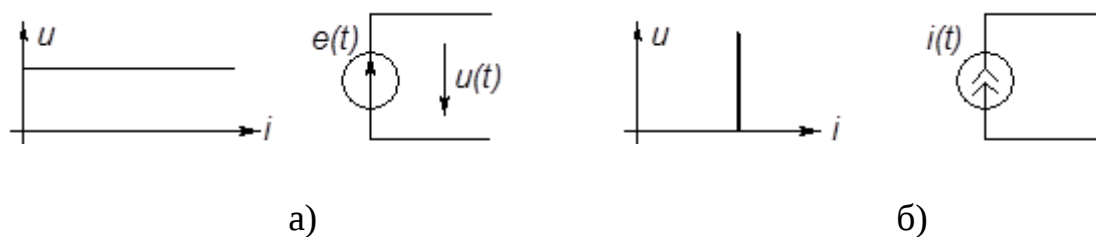


Рисунок 3.1 – Еквівалентні схеми й ВАХ джерел: а – напруги; б – струму

У режимі короткого замикання величина струму джерела прямує до нескінченності, джерело віддає нескінченно велику потужність у навантаження.

Незалежне ідеальне джерело струму являє собою двополюсний елемент, вихідний струм якого змінюється, можливо, за деяким законом у часі незалежно від напруги на елементі. Джерело віддає в навантаження нескінченно велику потужність при холостому ході.

Еквівалентна схема й ВАХ джерела струму зображені на рис. 3.1, б).

Залежні ідеальні джерела напруги або струму являють собою триполюсні або чотириполюсні елементи (необоротні), які ставлять у залежність від вхідного сигналу (напруги або струму) вихідний струм або напругу. Якщо керування залежними джерелами здійснюється струмом, то його вхід являє собою коротке замикання, якщо напругою – холостий хід. Залежні ідеальні джерела живлення наведені на рис. 3.2.

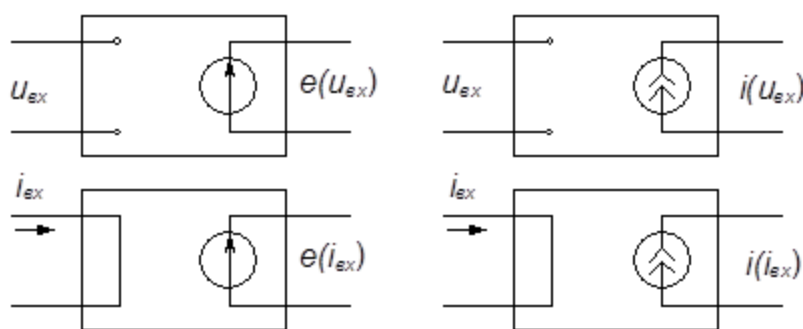


Рисунок 3.2 – Залежні ідеальні джерела живлення

До базового набору моделей елементів також входять моделі ідеальних пасивних елементів  $R$ ,  $L$ ,  $C$ , які будуть розглянуті у подальших пунктах [1].

### 3.2 Опис моделей резисторів

В загальному випадку опис моделі елемента – це сукупність термінів та визначень, необхідних для виконання автоматизованого проектування. В мові Spice кожному елементу принципової схеми відповідає одне або два речення.

В реченні вказують такі відомості:

1. Ідентифікатор виду елемента або ж назву математичної моделі, що використовується для даного елемента;
2. Порядковий номер елемента у схемі;
3. Чисельні значення параметрів.

Ідеальний лінійний резистор характеризується опором  $R$  і провідністю  $G$ . Розмірності величин:  $[R]$  – Ом,  $[G]$  – См.

Нелінійний опір характеризується ВАХ нелінійного резистора, зображеною на рис. 3.3.

Фазові спотворення в елементі відсутні.

ВАХ реального резистора на постійному струмі не відрізняється від ідеальної. При створенні високочастотної моделі необхідно враховувати особливості виготовлення. За рахунок наявності індуктивності виводів і резистивного шару опір резисторів при високій частоті стає комплексним, набуває індуктивного характеру:

$$X_R = R + j\omega L.$$

З підвищенням частоти проявляються також ємнісні властивості резистора – в основному, ємність між його виводами. Виходячи з цього, еквівалентні схеми резистора (повна та спрощена) із зосередженими параметрами мають вигляд, наведений на рис. 3.4.

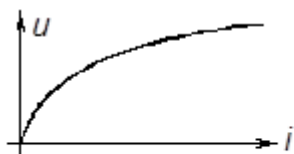


Рисунок 3.3 – ВАХ нелінійного резистора

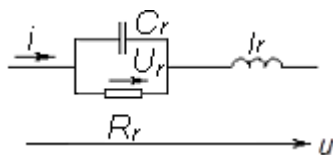
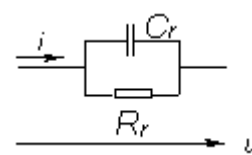


Рисунок 3.4 – Еквівалентні схеми резистора (повна та спрощена)



Загалом параметри резистора мають бути розподіленими, однак це значно ускладнює аналіз і використовується лише при роботі в НВЧ-діапазоні і при розрахунку елементів у гібридному виконанні. Наведену еквівалентну схему можна описати системою 2-х диференціальних рівнянь:

$$\begin{cases} u = u_r + L_r \frac{di}{dt}; \\ i = \frac{u_r}{R_r} + C_r \frac{du_r}{dt}, \end{cases}$$

де  $C_r$ ,  $L_r$  – величини паразитних параметрів;  $u_r$  – напруга на активній складовій резистора.

Залежно від співвідношення параметрів еквівалентна схема резистора може бути спрощена. Для високоомних резисторів, у яких  $R \gg \sqrt{\frac{L_r}{C_r}}$ , тобто активний опір значно більший хвильового, можна не враховувати його індуктивну складову (див. рис. 3.4).

Повний опір синусоподібного сигналу можна знайти з розрахунку:

$$Z_R = \frac{R_r \cdot \frac{1}{j\omega C_r}}{R_r + \frac{1}{j\omega C_r}} = \frac{R_r}{1 + j\omega C_r R_r} = \frac{R_r}{1 + \omega^2 R_r^2 C_r^2} - j \cdot \frac{\omega C_r R_r^2}{1 + \omega^2 R_r^2 C_r^2}.$$

Тут перший доданок – активна складова повного опору, друга —

реактивна складова. При  $\omega \rightarrow 0$ ,  $Z_r \rightarrow R_r$ , при  $\omega \rightarrow \infty$ ,  $Z_r \rightarrow \frac{1}{j\omega C_r}$ . Для низькоомних резисторів  $R \ll \sqrt{\frac{L_r}{C_r}}$ , можна знехтувати ємнісною складовою комплексного опору.

Зазначену особливість необхідно враховувати при проектуванні схем, особливо в НВЧ-діапазоні [1].

В мові Spice використовуються дві моделі резисторів – ідеальна та реальна.

В ідеальній моделі резистор має тільки значення опору, а в реальній враховується температурний коефіцієнт, що розраховується за формулою:

$$\langle Onip \rangle \times R \left[ 1 - TC1(t - T0) - TC2(T - T0)^2 \right],$$

де  $\langle Onip \rangle$  – номінал, що вказаний в схемі;  $R$  – масштабний множник опору;  $TC1$ ,  $TC2$  – лінійний та квадратичний температурний коефіцієнти опору, відповідно;  $T0$  – номінал температури навколишнього середовища ( $\sim 27^\circ\text{C}$ );  $T$  – поточна температура, для якої визначається опір.

Для задання реальної моделі використовується друге речення опису, що розпочинається з директиви *.MODEL*, після якої вказується ім'я моделі, її тип та значення параметрів. Наприклад:

```
R12 3 2 RTEMP 20K
```

```
.Model RTEMP RES (R=1 TC1=0.3 TC2=0.002)
```

У даному випадку 3 та 2 – номери вузлів, до яких у схемі під'єднано опір, *RTEMP* – назва моделі резистора.

Якщо ж потрібно врахувати частотні властивості резистора, то його схему потрібно замінити еквівалентною, тобто врахувати ємності між виводами та наявність індуктивності виводів [11].

### 3.3 Опис моделей індуктивностей

Ідеальний індуктивний елемент характеризується потокозчепленням, що виникає при протіканні струму через деякий контур, причому вважається, що активний опір контуру дорівнює 0. Відношення потокозчеплення до струму називається індуктивністю  $L$ :

$$L = \frac{\Psi}{i}.$$

Розмірності величин:  $[i]=A$ ,  $[\Psi]=Wb=V \cdot c$ ,  $[L]=Wb/A=(V \cdot c)/A=Гн$ .

Енергія, накопичена в полі:

$$W_M = \frac{\Psi I}{2}.$$

Закон електромагнітної індукції пов'язує швидкість зміни магнітного потоку й ЕРС, що наводиться в контурі:

$$e = \frac{d\Psi}{dt}.$$

Якщо  $L=const$ , то  $e = L \frac{di}{dt}$ .

Закон можна переписати в іншому вигляді:

$$i(t) = i(0) + \frac{1}{L} \int_0^t e(t) dt.$$

Реактивний опір контуру на змінному струмі залежить від частоти синусоподібного сигналу:  $|X_L| = \omega L$ . Фазове зміщення між струмом і напругою –  $90^\circ$ . Струм відстає від напруги на зазначений кут. Якщо на індуктивність подана напруга  $u = U_m \cos \omega t$ , то струм у колі:

$$i = \frac{U_m}{\omega L} \sin \omega t.$$

Потужність, що циркулює в колі:

$$P = u \cdot i = \frac{U_m^2}{\omega L} \sin \omega t \cos \omega t = \frac{U_m^2}{2\omega L} \sin 2\omega t.$$

Таким чином, потужність не споживається навантаженням, а здійснюється лише періодичний обмін енергією, що проілюстровано часовими діаграмами гармонійного сигналу на індуктивності (рис. 3.5).



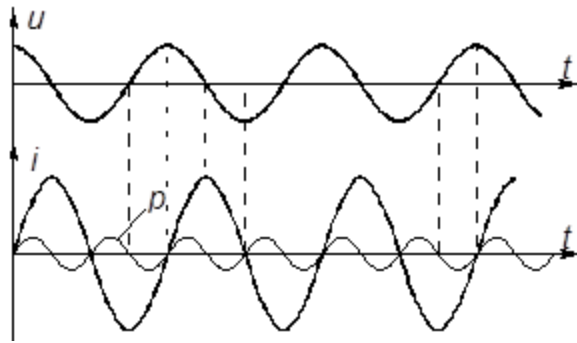


Рисунок 3.5 – Часові діаграми гармонічного сигналу на індуктивності

Індуктивність являє собою магнітопровід із зробленою на ньому обмоткою. Еквівалентна схема індуктивності з урахуванням омичного опору обмотки має вигляд, зображений на рис. 3.6, а).

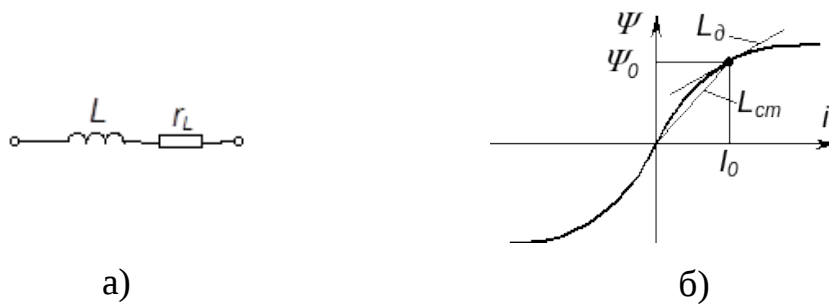


Рисунок 3.6 – Еквівалентна схема індуктивності (а) та характеристика індуктивності з осердям (б)

Повний опір реальної індуктивності на змінному струмі з частотою  $\omega$  дорівнює:

$$Z = r_L + j\omega L, \quad |Z| = \sqrt{r_L^2 + (\omega L)^2}.$$

Індуктивність, виконана на осерді з феромагнітного матеріалу, як правило, нелінійна при роботі в сильних полях. Її характеристика збігається з магнітною характеристикою феромагнетика й може бути зображена у вигляді характеристики індуктивності з осердям (рис. 3.6, б) [1].

Якщо на індуктивність подано постійний підмагнічувальний струм  $I_0$ ,

то можна виділити диференціальну індуктивність  $L_{\partial}$  і статичну індуктивність  $L_{cm}$ , обумовлені виразами:

$$L_{cm} = \frac{\Psi_0}{I_0}; \quad L_{\partial} = \left. \frac{d\Psi}{dI} \right|_{i=I_0}.$$

У мові Spice індуктивність, зазвичай, враховується як ідеальний елемент. Приклад опису такої індуктивності:  $L5 \ 7 \ 3 \ 20\mu H$ .

На високих частотах слід будувати еквівалентну схему, яка враховує омичний опір магнітопроводу [11].

### 3.4 Опис моделей конденсаторів

Ідеальна ємність накопичує енергію електростатичного поля.

Заряд ємності визначається як:

$$Q = Q_0 + \int_0^t i(t) dt.$$

Якщо ємність лінійна, то заряд  $Q = CU$ , де  $C$  – ємність конденсатора. Розмірність величин:  $[Q]$  – Кл,  $[U]$  – В,  $[C]$  – Кл/В = Ф.

Енергія, накопичена в ємності:

$$W_{\epsilon} = \frac{QU}{2}.$$

Для лінійної ємності можна також записати:

$$W_{\epsilon} = \frac{CU^2}{2}; \quad u = U_0 + \frac{1}{C} \int_0^t i dt.$$

Опір ємнісного кола змінного струму:

$$|X_c| = \frac{1}{\omega C}.$$

Напруга на ємності відстає від струму на  $90^\circ$ . Якщо струм у колі:

$$i_c = I_m \cos \omega t,$$

то напруга:

$$u_c = \frac{I_m}{\omega C} \sin \omega t.$$

Потужність у колі:

$$P = u_c \cdot i_c = \frac{I_m^2}{2\omega C} \sin 2\omega t.$$

Тобто, як і у випадку індуктивності, споживання потужності не відбувається, здійснюється лише обмін енергією між джерелом і навантаженням, що проілюстровано часовими діаграмами гармонічного сигналу на ємності (рис. 3.6).

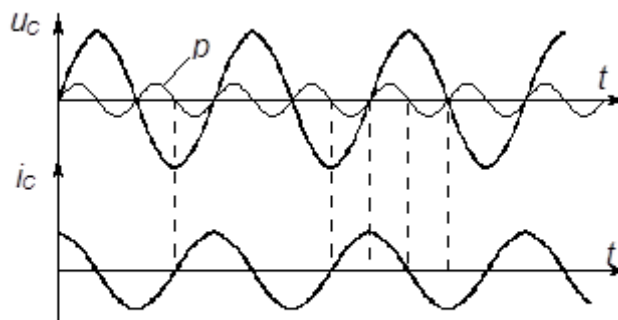


Рисунок 3.7 – Часові діаграми гармонічного сигналу на ємності

В електроніці конденсатори виконують у вигляді струмопровідних обкладок, розділених діелектриком. Величина ємності залежить від площі обкладок  $S$  і  $\varepsilon$  діелектрика. Розрізняють конденсатори з газоподібним, рідким, твердим, оксидним діелектриком. На високих частотах необхідно враховувати втрати в діелектрику, струми витoku, індуктивний опір обкладок і виводів.

Еквівалентна схема конденсатора на високих частотах має вигляд, зображений на рис. 3.8, а).

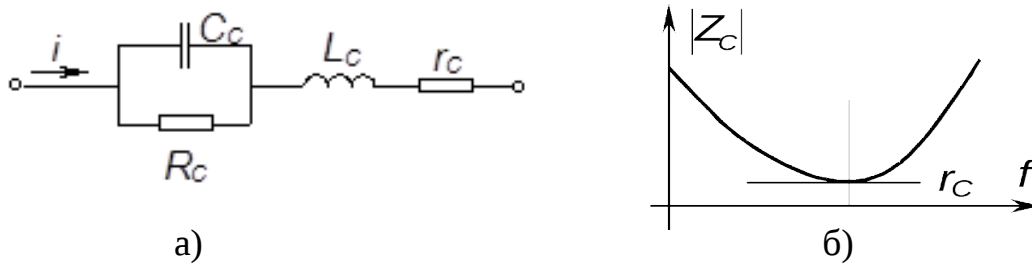


Рисунок 3.8 – Еквівалентна схема (а) та залежність опору (б) від частоти конденсатора на високих частотах

Повний опір конденсатора залежить від частоти й визначається за формулою:

$$|Z_c| = \sqrt{r_c^2 + \left(\frac{1}{\omega C_c} - \omega L_c\right)^2}$$

Тут зроблене припущення про те, що  $R_c \gg r_c$ .

Це справедливо на високих частотах практично завжди.

Опір втрат  $R_c$  враховується, як правило, лише в прецизійних схемах і при розрахунку повільних процесів.

Залежність опору  $Z_c(f)$  від частоти конденсатора на високих частотах відповідно до наведеного співвідношення має вигляд як на рис. 3.8, б).

Таким чином, на частотах, вищих резонансної частоти  $f = \frac{1}{2\pi\sqrt{L_c C_c}}$ , опір конденсатора набуває індуктивного характеру.

Слід зазначити, що для електролітичних конденсаторів величина ємності не постійна, а також залежить від частоти роботи, що пояснюється процесами поляризації діелектрика.

Можна запропонувати апроксимацію залежності  $C_c(f)$  у вигляді:

$$C_c = \frac{C_0}{1 + \left(\frac{f}{f_0}\right)^\alpha}$$

де  $C_0$  – номінальна ємність конденсатора;  $f_0$  – частота, на якій ємність зменшується вдвічі;  $\alpha$  – деякий коефіцієнт, що враховує швидкість спаду

ємності [1].

Конденсатор може використовуватись як ідеальний елемент або як вбудована модель. Приклад опису ідеального конденсатора: *C2 8 5 56 pF*.

В реальній моделі конденсатора враховуються температурні коефіцієнти ємності, а також залежності ємності від прикладеної напруги, при цьому ємність виражається як:

$$\langle \text{ємність} \rangle \cdot C [1 + VC1 \cdot u + VC2 \cdot u^2] \cdot [1 + TC1(T - T0) + TC2(T - T0)^2],$$

де  $\langle \text{ємність} \rangle$  – номінал, що вказаний в схемі;  $C$  – масштабний множник ємності;  $TC1$ ,  $TC2$  – лінійний та квадратичний температурний коефіцієнти ємності, відповідно;  $VC1$ ,  $VC2$  – лінійний та квадратичний коефіцієнти напруги, відповідно;  $T$  – поточна температура, для якої визначається ємність.

Залежність ємності від напруги враховується тільки при розрахунку перехідних процесів.

При розрахунках частотних характеристик  $VC1=VC2=0$ .

Приклад опису реальної моделі конденсатора:

*C2 8 5 Creal 56 pF*

*.Model Creal CAP (C=1 VC1=0.1 VC2=0.01 TC1=0.15 TC=0.05)*

У даному випадку 8 та 5 – номери вузлів, до яких у схемі під'єднано ємність, *Creal* – назва моделі ємності.

На високих частотах необхідно враховувати втрати в діелектрику, обкладинках, струми втрат, а також індуктивний опір обкладинок та виводів, тому конденсатор замінюють його еквівалентною схемою [11].

На С. 42 обкладинки !

### 3.5 Опис моделей трансформаторів

Трансформатор (рис. 3.9) являє собою магнітопровід з двома або більше обмотками.

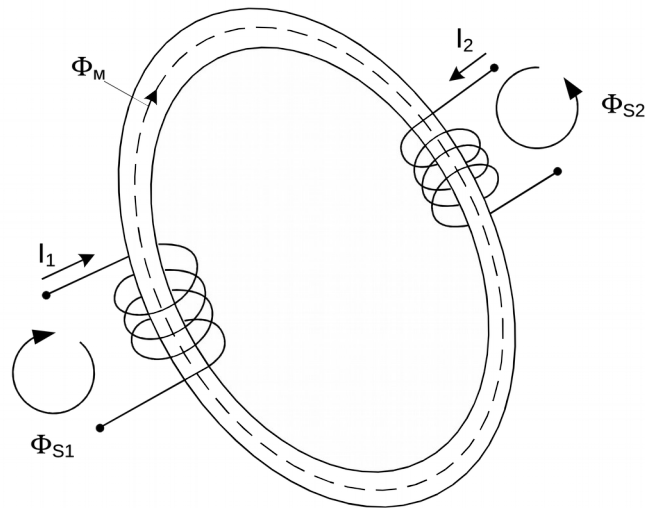


Рисунок 3.9 – Трансформатор

По магнітопроводу замикається магнітний потік  $\Phi_M$ , що забезпечує магнітний зв'язок між обмотками. Існують потоки розсіювання  $\Phi_{S1}$  і  $\Phi_{S2}$ , створені струмами  $i_1$  і  $i_2$ , що не замикаються по магнітопроводу й не є загальними для обмоток. Обмотки мають омичні опори  $r_1$  і  $r_2$ , а також інші паразитні параметри, наприклад, міжвиткові й міжобмоткові ємності.

Ідеальний трансформатор показаний елементом, у якому відсутні потоки розсіювання, активні опори обмоток та паразитні ємності. Індуктивність намагнічування дорівнює нескінченності, тобто струм намагнічування дорівнює 0. При цьому обмотки пронизують однакові потоки й за законом електромагнітної індукції для обох обмоток можна записати, що:

$$u_1 = e_1 = \frac{d\Psi_M}{dt} = w_1 \frac{d\Phi_M}{dt}; \quad u_2 = e_2 = \frac{d\psi_M}{dt} = w_2 \frac{d\Phi_M}{dt}; \quad e_2 = e_1 \frac{w_2}{w_1} = e_1 n.$$

Тут  $n = w_2/w_1$  – коефіцієнт трансформації трансформатора.

Із закону збереження енергії випливає, що  $i_1 \cdot e_1 = i_2 \cdot e_2$ , тобто  $i_2 = \frac{i_1}{n}$ .

Цей же висновок можна отримати, використовуючи закон повного струму з урахуванням тієї обставини, що  $i_{\bar{l}} = 0$ :

$$\sum i w = 0; \quad i_1 \cdot w_1 = i_2 \cdot w_2, \quad i_2 = \frac{i_1}{n}.$$

При зазначених припущеннях трансформатор можна подати еквівалентною схемою, що зображена на рис. 3.10, а).

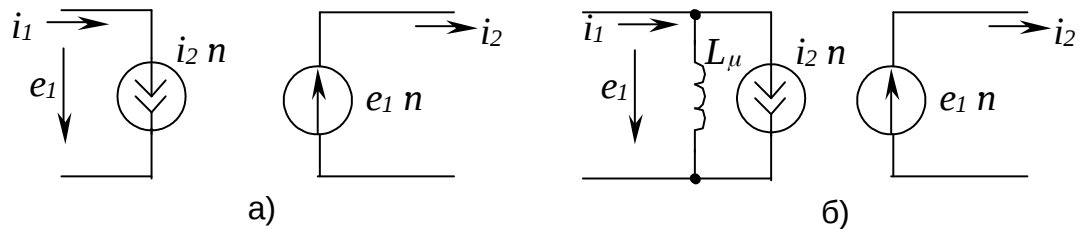


Рисунок 3.10 – Еквівалентні схеми трансформатора

Модель вкрай проста, однак застосовується досить рідко, оскільки не дозволяє врахувати принципово важливі для трансформатора явища – накопичення енергії в магнітопроводі й необхідність відновлення магнітного стану осердя. Для її врахування трансформатор наводять ідеальним елементом, у якому відсутні потоки розсіювання, але є деяка скінченна величина індуктивності намагнічування  $L_{\mu}$ . Остання визначає струм намагнічування, що протікає по первинній обмотці, і приводить до еквівалентної схеми у вигляді рис. 3.10, б).

У ряді випадків, особливо при розрахунку досить простих кіл, зручно «привести» трансформатор до однієї зі сторін: первинної або вторинної, наводячи процес передачі енергії через магнітопровід безпосереднім підключенням навантаження до кола джерела енергії. Якщо, наприклад, трансформатор навантажений на активний опір  $R$ , то величина струму навантаження в первинному колі визначається:

$$i_{12} = i_2 \cdot n = \frac{u_2}{R_2} \cdot n.$$

Тоді еквівалентний опір навантаження, підключеного до первинної обмотки, дорівнює:

$$R_2' = \frac{u_1}{i_{12}} = \frac{u_1 \cdot R_2}{u_2 \cdot n} = \frac{R_2}{n^2}.$$

Нескладно показати, що індуктивність, підключена як навантаження, приводиться до первинної обмотки у вигляді:

$$L_2' = \frac{L_2}{n^2}.$$

Ємність – відповідно до співвідношення:

$$C_2' = C_2 \cdot n^2.$$

Таким чином, еквівалентна схема «приведеного» до первинної обмотки трансформатора має вигляд, наведений на рис. 3.11.

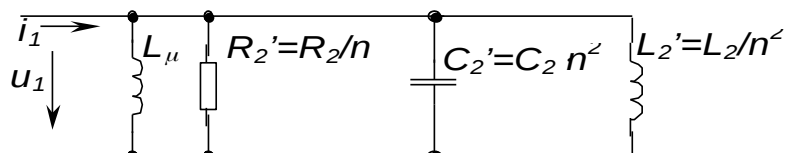


Рисунок 3.11 – Еквівалентна схема трансформатора, «приведеного» до первинної обмотки

Виконання трансформатора із осердям з магнітопроводу у вигляді феромагнетика призводить до нелінійної залежності  $L(i)$ .

Ідеальна еквівалентна схема трансформатора з врахуванням  $L_\mu$  дозволяє врахувати найважливішу його особливість – необхідність відновлення вихідного магнітного стану осердя.

Так, з врахуванням наведеної вище еквівалентної схеми, можна показати, що подача імпульсної напруги на трансформатор згідно зі схемою на рис. 3.12, а), призведе до того, що на вторинній обмотці буде формуватися імпульс напруги вигляду як на рис. 3.12, б).



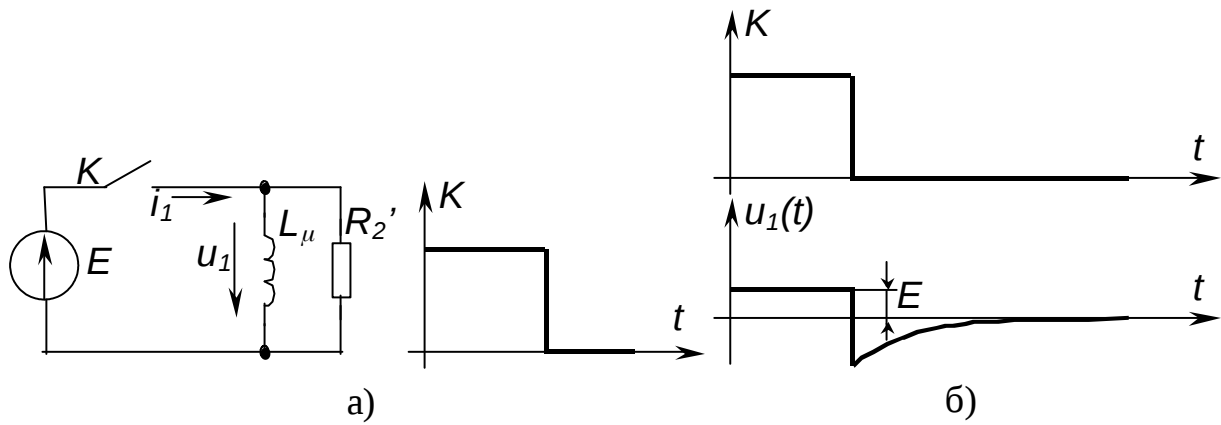


Рисунок 3.12 – Трансформатор: а – схема; б – формування імпульсу напруги

Дійсно, процес перемагнічування осердя в період часу  $0 \leq t \leq t_u$  описується співвідношенням:

$$\frac{d\psi}{dt} = E.$$

Після завершення накопичення енергії (при  $t=t_u$ ) струм  $i_\mu$  замикається по колу  $L_\mu, R_2$ . При цьому, як і раніше:

$$\frac{d\psi}{dt} = u_1(t),$$

причому

$$u_1(t) = -R_2' \cdot i_\mu(t).$$

Якщо робоча точка при перемагнічуванні повернулася у вихідний стан, то цей факт означає, що:

$$\Delta B_1 = \Delta B_2,$$

тобто

$$\Delta\psi_1 = \Delta\psi_2, \quad \int_0^{t_u} Edt = - \int_{t_u}^{\infty} u_1(t) dt.$$

Остання умова формулюється як рівність вольт-секундних площ на обмотці при її перемагнічуванні. Причому, рівність виконується незалежно від форми напруги на обмотці.

Таким чином, при розрахунку трансформаторних кіл необхідно враховувати час відновлення магнітного стану й формувати спеціальними колами (колами відновлення) форму напруги при відновленні. У наведеному вище випадку, при розмиканні ключа  $K$  і за відсутності навантаження, напруга на ключі може прямувати до нескінченності, що призведе до його пробую.

Наведені моделі трансформатора є низькочастотними, оскільки не враховують вплив індуктивностей розсіювання і ємностей між обмотками. Повна еквівалентна схема трансформатора, приведена до первинної обмотки, наведена на рис. 3.13.

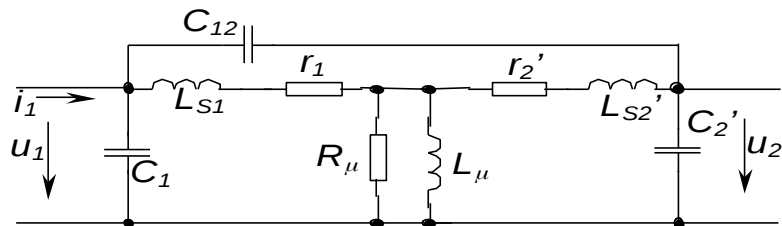


Рис. 1.23.

Рисунок 3.13 – Повна еквівалентна схема трансформатора, приведена до первинної обмотки

На даній схемі  $L_{S1}$ ,  $L_{S2}'$  – індуктивність розсіювання первинної обмотки й приведена до первинної обмотки індуктивність розсіювання вторинної обмотки;  $C_1$ ,  $C_2'$  – ємність первинної обмотки й приведена до первинної ємності ємність вторинної обмотки;  $C_{12}$  – ємність між первинною й вторинною обмотками,  $R_{\mu}$  – еквівалентний опір, що відображає втрати в магнітопроводі.

Розрахунок перехідних процесів у такій схемі, зрозуміло, досить складний, тому її в ряді випадків спрощують. Справа в тому, що індуктивність намагнічування має величину на 3–4 порядки більшу, ніж

індуктивність розсіювання. Тому сталі часу при розрахунку можуть відрізнятись на 4 і більше порядків. З іншого боку, залежно від того, є трансформатор понижувальним або підвищувальним, істотно змінюється приведена величина індуктивності розсіювання  $L_{S2}'$ :

$$L_{S2}' = \frac{L_{S2}}{n^2}.$$

Аналогічно, у підвищувальних трансформаторах найбільший вплив має приведена до первинної обмотки ємність вторинної  $C_2'$ :

$$C_2' = C_2 \cdot n^2.$$

Як правило, індуктивності розсіювання не чинять істотного впливу на процеси в колах намагнічування, тому є можливим об'єднати індуктивності розсіювання в одну. Аналогічно й для резисторів, що відображають омичні втрати в обмотках. Тому для трансформатора з підвищенням ( $n \gg 1$ ) еквівалентна схема трансформатора має вигляд, наведений на рис. 3.14, а), з пониженням – рис. 3.14, б) [1].

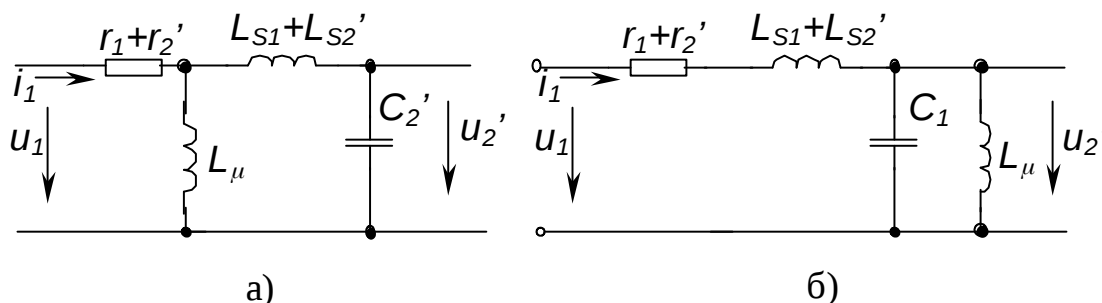


Рисунок 3.14 – Спрощені еквівалентні схеми трансформаторів: а – з підвищенням; б – з пониженням

Трансформатор у мові Spice описується  $K_{xxx} L_{ууу} L_{zzz} \dots$  <коефіцієнт зв'язку>, де  $xxx$ ,  $ууу$ ,  $zzz$  – будь-які символи.

Порядок перерахування імен індуктивностей  $L_{ууу}$ ,  $L_{zzz}$ , ... не має значення, знак взаємної індуктивності визначається порядком перерахування вузлів в описі кожної індуктивності. Параметром взаємної індуктивності є <коефіцієнт зв'язку>. Якщо в трансформаторі є кілька обмоток, то можна або визначити взаємні індуктивності для кожної попарної комбінації обмоток в окремих реченнях, або в одному реченні

вказати список всіх індуктивностей, що мають однаковий коефіцієнт зв'язку. Наприклад, високочастотний трансформатор з трьома обмотками (рис. 3.15) описується в такий спосіб:

$I1\ 1\ 0\ AC\ 1\ mA$

$L1\ 1\ 0\ 10\ \mu H$

$L2\ 2\ 3\ 10\ \mu H$

$L3\ 3\ 4\ 10\ \mu H$

$K12\ L1\ L2\ L3\ 0.8$

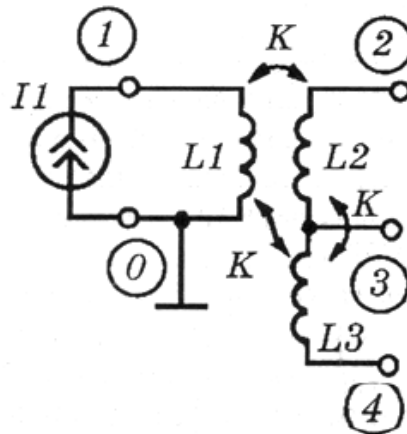


Рисунок 3.15 – Високочастотний трансформатор з трьома обмотками

Тут  $I1$  – джерело струму, комплексна амплітуда якого в режимі AC має значення 1 мА. Перший вузол в описах індуктивностей  $L1$ ,  $L2$ ,  $L3$  позначає початок обмотки.

Коефіцієнт зв'язку двох обмоток визначається виразом

$$\text{коефіцієнт зв'язку} = \frac{M_{ij}}{\sqrt{L_i L_j}},$$

де  $L_i$ ,  $L_j$  – індуктивності обмоток;  $M_{ij}$  – їхня взаємна індуктивність.

Напруга на котушці  $L_i$  з урахуванням взаємної індукції визначається виразом:

$$U = L_i \frac{dI_i}{dt} + M_{ij} \frac{dI_j}{dt} + M_{ik} \frac{dI_k}{dt} \dots$$

Магнітне осердя трансформатора описується  $K_{xx}\ L_{yy}\ L_{zz}\ \dots$   
 <коефіцієнт зв'язку> <ім'я моделі> + [<масштабний коефіцієнт>].

На одному осердді міститься одна або кілька обмоток з іменами  $L_{uu}$ ,  $L_{zz}$ , ... Всі обмотки мають однаковий <коефіцієнт зв'язку>. При описі кожної обмотки  $L_{uu}$ , згаданої в складі осерддя, змінюється зміст параметра <значення> – тепер він визначає не індуктивність, а число витків обмотки осерддя.

Наприклад, трансформатор з пермалоевим осердям (рис. 3.16) описується

$O L1 15 10 50; 50$  витків

$L2 10 0 150; 150$  витків

$K2 L1 L2 0.99 T1125V$

$.MODEL T1125V CORE (LEVEL=2 MS=334E3 ALPHA=2.5E-2 + A=4.05E3$   
 $K=166 C=0.05 AREA=0.064 PATH=2.25)$

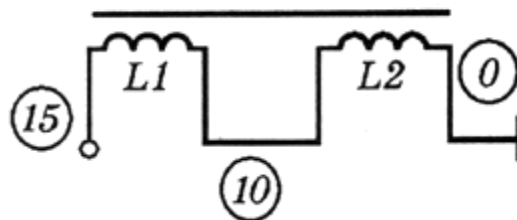


Рисунок 3.16 – Трансформатор з пермалоевим осердям

Параметр <масштабний коефіцієнт> змінює площу поперечного перерізу магнітопроводу (за замовчуванням дорівнює одиниці). Вона дорівнює добутку цього коефіцієнта на параметр моделі осерддя  $AREA$ .

Графік кривої намагніченості осерддя виводиться на екран за наявності директиви  $.PROBE$ . За допомогою директив  $.PRINT$  і  $.PLOT$  ці дані не виводяться.

У програмі PSpice використовується модель магнітного осерддя, запропонована Джилсом і Атертоном. Вона базується на відомих уявленнях про рух доменних меж магнітних матеріалів. З її допомогою вдається відобразити всі основні характеристики гістерезису, такі як крива початкової намагніченості, намагніченість насичення, коерцитивна сила, залишкова намагніченість і динамічні втрати на гістерезис.

Існують два різновиди цієї моделі –  $LEVEL = 1$  і  $LEVEL = 2$ , з яких модель  $LEVEL=1$  виявилася невдалою і її виключили з PSpice 8.0. Параметри моделі  $LEVEL=2$  наведені в табл. 3.1.

Таблиця 3.1 – Параметри моделі  $LEVEL=2$ 

Ім'я параметра	Параметр	Розмірність	Значення за замовчуванням
<i>LEVEL</i>	Індекс моделі	–	2
<i>A</i>	Параметр форми безгістерезисної кривої намагнічування	А/м	$10^3$
<i>AREA</i>	Площа поперечного перерізу магнітопроводу	см <sup>2</sup>	0,1
<i>C</i>	Стала пружного зсуву доменних меж	–	0,2
<i>GAP</i>	Ширина повітряного зазору	см	0
<i>K</i>	Стала рухливості доменів	А/м	500
<i>MS</i>	Намагніченість насичення	А/м	$10^6$
<i>PACK</i>	Коефіцієнт заповнення осердя	–	1
<i>PATH</i>	Середня довжина магнітної силової лінії	см	1

Параметри *AREA*, *PATH*, *GAP* і *PACK* визначаються геометричними розмірами осердя, інші – властивостями магнітного матеріалу [11].

### 3.6 Опис моделей лінії передачі

Лінія передачі описується у мові Spice двома способами: лінія передачі без втрат та з втратами.

1. Лінія передачі без втрат описується:  $Txxx <+вузол\ порту\ A> <-вузол\ порту\ A> <+вузол\ порту\ B> + <-вузол\ порту\ B> [ім'я\ моделі] + ZO=<значення> [TD=<значення>] + [F=<значення> [NL=<значення>]] + IC=<напруга\ на\ вході\ A> <струм\ на\ вході\ A> + <напруга\ на\ вході\ B> <струм\ на\ вході\ B>.$

Тут  $ZO$  – хвильовий опір лінії (Ом);  $TD$  – затримка сигналу в лінії (с);  $NL$  – електрична довжина лінії на частоті  $F$  ( $NL=l/\lambda$ , де  $l$  – геометрична довжина лінії,  $\lambda$  – довжина хвилі в лінії, за замовчуванням  $NL=0,25$ ).

Однаково сприймаються  $ZO$  (буква «O») і  $Z0$  (цифра «0»). При описі лінії передачі задається параметр  $TD$  або  $F$  і  $NL$ . Наприклад:

$T1\ 1\ 2\ 3\ 4\ ZO=50$

$T2\ 1\ 2\ 3\ 4\ ZO=75\ TD=115\ ns$

T3 1 2 3 4 ZO=75 F=4.5 MEG

T4 1 2 3 4 ZO=75 F=4.5 MEG NL=0.5

За допомогою ключового слова *IC* задаються початкові значення напруг і струмів на виводах лінії передачі. Модель лінії передачі без втрат зображена на рис. 3.17, а). При розрахунку частотних характеристик ця лінія являє собою безінерційну ланку.

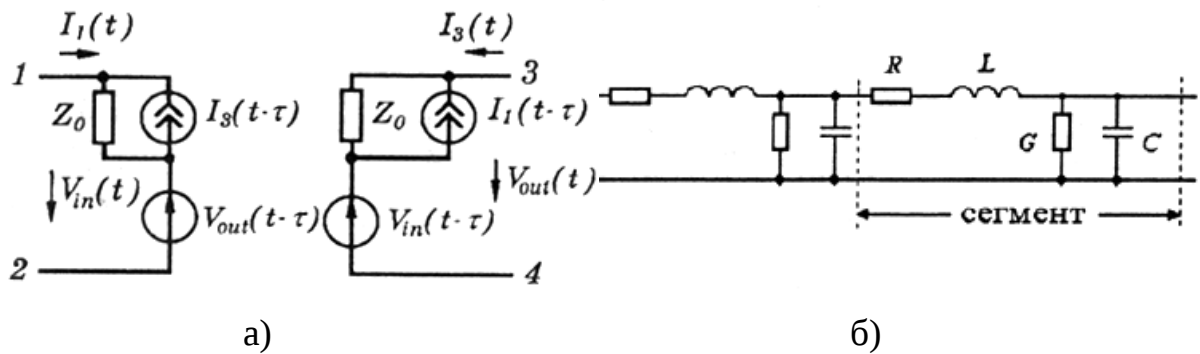


Рисунок 3.17 – Модель лінії передачі без втрат (а) і з їх врахуванням (б)

2. Лінія передач із втратами (рис. 3.17, б) описується: *Txxx* <+вузол порту А> <-вузол порту А> <+вузол порту В> + <-вузол порту В> [*ім'я моделі*] [*електрична довжина*]+*LEN*=<значення> *R*=<значення> *L*=<значення> *G*=<значення>+*C*=<значення>.

Тут *LEN* – довжина лінії (м); *R*, *L*, *G*, *C* – погонні опір лінії (Ом/м), індуктивність (Гн/м), провідність (См/м), ємність (Ф/м), відповідно.

Загалом довжину лінії можна вказувати в будь-яких одиницях, наприклад у кілометрах, але тоді потрібно відповідним чином перерахувати значення погонних параметрів *R*, *L*, *G* і *C*.

За цими даними аналітично розраховується комплексний коефіцієнт передачі лінії. Аналіз перехідних процесів виконується за допомогою інтеграла згортки з імпульсною характеристикою лінії, що обчислюється як перетворення Фур'є коефіцієнта передачі.

Модель лінії передачі описується: *.MODEL* <ім'я моделі> *TRN* (*параметри моделі*).

Вона характеризується параметрами, ідентичними до опису самої лінії передачі без втрат або з втратами [11].

### 3.7 Опис моделей керованих ключів

У мові Spice є моделі ключів, керованих напругою або керованих струмом. Ключі, керовані напругою або струмом, використовуються при розрахунку перехідних процесів.

Ключ, керований напругою, описується:  $S_{xxx} <+вузол> <-вузол> <+керуючий вузол> <-керуючий вузол> + <ім'я моделі>$ .

Тут  $<+вузол>$  і  $<-вузол>$  – номери вузлів, до яких приєднаний ключ;  $<+керуючий вузол>$ ,  $<-керуючий вузол>$  – номери вузлів, різниця потенціалів між якими керує ключем.

Параметри ключа задаються за директивою:  $.MODEL <ім'я моделі> VSWITCH (VON=<значення> + VOFF==<значення> RON=<значення> ROFF=<значення>)$ .

Параметри моделі ключа, керованого напругою, наведені в табл. 3.2.

Таблиця 3.2 – Параметри моделі ключа, керованого напругою

Позначення	Параметр	Розмірність	Значення за замовчуванням
$VON$	Напруга замикання ключа	В	1
$VOFF$	Напруга розмикання ключа	В	0
$RON$	Опір замкнутого ключа	Ом	1
$ROFF$	Опір розімкнутого ключа	Ом	$10^6$

Якщо  $VON > VOFF$ , то ключ замкнутий при керуючій напрузі  $V_{кер} > VON$  і розімкнутий при  $V_{кер} < VOFF$ . На інтервалі  $VOFF < V_{кер} < VON$  опір ключа плавно змінюється від значення  $ROFF$  до  $RON$ . Якщо  $VON < VOFF$ , то ключ замкнутий при  $V_{кер} < VON$  і розімкнутий при  $V_{кер} > VOFF$ .

Ключ, керований струмом незалежного джерела напруги, описується:  $W_{xxx} <+вузол> <-вузол> <ім'я керуючої вітки> <ім'я моделі> [11]$ .

### 3.8 Моделі напівпровідникових приладів

Моделі напівпровідникових приладів, як правило, нелінійні, можна сформувати двома способами. При першому (фізичному) способі проводять аналіз електричних процесів, що протікають в структурі напівпровідникового приладу, виражають струми й напруги на електродах



приладу відповідно до процесів, що відбуваються в ньому, що, власне, і дає ВАХ приладу. Отриману модель показують базовими моделями резисторів, індуктивностей і ємностей, залежними й незалежними джерелами струму й напруги, додавши реактивні компоненти, що відповідають властивостям конструктивного оформлення приладу, одержують повну модель, причому до неї входять параметри структури приладу і його корпусу.

При другому підході використовується ВАХ приладу, що відома або знята експериментально. ВАХ апроксимується із заданою точністю на тій ділянці, на якій передбачається робота приладу. При цьому модель буде містити деякі емпіричні коефіцієнти, не пов'язані з реальною фізичною структурою приладу. При другому підході за основу може бути прийнята також деяка графічна модель.

Розглянемо моделі діодів та транзисторів.

### 3.8.1 Опис моделі напівпровідникового діода

Виведення ВАХ напівпровідникового діода використовує розв'язок рівняння безперервності, що пов'язує концентрацію носіїв заряду в будь-якій точці напівпровідника з параметрами електричного поля в напівпровіднику, швидкістю генерації й рекомбінації носіїв заряду, дифузією носіїв і часом. При цьому, з урахуванням деяких допущень (нульова ширина р-п-переходу, відсутність омичного опору базової області переходу, відсутність явища пробою переходу), ВАХ переходу має вигляд:

$$I = I_0 \cdot \left( e^{\frac{U}{\varphi_T}} - 1 \right),$$

де  $I_0 = q \cdot S \cdot \left( \frac{D_p p_{n0}}{L_p} + \frac{D_n n_{p0}}{L_n} \right)$ ;  $q$  – заряд електрона;  $S$  – площа р-п-переходу;

$p_{n0}$ ,  $n_{p0}$  – концентрації неосновних носіїв в областях п і р, відповідно;  $D_p$ ,  $D_n$  – коефіцієнти дифузії дірок і електронів.

Статична характеристика діода, що відповідає наведеному виразу, зображена на рис. 3.18.

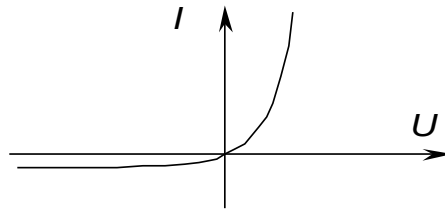


Рис. 1.26.

Рисунок 3.18 – Статична характеристика діода

Останній вираз і характеристика є статичними характеристиками (моделлю) р-п-переходу. Точність моделі відносно невисока, що пояснюється прийнятими припущеннями. Реальна ВАХ відрізняється від ідеальної наявністю ділянки рекомбінаційної складової прямого струму, термогенераційної складової зворотного струму, струмами втрат, явищами пробую на зворотній вітці ВАХ, нарешті, наявністю опору базової області р-п-переходу і ємностями р-п-переходу, що характеризують інерційність процесів у приладі.

ВАХ реального діода показана на рис. 3.18, а), а його еквівалентна схема – рис. 3.19, б).

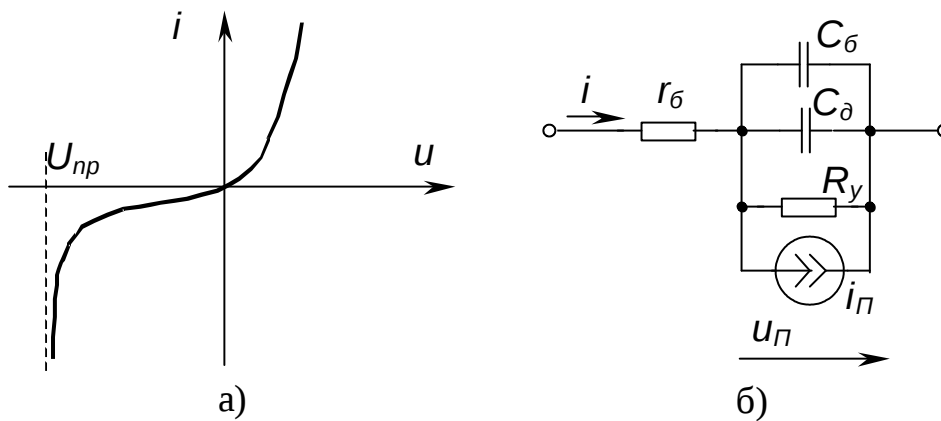


Рисунок 3.19 – ВАХ (а) та еквівалентна схема (б) реального діода

У зазначеній моделі  $i_n(u_n)$  – нелінійне джерело струму, що описує статичний режим приладу у вигляді:

$$i = (i_n + i_y) = I_0 \cdot \left[ e^{\frac{u - i \cdot r_b}{m \varphi_T}} - 1 \right] + \frac{u - i \cdot r_b}{R_y} - I_0 \cdot e^{A(U_{пр} + B u)}.$$

В останньому виразі  $I_0$  – тепловий струм приладу;  $r_6$  – опір базової області переходу;  $m$  – коефіцієнт, що враховує реальну конфігурацію переходу;  $U_{np}$  – напруга лавинного або тунельного пробію переходу;  $A, B$  – коефіцієнти апроксимації ВАХ поблизу ділянки пробію, причому  $A < 0, B > 0$ ;  $R_v$  – опір витоку р-п-переходу. Необхідно відзначити, що багато величин, які входять у модель – нелінійні. Опір бази  $r_6$ , наприклад, зменшується із зростанням струму через діод внаслідок ефекту модуляції опору, пов'язаного з кількістю носіїв заряду в базовій області. Залежність  $r_6(i)$  можна подати у вигляді:

$$r_6 = \frac{1}{k_1 + k_2 \cdot i},$$

де  $k_1, k_2$  – деякі коефіцієнти.

Визначення коефіцієнтів відбувається при подачі на діод імпульсів прямокутного струму амплітудою  $I$ . Напруга на діоді формується за рахунок заряду бар'єрної й дифузійної ємностей, а також устанавлення напруги на  $r_6$ .

Ємнісні складові моделі – бар'єрна й дифузійна ємність р-п-переходу. Бар'єрна ємність  $C_6$  – нелінійна ємність, обумовлена наявністю об'ємного заряду нерухоливих іонів донорних і акцепторних домішок поблизу границь області об'ємного заряду. Нелінійна величина, що залежить від напруги на переході відповідно до виразу:

$$C_6 = C_{60} \left[ \frac{\Delta\varphi_0}{\Delta\varphi_0 - U_{п}} \right]^n,$$

де  $C_{60}$  – величина бар'єрної ємності при  $U_{п}=0$ ,  $\Delta\varphi_0$  – контактна різниця потенціалів, що дорівнює 0,5–1 В для напівпровідників різних типів,  $n$  – показник ступеня (1/2 для різких переходів і 1/3 для плавних).

Формула точно відображає поведінку бар'єрної ємності при зворотному зсуві ( $U_{п} < 0$ ) і може дати більші похибки при  $U_{п} > 0$ , оскільки в області великих струмів діода, коли  $U_{п} \approx \Delta\varphi_0$ ,  $C_6 \rightarrow \infty$ . Під час розрахунку можуть виникати помилки типу ділення на нуль або переповнення розрядної сітки машини. Ємністю  $C_6$  при розрахунках знехтувати практично не можна, тому запропоновано різні варіанти усунення подібних неточностей і

помилки. Наприклад, модель Гуммеля–Пуна апроксимує ємність  $C_6(U_{II})$  на прямій ділянці ВАХ виразом:

$$C_6 = C_{60} \left[ 1 + n \cdot \frac{U_{II}}{\Delta\varphi_0} \right]$$

На прямій ділянці вольт-амперної характеристики важливе значення має нелінійна дифузійна ємність р-п-переходу, що визначається ефектом накопичення носіїв заряду в області бази при прямому зсуві. Величина  $C_6(i)$  визначається співвідношенням:

$$C_6 = (I_0 + i) \cdot \frac{\tau}{m\varphi_T}$$

де  $\tau$  – стала часу життя неосновних носіїв заряду в області бази р-п-переходу [1].

Напівпровідникові прилади, математичні моделі яких вбудовані в програму PSpice, описуються великою кількістю параметрів, що задаються за допомогою директиви *.MODEL*. Директиву *.MODEL* можна помістити в опис аналізованої схеми або у файл бібліотеки, доступ до якого здійснюється за допомогою директиви *.LIB*. Опис конкретного напівпровідникового приладу містить його ім'я, номери вузлів підключення, ім'я моделі і коефіцієнт кратності *Area*, за допомогою якого імітується паралельне підключення декількох однакових приладів.

Діод описується: *Dxxx <вузол анода> <вузол катода> <ім'я моделі> + [<коефіцієнт кратності Area>]*.

Модель діода задається у вигляді: *.MODEL <ім'я моделі> D[(параметри моделі)]*.

Розглянемо приклади. Підключимо між вузлами 1 і 2 діод D9B, параметри якого вводяться за допомогою директиви *.MODEL*

*D1 1 2 D9B*

*.MODEL D9B D(IS=5UA RS=14 BV=2.81 IBV=5UA)*

Підключимо між вузлами 1 і 2 діод D104A, параметри якого записані в бібліотечному файлі *D.LIB*

*V1 0 10V*

*R 1 2 100*

```

D1 1 2 D104A .LIB D.LIB
.DC V -0.5V 10V 0.02V
.MODEL DMOD D
.PROBE
.END

```

### 3.8.2 Опис моделей біполярного транзистора

Транзистор являє собою структуру, що складається з 2-х р-п-переходів. У нормальному підсилювальному режимі роботи емітерний перехід інжектує носії заряду, які, частково рекомбінуючи в області бази, потрапляють у колекторний перехід і створюють колекторний струм. Струм рекомбінації носіїв заряду в базі створює струм бази. Геометрія транзистора визначається таким чином, щоб рекомбінаційна складова струму бази була відповідною інжекційній складовій струму емітера, тому створюється можливість малою величиною базового струму керувати значними величинами струму колектора. Таким чином, транзистор має здатність підсилення струму, що визначає його основні властивості.

Модель біполярного транзистора Еберса–Молла може бути відображена двома джерелами струму й двома р-п-переходами, а також елементами, що доповнюють моделі р-п-переходів (рис. 3.20).

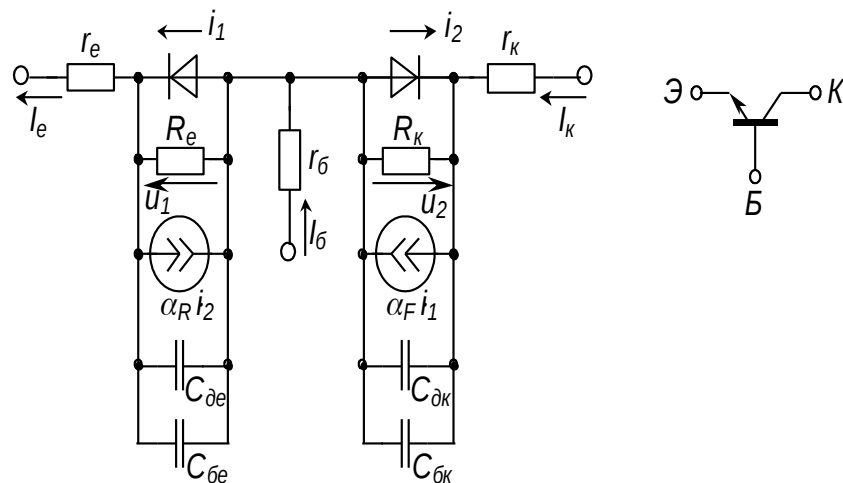


Рисунок 3.20 – Модель біполярного транзистора Еберса-Молла

Така модель, запропонована Еберсом і Моллом в 1954 році, одержала

назву «Модель Еберса–Молла».

Як і для випадку р-п-переходу, струми діодів, інжектованих через переходи, визначаються у вигляді:

$$i_1 = I'_{e0} \left[ e^{\frac{U_1}{m_s \varphi_T}} - 1 \right]; \quad i_2 = I'_{k0} \left[ e^{\frac{U_2}{m_k \varphi_T}} - 1 \right].$$

У статиці, нехтуючи струмами через опори витоків, можна записати:

$$I_e = I'_{e0} \left[ e^{\frac{U_1}{m_s \varphi_T}} - 1 \right] - \alpha_R \cdot I'_{k0} \left[ e^{\frac{U_2}{m_k \varphi_T}} - 1 \right]; \quad I_k = \alpha_F \cdot I'_{e0} \left[ e^{\frac{U_1}{m_s \varphi_T}} - 1 \right] - I'_{k0} \left[ e^{\frac{U_2}{m_k \varphi_T}} - 1 \right];$$

$$I_b = I_e - I_k.$$

Таким чином, величини струмів  $I_{e0}'$  і  $I_{k0}'$  відповідають струмам емітерного і колекторного переходів при величині напруги  $U_2=0$  і  $U_1=0$ , відповідно, тобто при закороченні виводів колектора й бази та емітера й бази, відповідно, а також значній величині негативної напруги на переході.

Біполярний транзистор характеризується коефіцієнтом передачі струму емітера  $\alpha$  або струму бази  $\beta$ . Зв'язок між коефіцієнтами здійснюється співвідношеннями:

$$\alpha = \frac{I_k}{I_e}; \quad \beta = \frac{I_k}{I_b}; \quad \alpha = \frac{\beta}{\beta + 1}; \quad \beta = \frac{\alpha}{1 - \alpha}.$$

Динамічні параметри транзистора враховуються в моделі Еберса–Молла ємнісними параметрами, а саме бар'єрними ємностями  $C_{be}$  й  $C_{bk}$  та дифузійними ємностями  $C_{de}$  й  $C_{dk}$ :

$$C_{de} = \frac{\tau_F}{m_e \varphi_T} \cdot (i_e + I'_{e0}); \quad C_{dk} = \frac{\tau_R}{m_k \varphi_T} \cdot (i_k + I'_{k0}),$$

де  $\tau_F$  і  $\tau_R$  – сталі часу життя для нормального й інверсного включення транзистора.

Слід зазначити, що транзистор являє собою нелінійну структуру, тому всі його параметри  $\alpha$ ,  $\beta$ ,  $C_k$ ,  $C_e$  не постійні, а залежать від режиму, тобто від

величин напруг  $u_1$  і  $u_2$ , а також струмів  $i_1$  і  $i_2$  [1].

Біполярний транзистор в програмі PSpice вибирається або з бібліотеки, або записується у вигляді моделі.

Біполярний транзистор описується:  $Q_{xxx}$  <вузол колектора> <вузол бази> <вузол емітера> + [<вузол підкладки>] <ім'я моделі> + [<коефіцієнт кратності Area>].

Моделі біполярних транзисторів задаються у вигляді:

.MODEL <ім'я моделі> NPN [(параметри моделі)]

.MODEL <ім'я моделі> PNP [(параметри моделі)]

.MODEL <ім'я моделі> LPNP [(параметри моделі)]

Статично індукований біполярний транзистор описується:  $Z_{xxx}$  <вузол колектора> <вузол затвора> <вузол емітера> + <ім'я моделі> [AREA=<значення>] [WB=<значення>] + [AGD=<значення>] [KP=<значення>] [TA=<значення>].

Моделі статично індукованих біполярних транзисторів задаються у вигляді .MODEL <ім'я моделі> NIGBT [(параметри моделі)].

До основних параметрів моделей біполярного транзистора можна віднести:

- $I_S$  – струм насичення;
- $V_{AF}$  – ефект Ерлі;
- $B_F$  – максимальний коефіцієнт підсилення у схемі зі спільним емітером;
- $B_R$  – максимальний коефіцієнт підсилення у інверсному режимі;
- $C_{JC}$  – ємність колектор–база при нульовому зміщенні;
- $C_{JE}$  – ємність емітер–база при нульовому зміщенні;
- $T_R$  – час переносу заряду в інверсному режимі;
- $T_F$  – час переносу носіїв через базу в нормальному режимі;
- $R_C$  – опір колектора;
- $R_B$  – опір бази;
- $X_{T1}$  – температурний коефіцієнт струму насичення;
- $E_G$  – ширина забороненої зони;
- $N_E$  – коефіцієнт неідеальності емітерного переходу;
- $I_{SE}$  – зворотний струм емітерного переходу;
- $I_{SC}$  – зворотний струм колекторного переходу;
- $I_{KF}$  – струм початку спаду залежності  $B_F$  від струму колектора;
- $X_{TB}$  – температурний коефіцієнт для параметрів  $B_F$  і  $B_R$ ;
- $N_K$  – коефіцієнт неідеальності колекторного переходу;

- $FC$  – коефіцієнт нелінійності бар'єрних ємностей колектор–база та емітер–база;
- $ITF$  – коефіцієнт, що характеризує залежність параметра  $TF$  від струму колектора;
- $UTF, XTF$  – напруга та коефіцієнт, що характеризують залежність параметра  $TF$  від зміщення база–колектор.

Більшість цих параметрів можна не задавати при описі моделі, оскільки вони мають значення за замовчуванням. Приклад опису біполярного транзистора:

`Q1 1 2 0 KT`

`.Model KT NPN (IS=6.25F BF=529 BR=2.54 CJC=0.189pF CJE=0.3pF TR=80ns TF=25ps RC=22 RB=45 ITF=0.52)`

### 3.8.3 Опис моделей польового транзистора

Польові транзистори – прилади, у яких використовуються ефекти зміни параметрів напівпровідника під впливом електричного поля. Польові транзистори за принципом дії поділяються на прилади з керуючим р-п-переходом, МДН-транзистори, МДН-транзистори з вертикальним каналом. Розглянемо принцип роботи МДН-транзистора (рис. 3.21).

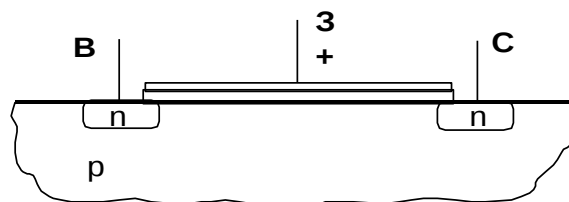


Рисунок 3.21 – МДН-транзистор

При подачі на затвор позитивного потенціалу достатньої величини відбувається інверсія провідності в приповерховому шарі напівпровідникової підкладки, у результаті чого утвориться канал п-типу, провідність якого залежить від величини прикладеної напруги. Таким чином, величина струму стоку польового транзистора виявляється залежною від величини напруги затвор–витік. Більш детальний розгляд процесів у транзисторі приводить до того, що струм стоку залежить від



напруги стік–витік внаслідок ефекту перекриття каналу поблизу області стоку. Таким чином, польовий транзистор у статиці наводиться нелінійним джерелом струму виду  $i(U_{зв}, U_{св})$ . Модель польового транзистора для невисоких частот та її ВАХ наведені на рис. 3.22.

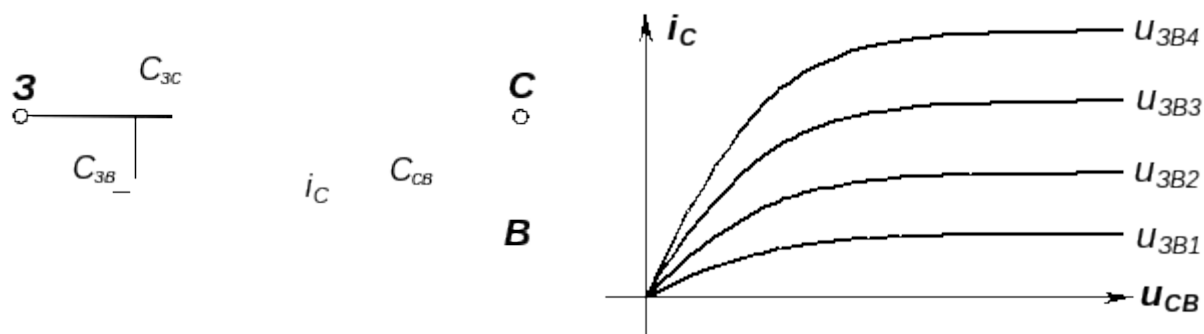


Рисунок 3.22 – Модель польового транзистора для невисоких частот та її ВАХ

На рисунку  $C_{зв}$ ,  $C_{зс}$ ,  $C_{св}$  – нелінійні ємності, утворені перекриттям затвора з областями стоку й витоку, а також ємностями виводів контактів стоку й витоку.

Нелінійну ВАХ джерела струму можна апроксимувати з достатньою точністю виразом

$$i = S \cdot (U_{зв} + U_0 + bU_{зв}^2) \cdot \left[ 1 - e^{-\frac{pU_{св}}{U_{зв} + U_0 + bU_{зв}^2}} \right],$$

де  $S$ ,  $b < 0$ ,  $p > 0$  – параметри апроксимації.

Для іншого різновиду МДН-транзисторів – польових приладів з вертикальною структурою – характерний перегин передатних вольт-амперних характеристик. При цьому апроксимація ВАХ виявляється більш точною в такому вигляді при  $k > 0$ :

$$I_c = M \cdot \left[ 1 - e^{-\frac{kU_{св}}{M}} \right]; \quad M = I_s \cdot \left[ 1 + th \left( \frac{S(U_{зв} - U_s)}{I_s} \right) \right].$$

У програмі PSpice використовується модель Шихмана–Ходжеса:

$$I_c = \begin{cases} 0 & \text{при } U_{зв} \leq U_{пор}; \\ \beta \cdot (1 + \lambda \cdot U_{св}) \cdot (U_{зв} - U_{пор})^2 & \text{при } U_{пор} \leq U_{зв} \leq U_{св} + U_{пор} \quad (\text{насич.}); \\ \beta \cdot (1 + \lambda \cdot U_{св}) \cdot U_{св} \cdot [2 \cdot (U_{зв} - U_{пор}) - U_{св}] & \text{при } U_{св} + U_{пор} \leq U_{зв} \quad (\text{лін.}). \end{cases}$$

Польові транзисторі у мові Spice описуються таким чином.

Польовий транзистор з керуючим р-п-переходом описується: *Jxxx* <вузол стоку> <вузол затвора> <вузол витоку> + <ім'я моделі> [<коефіцієнт кратності Area>].

Моделі польових транзисторів задаються у вигляді:

*.MODEL* <ім'я моделі> *NJF* [(параметри моделі)]

*.MODEL* <ім'я моделі> *PJF* [(параметри моделі)]

Арсенід-галієвий польовий транзистор з керуючим р-п-переходом і каналом п-типу описується: *Vxxx* <вузол стоку> <вузол затвора> <вузол витоку> + <ім'я моделі> + [<коефіцієнт кратності Area>].

Модель арсенід-галієвого польового транзистора задається у вигляді:

*.MODEL* <ім'я моделі> *GASFET* [(параметри моделі)].

МОН-транзистор описується: *Mxxx* <вузол стоку> <вузол затвору> <вузол витоку> + <вузол підкладинки> <ім'я моделі> [*L*=<значення>] + [*W*=<значення>][*AD*=<значення>] [*AS*=<значення>] + [*PD*=<значення>] [*PS*=<значення>][*NRD*=<значення>] + [*NRS*=<значення>] [*NRG*=<значення>] [*NRB*=<значення>] + [*M*=<значення>].

Параметри *L* і *W* можуть бути задані при описі моделі МОН-транзистора за директивою *.MODEL*. Крім того, параметри *L*, *W*, *AD* і *AS* за замовчуванням набувають значень, що задаються директивою *.OPTIONS*.

Моделі МОН-транзисторів задаються у вигляді:

*.MODEL* <ім'я моделі> *NMOS* [(параметри моделі)]

*.MODEL* <ім'я моделі> *PMOS* [(параметри моделі)]

### 3.9 Моделювання схем засобами PSpice OrCAD

Схема для моделювання, зображена на рис. 3.23, містить джерело напруги та чотири резистори. Вузли на схемі позначено номерами від 0 до 3.

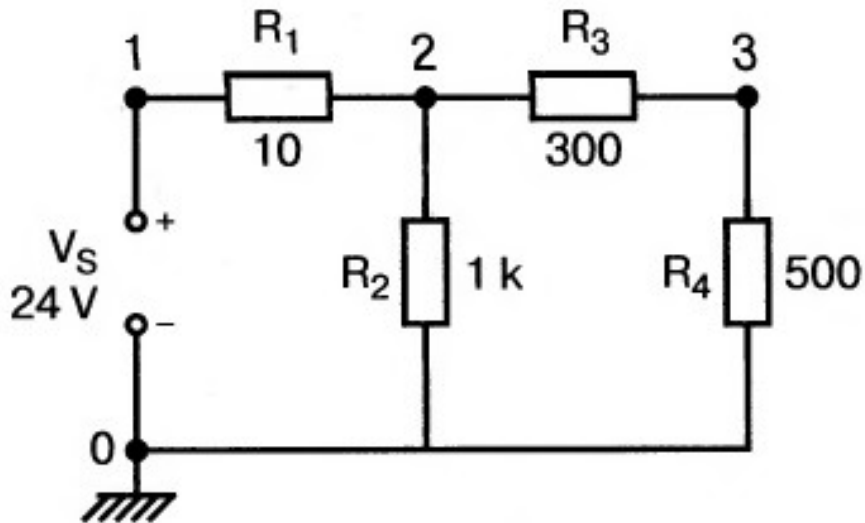


Рисунок 3.23 – Схема для моделювання

Для її моделювання необхідно відкрити програму *pspice.exe* та вибрати команду меню *File-new-text file*. Далі необхідно ввести такий текст, що описує дану схему:

```

First Circuit for Pspice
VS 1 0 24V
R1 1 2 10
R2 2 0 1k
R3 2 3 300
R4 3 0 500
.OPT porage
.OP
.END

```

Даний текст містить назву схеми, опис елементів схеми, команду *.OPT*, яка визначає опції (параметри) запуску схеми (*porage* – параметр, що забирає зайві колонтитули з вихідного файлу), команду *.OP*, що визначає робочу точку схеми (у даному випадку є несуттєвою), команду *.END* – кінець програми.

Далі необхідно зберегти даний файл з розширенням *\*.CIR*.

Збережений файл можна запуснути на виконання, виконавши команду меню *Simulation-Run*. Для перегляду результатів моделювання необхідно виконати команду *View-Output File*, після чого на екран виведеться результат моделювання (рис. 3.24).

```

First Circuit for PSpice

****   CIRCUIT DESCRIPTION

VS 1 0 24V R1 1 2 10
R2 2 0 Ik
R3 2 3 300
R4 3 0 500
.DC VS 24V 24V 24V
.PRINT DC I(R1) I(R2) I(R3)
.OPT nopage
.OP
.END

****   DC TRANSFER CURVES           TEMPERATURE = 27.000 DEG C
VS   I(R1)   I(R2)   I(R3)
2.400E+01  5.281E-02  2.347E-02  2.934E-02

****   SMALL SIGNAL BIAS SOLUTION TEMPERATURE = 27.000 DEG C

NODE VOLTAGE   NODE VOLTAGE   NODE VOLTAGE   NODE VOLTAGE
( 1) 24.0000   ( 2) 23.4720   ( 3) 14.6700

VOLTAGE SOURCE CURRENTS
NAME      CURRENT
VS        -5.281E-02

TOTAL POWER DISSIPATION 1.27E+00 WATTS

****   OPERATING POINT INFORMATION TEMPERATURE = 27.000 DEG C
JOB CONCLUDED
TOTAL JOE TIME           .10

```

Рисунок 3.24 – Результат моделювання

Результатом моделювання буде виведення на екран потенціалів у вузлах схеми та повної розсіюваної потужності.

Щоб отримати більш повну інформацію потрібно додати до опису схеми рядки:

```

.DC VS 24V 24V 24V
.PRINT DC I(R1) I(R2) I(R3)

```

Перший рядок вказує на аналіз при постійному струмі (назва джерела, початкове значення, кінцеве значення, крок). Другий рядок – на виведення необхідних даних. Вихідний файл уже містить розрахунок струмів на резисторах  $R_1$ ,  $R_2$ ,  $R_3$ .

Отримати більше вихідної інформації можна, додавши рядок:  
`.TF V(2) VS`

Дана команда виводить відношення потенціалу вузла 2 до потенціалу джерела  $V_s$ , а також вхідний опір відносно  $V_s$  та вихідний опір відносно  $V(2)$ .

Аналіз кола на змінному струмі (рис. 3.25) виконується аналогічно.

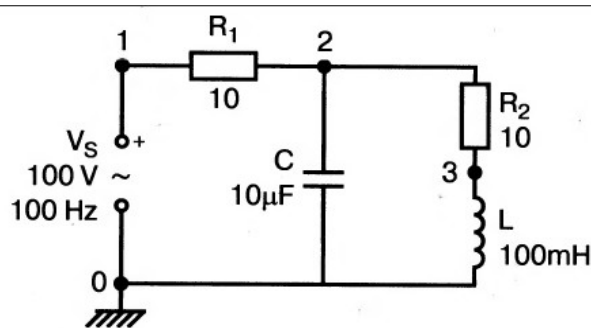


Рисунок 3.25 – Коло на змінному струмі

Моделювання схеми на рис. 3.25 здійснюється за допомогою програми:

```
Series-parallel ac Circuit
Vs 1 0 ac 100V
R1 1 2 10
R2 2 3 10
L 3 0 100mH
C 2 0 10uF
.ac LIN 1 100Hz 100Hz
.PRINT ac I(R1) IP(R1) V(2) VP(2)
.PRINT ac I(C) IP(C) I(R2) IP(R2)
.OPT nopage
.END
```

ents%20and%20Settings/%D0%90%D0%B4%D0%BC%D0%B8%D0%BD%

Команда `.ac LIN` проводить лінійний аналіз за змінним струмом.  $I(..)$  – визначає струм в даній вітці,  $IP(..)$  – фазовий кут цього струму у градусах.

Модифікувавши файл, отримаємо:

```
Series-parallel ac Circuit
Vs 1 0 ac 100V
R1 1 2 10
R2 2 3 10
L 3 0 100mH
C 2 0 10uF
.ac LIN 151 50Hz 200Hz
.probe
.END
```

Аналіз буде проведено для 151 значення частоти в діапазоні від 50 до 200 Гц. Команда `.probe` приведе до виведення на екран графіка цієї залежності.

Аналогічно можна промодельовати і схему з транзистором (рис. 3.26).

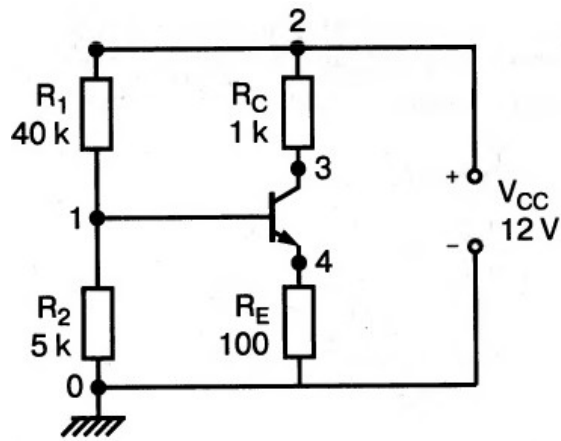


Рисунок 3.26 – Схема з транзистором

Моделювання можна описати за допомогою такої програми:

```

BJT Biasing Circuit
VCC 2 0 12V
R1 2 1 40k
R2 1 0 5k
RC 2 3 1k
RE 4 0 100
Q1 3 1 4 QN
.MODEL QN NPN(BF=80)
.dc VCC 12V 12V 12V
.OP
.OPT nopage
.PRINT dc I(R1) I(R2) I(RC) I(RE)
.END

```



## КОНТРОЛЬНІ ЗАПИТАННЯ

1. Який базовий набір моделей елементів в OrCAD?
2. Які особливості опису моделей резисторів?
3. Як описуються моделі індуктивностей?
4. Як можна описати моделі конденсаторів?
5. Як описуються моделі трансформаторів?
6. Які особливості опису моделей лінії передачі?
7. Як описуються моделі керованих ключів?
8. Як можна описати моделі напівпровідникового діода?
9. Як описуються моделі біполярного транзистора?
10. Які особливості опису моделей польового транзистора?



## 4 МОДЕЛІ КОМПОНЕНТІВ ЦИФРОВОЇ ЕЛЕКТРОНИКИ

### 4.1 Основні поняття опису цифрових компонентів

Спочатку програма PSpice була призначена для моделювання чисто аналогових пристроїв. В даний час вона забезпечує моделювання змішаних аналого-цифрових схем, які, зокрема, можуть складатися тільки з цифрових пристроїв і не містити аналогових блоків.

Зазвичай змішані пристрої моделюються в режимі *.TRAN* (розрахунок перехідних процесів), однак інші режими також доступні. В режимі *.DC* затримки сигналів в цифрових блоках ігноруються і розраховуються логічні рівні виходів цифрових компонентів в стаціонарному режимі. В режимах *.AC*, *.NOISE*, *.TF* і *.SENS* цифрові компоненти не беруть участі в аналізі малосигнальних частотних характеристик кола, лише для аналогових частин аналого-цифрових та цифро-аналогових інтерфейсів складаються лінеаризовані схеми заміщення їхніх вхідних і вихідних комплексних опорів. Розглянемо специфіку моделювання цифро-аналогових пристроїв.

Модель інтегральної схеми (ІС) (рис. 4.1) в програмі PSpice показано у вигляді примітивів *Uxxx*, що відбивають їх функціонування на логічному рівні, і двох аналого-цифрових та цифро-аналогових інтерфейсів, що відображають їх вхідні і вихідні каскади.

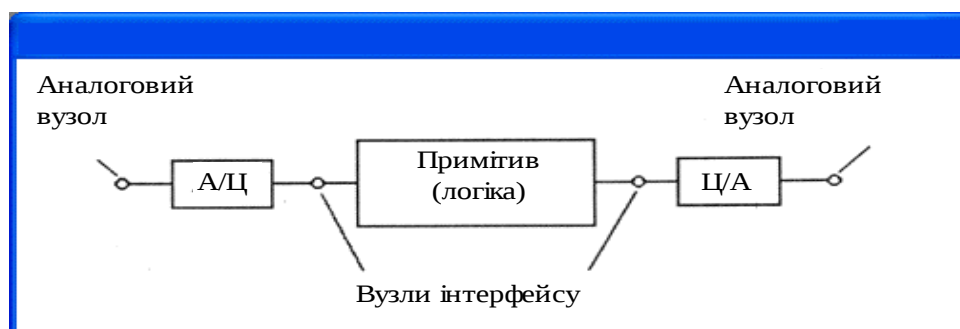


Рисунок 4.1 – Модель цифрової ІС

У завданні на моделювання вказуються тільки примітиви цифрових пристроїв  $U_{xxx}$ . Якщо при цьому цифрові ІС з'єднуються безпосередньо один з одним, то блоки інтерфейсів до уваги не беруться. Якщо ж до входу або виходу мікросхеми підключений аналоговий компонент, то необхідно включити відповідний інтерфейс. В принципі, вони можуть бути вручну внесені в опис завдання на моделювання (у файл. \*.CIR) як окремі компоненти, проте програма PSpice Schematics робить це автоматично.

Отже, змішані аналого-цифрові схеми складаються з компонентів трьох типів:

- аналогові компоненти;
- пристрої з'єднання аналогових і цифрових компонентів (пристрої інтерфейсу);
- цифрові компоненти (примітиви).

Відповідно розрізняють три типи вузлів:

- аналогові вузли, до яких підключені тільки аналогові пристрої;
- цифрові вузли, до яких підключені тільки цифрові пристрої;
- вузли інтерфейсу, до яких підключена комбінація аналогових і цифрових пристроїв.

Програма PSpice Schematics автоматично розщеплює кожен вузол інтерфейсу на два вузли – чисто аналоговий і чисто цифровий – і підключає між ними макромодель аналого-цифрового або цифро-аналогового інтерфейсу. Крім того, до моделей інтерфейсів автоматично підключається джерело живлення цифрових схем.

Логічні рівні цифрових вузлів набувають одного з таких значень:

- 1 – високий рівень;
- 0 – низький рівень;
- R – позитивний фронт (*Raise*, перехід зі стану «0» в «1»);
- F – негативний фронт (*Fall*, перехід зі стану «1» в «0»);
- X – невизначений стан (може набувати значень «0», «1», проміжний або нестабільний стан);
- Z – стан високого вихідного опору (логічний рівень може бути високим, низьким, проміжним або нестабільним).

При обчисленні логічних рівнів вузлів, до яких підключено кілька цифрових компонентів, беруться до уваги вихідні опори джерел сигналів.

Розглянемо описи пристроїв інтерфейсу.

Пристрої інтерфейсу підключаються між аналоговими і цифровими компонентами і виконують дві функції. По-перше, з їх допомогою при



моделюванні електричних процесів в аналоговій частині кола задаються схеми заміщення вхідних і вихідних каскадів цифрових компонентів, з'єднаних безпосередньо з аналоговими компонентами. По-друге, вони забезпечують перетворення електричної напруги в логічний рівень і навпаки, щоб забезпечити обмін даними між підпрограмами моделювання електричних процесів в аналоговій частині кола і логічного моделювання цифрової частини. Вони поділяються на пристрої передачі даних від аналогових компонентів на вхід цифрових компонентів, так званий інтерфейс А/Ц (Digital Output), і на пристрої передачі даних від цифрових компонентів на вхід аналогових компонентів, інтерфейс Ц/А (Digital Input).

Якщо аналогові і цифрові компоненти взаємодіють в процесі моделювання, пристрої інтерфейсу підключаються в схему заміщення вітки кола автоматично, коли вони з'єднуються один з одним. Користувач при цьому не вносить пристрій з'єднання в файл опису схеми з розширенням \*.CIR. Для забезпечення такого режиму попередньо в бібліотеки цифрових компонентів вносяться асоційовані з кожним компонентом моделі пристроїв інтерфейсу, оформлені у вигляді макромоделей. При розщепленні вузла інтерфейсу для автоматичного підключення пристрою інтерфейсу програма PSpice Schematics створює новий цифровий вузол. Перелік цих вузлів, створених програмою, розташовується в вихідному файлі з розширенням \*.OUT в розділі Generated AtoD and DtoA. Зауважимо, що вузол інтерфейсу характеризується електричною напругою, а додатковий цифровий вузол – логічним станом. Користувач має право безпосередньо внести в опис кола пристрою інтерфейсу А/Ц і Ц/А, при цьому програма нові пристрої підключати вже не буде.

Розглянемо, як приклад, змішане аналого-цифрове коло на рис. 4.2. Воно складається з двох цифрових компонентів: генератора цифрового сигналу  $U1$  і логічного вентиля 2I-NI, наведеного у вигляді макромоделі  $X1$ . На один вхід вентиля  $X1$  підключається генератор аналогового сигналу  $VSIN$ , на інший – генератор цифрового сигналу  $U1$ , а до виходу – аналогове RC-коло. Таким чином, тут є два вузли інтерфейсу 1 і 2. До вузла 1 підключений аналоговий компонент  $VSIN$  і вхід цифрового компонента  $X1$ , тому між цими компонентами програмою підключений в схему заміщення інтерфейс А/Ц і створюється додатковий цифровий вузол  $1\$AtoD$ . Аналогічно між виходом цифрового компонента  $X1$  і аналоговим RC-колом буде підключений інтерфейс Ц/А і створений додатковий цифровий вузол  $3\$DtoA$ .

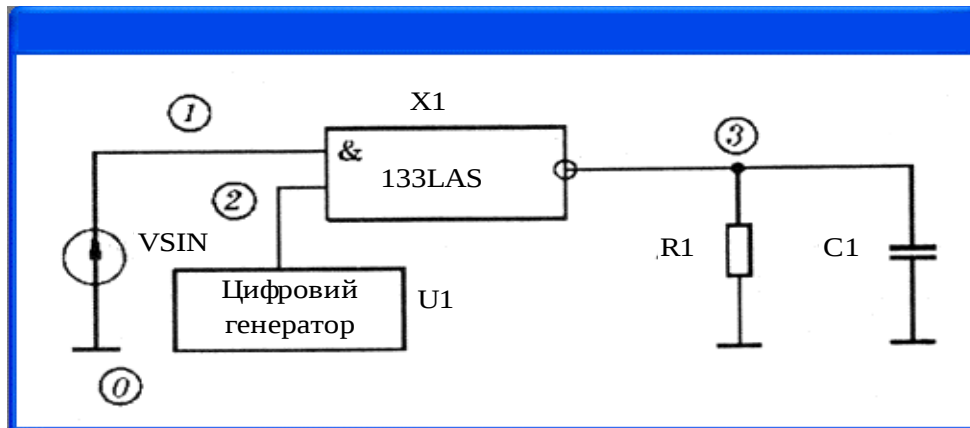


Рисунок 4.2 – Приклад змішаного аналого-цифрового кола

Імена додаткових цифрових вузлів складаються за таким правилом: на початку їхнього імені повторюється ім'я вузла інтерфейсу, до нього додається символ \$ і потім суфікс *AtoD* або *DtoA* – залежно від типу інтерфейсу (ім'я вузла на виході А/Ц набуває суфіксу *AtoD*, на вході Ц/А – *DtoA*). Якщо до одного вузла інтерфейсу підключається не один, а декілька однотипних компонентів, то утворюються додаткові цифрові вузли, в кінці імен яких додаються цифри 2, 3 ... . Крім того, в схемі на рис. 4.2 до інтерфейсу Ц/А автоматично підключається джерело живлення через глобальний вузол  $\$G\_POS$ .

Зазначимо, що схема на рис. 4.2 містить аналогові вузли 1, 3,  $\$G\_POS$  та цифрові вузли  $1\$AtoD$ ,  $2, 3\$DtoA$ . Потенціали аналогових вузлів і логічні стани цифрових вузлів можна вивести на друк, наприклад, за директивою `.PRINT TRAN V (1) D (2) V (3)`.

Отже, кожному реальному цифровому компоненту ставляться у відповідність:

1. Два пристрої інтерфейсу для сполучення з аналоговими пристроями, які можуть підключатися до його входу і виходу, названі інтерфейсами Ц/А і А/Ц, які здійснюють обмін даними між підпрограмами моделювання аналогових і цифрових пристроїв програми PSpice;

2. Моделі вхід/вихід, які відобразатимуть його вхідні і вихідні комплексні опори;

3. Моделі динаміки, що враховують затримку сигналів. Моделі цифрових компонентів і асоційовані з ними супутні моделі поміщаються в спеціальні бібліотеки [12].

## 4.2 Загальний опис цифрових компонентів PSpice OrCAD

Цифрові компоненти (примітиви) задаються за форматом:  $U_{xxx} <тип> [(<список параметрів>)] <+вузол джерела живлення> + <-вузол джерела живлення> <список вузлів> + <ім'я моделі динаміки> <ім'я моделі вхід/вихід> + [MNTYMXDLY = <вибір значення затримки>] + [IO_LEVEL = <рівень моделі інтерфейсу>].$

Параметр  $<тип>$  вказує тип логічного пристрою; в круглих дужках вказуються значення одного або більше параметрів через кому. Після списку вузлів підключення логічного пристрою йдуть імена двох моделей. Перша модель описує динамічні властивості пристрою, друга – характеристики вхідних і вихідних опорів.

Моделі динаміки мають ключові слова, наведені в табл. 4.1.

Таблиця 4.1 – Ключові слова моделі динаміки

Ім'я моделі динаміки	Тип компонента
<i>UADC</i>	Аналого-цифровий перетворювач
<i>UBTG</i>	Двонаправлений перемикальний вентиль
<i>UDAC</i>	Цифро-аналоговий перетворювач
<i>UDLY</i>	Цифрова лінія затримки
<i>UEFF</i>	Тригер з динамічним керуванням
<i>UGATE</i>	Стандартний вентиль
<i>UGFF</i>	Тригер з потенційним керуванням
<i>UIO</i>	Модель вхід/вихід цифрового пристрою
<i>UPLD</i>	Програмовані логічні матриці
<i>UROM</i>	Постійний запам'ятовувальний пристрій
<i>URAM</i>	Оперативний запам'ятовувальний пристрій
<i>UTGATE</i>	Вентиль з трьома станами

Моделі вхід/вихід мають ключове слово *UIO*.

Параметр *MNTYMXDLY* дозволяє конкретному індивідуальному приладу призначити мінімальне, типове або максимальне значення часу затримки, вказане в специфікації моделі його динаміки:

- 0 – значення затримки, задане параметром *DIGMNTYMX* директиви *.OPTIONS* (за замовчуванням параметр дорівнює 2);
- 1 – мінімальне значення;
- 2 – типове значення;
- 3 – максимальне значення;

- 4 – розрахунок найгіршого випадку (мінімум/максимум).

Параметр *IO\_LEVEL* вказує тип цифро-аналогового і аналого-цифрового інтерфейсу даного цифрового пристрою:

- 0 – відповідно до значення параметра *DIGIOLVL* директиви *.OPTIONS* (за замовчуванням він дорівнює 1);

- 1 – інтерфейс *AtoD1 / DtoA1*;

- 2 – інтерфейс *AtoD2 / DtoA2*;

- 3 – інтерфейс *AtoD3 / DtoA3*;

- 4 – інтерфейс *AtoD4 / DtoA4*.

Затримка сигналу в примітивах цифрових пристроїв визначається в двох моделях: динаміки і вхід/вихід.

Модель динаміки визначає затримки поширення і такі часові обмеження, як час установа (setup) і утримування (hold). Модель вхід/вихід задає вхідні і вихідні опори, ємності і час перемикання.

Коли вихід примітиву з'єднується з іншим примітивом загальний час затримки поширення першого примітиву дорівнює сумі часу встановлення напруги на його навантаженні і часу поширення сигналу, зазначеного в моделі динаміки. Час встановлення напруги на навантаженні (loading delay) розраховується за формулою:

$$T_{нав} = 0,69R_{вих} C_{нав},$$

де  $R_{вих}$  – вихідний опір пристрою, що дорівнює *DRVH* або *DRVL* залежно від логічного рівня на виході;  $C_{нав}$  – сума вхідних і вихідних ємностей цифрових пристроїв *INLD*, *OUTLD*, підключених до даного виводу.

Коли вивід примітиву підключений до аналогового пристрою, затримка поширення зменшується на величину, яка дорівнює часу перемикання, заданому в моделі вхід/вихід.

Мінімальна тривалість сигналу на вході цифрового примітиву, необхідна для зміни його логічного стану, повинна перевищувати час затримки, наведений в моделі динаміки (це обмеження не стосується цифрових ліній затримки). Більш короткі вхідні імпульси не викличуть на виході ніякого ефекту.

Приклад. Наведемо опис змішаного кола на рис. 4.2:

*Analog / Digital Interface Example*

*.OPT ACCT LIST LIBRARY EXPAND RELTOL = .001*

```

.LIB DIG.LIB
VSIN 1 0 SIN (05v 1MEG)
U1 STIM (1,1) $G_DPWR $G_DGND 2 IO_STD
+ TIMESTEP = 10ns
+ (OC, 1) LABEL = BEGIN
+ (10,0) (20,1) (3C.X) (40,0) (5C.Z)
+ 6C GOTO BEGIN -1 TIMES
X1 1 23133LA3
RL 3 0 25k
CL 3 0 5pF
.TRAN 5ns 500ns
.PRINT TRAN V (1) D (2) V (3)
.PROBE
.END

```

Тут відбувається звернення до бібліотечного файлу моделей цифрових вітчизняних компонентів dig.lib.

Звернемо увагу, що цифрові ІС, навіть найпростіші, для яких є примітиви, наведені в бібліотеці у вигляді макромоделей, імена котрих збігаються з позначенням за ЄСКД (в латинській транскрипції). Це дозволяє користувачеві не замислюватися про правила опису кожної конкретної ІС, надаючи це розробникам бібліотек моделей.

Інформація про макромоделі пристроїв, сполучення яких автоматично підключаються в схему, їх імена і імена нових цифрових вузлів поміщається в вихідному файлі з розширенням \*.OUT [12].

### 4.3 Опис багаторозрядних АЦП та ЦАП

Багаторозрядні АЦП (рис. 4.3) задаються у форматі: *Uxxx ADC (<m>)* *<список вузлів>* *<модель динаміки>* + *<модель вхід / вихід>* + *[MNTYMXDLY = <вибір значення затримки>]* + *[IO\_LEVEL=<рівень моделі інтерфейсу>]*.

Імена вузлів перераховуються в *<списку вузлів>* в такому порядку: *<+ вузол джерела живлення>*, *<-вузол джерела живлення>*, *<аналоговий вхід>*, *<опорна напруга>*, *<«земля»>*, *<сигнал дозволу>*, *<сигнал запуску>*, *<сигнал переповнення>*, *<n-ий розряд>*, ..., *<1-й розряд>*.

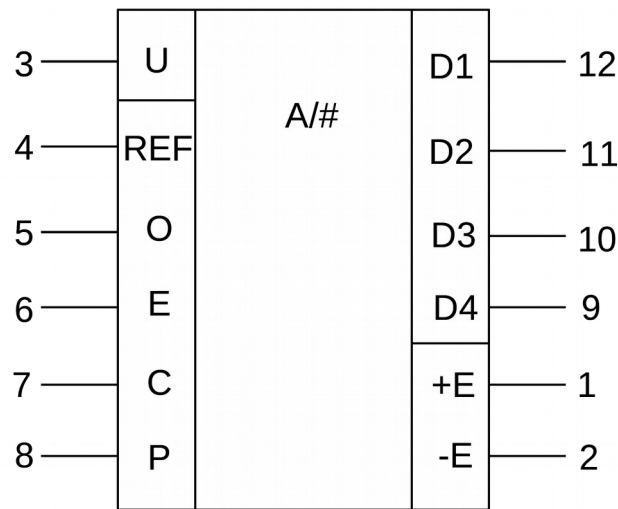


Рисунок 4.3 – Багаторозрядний АЦП

Модель динаміки має формат `.MODEL <Ім'я моделі> UADC [(параметри)]`.

Параметри моделі динаміки АЦП наведені в табл. 4.2 (значення за замовчуванням – 0, одиниця виміру – с).

Таблиця 4.2 – Параметри моделі динаміки АЦП

Ідентифікатор	Параметр
<i>TPCSMN</i>	Час затримки запуску – затримка між передніми фронтами імпульсів дозволу і запуску, мінімальне значення
<i>TPCSTY</i>	Те саме, типове значення
<i>TPCSMX</i>	Те саме, максимальне значення
<i>TPSDMN</i>	Час циклу кодування – інтервал часу між переднім фронтом імпульсу запуску і переходом вихідного сигналу в новий стан, мінімальне значення
<i>TPSDTY</i>	Те саме, типове значення
<i>TPSDMX</i>	Те саме, максимальне значення
<i>TPDSMN</i>	Запізнення заднього фронту сигналу запуску відносно моменту переходу вихідного сигналу в новий стан, мінімальне значення
<i>TPDSMTY</i>	Те саме, типове значення
<i>TPDSMX</i>	Те саме, максимальне значення

Часова діаграма АЦП показана на рис. 4.4.

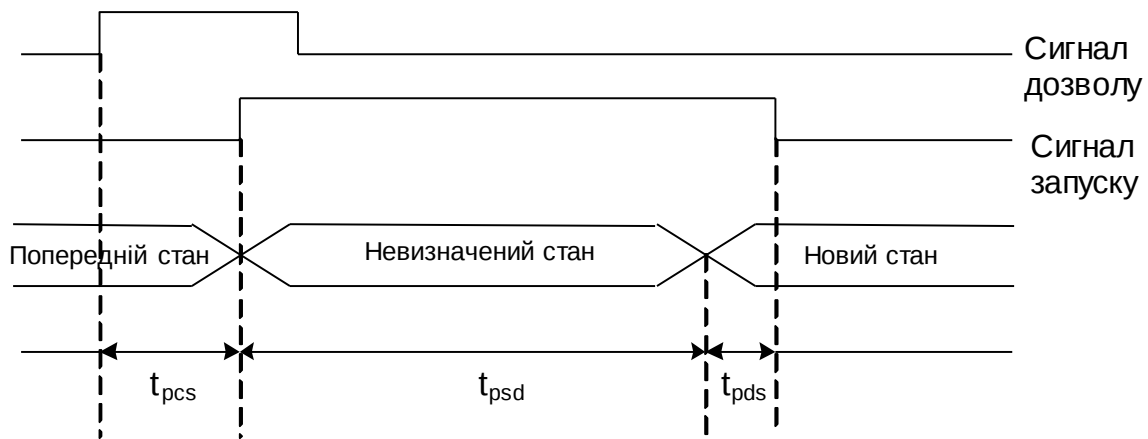


Рисунок 4.4 – Часова діаграма АЦП

Вихідний сигнал АЦП дорівнює найближчому цілому значенню відповідно до виразу:

$$[V(\text{<аналоговий вхід>}) / V(\text{<опорна напруга>})] * 2^m,$$

де  $m$  – кількість розрядів.

Якщо це значення більше за  $2^m - 1$ , всі розряди даних і розряд переповнення набудуть значення 1. Якщо воно менше нуля, розряди даних набудуть нульового значення, а розряд переповнення – 1. Таким чином, опорна напруга встановлює діапазон вхідної напруги АЦП.

Сигнал дозволу (convert pulse) може мати будь-яку, навіть і нульову тривалість. Якщо цикл кодування  $t_{psd} = 0$ , то  $m$  розрядів даних і розряд переповнення, не набуваючи невизначеного стану, одразу набувають нового значення. Між вузлами <опорна напруга> і <«земля»> підключається резистор з опором  $1/GMIN$ .

Вибірki вхідних напруг відбираються за переднім фронтом імпульсу дозволу, причому швидкість зміни вхідних напруг не впливає на результат перетворення.

Наведемо приклад опису 4-розрядного АЦП:

```
U3 ADC (4) $G_DPWR G_DGND 1 10 0 conv stat over out3 out2 out1 out0
DINAM IO_ADC
```

```
.MODEL DINAM UADC (+ Tpcsmn = 5ns, tpcsty = 8ns, tpcsmx = 10ns, +
Tpsdmn = 16ns, tpsdty = 20ns, tpsdmx = 22ns, + Tpdsmn = 4ns, tpdsty = 5ns,
tpdsmx = 6ns)
```

```
.MODEL IO_ADQ UIO (drvh = 50 drvl = 50)
```

Багаторозрядний ЦАП (рис. 4.5) задається у форматі:  $U_{xxx} DAC (<m>)$   $<список\ вузлів>$   $<модель\ динаміки>$  +  $<модель\ вхід\ /\ вихід>$  +  $[MNTYMXDLY = <вибір\ значення\ затримки>]$  +  $[IO\_LEVEL=<рівень\ моделі\ інтерфейсу>]$ .

Імена вузлів перераховуються в  $<списку\ вузлів>$  в такому порядку:  $<+ \text{вузол джерела живлення}>$ ,  $<- \text{вузол джерела живлення}>$ ,  $<аналоговий\ вихід>$ ,  $<опорна\ напруга>$ ,  $<«земля»>$ ,  $<t\text{-й розряд вхідного сигналу}>$ , ...,  $<1\text{-й розряд вхідного сигналу}>$ .

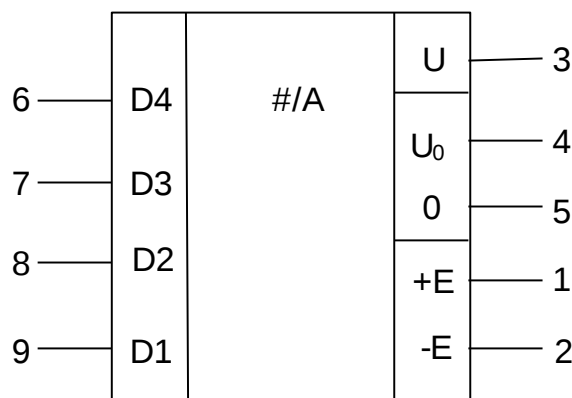


Рисунок 4.5 – Багаторозрядний ЦАП

Модель динаміки ЦАП має вигляд  $.MODEL <Ім'я\ моделі> UDAC$   $[(параметри)]$ . Параметри моделі динаміки ЦАП наведені в табл. 4.3 (значення за замовчуванням – 0, одиниця виміру – с).

Таблиця 4.3 – Параметри моделі динаміки ЦАП

Ідентифікатор	Параметр
$TSWMN$	Час встановлення (час від моменту зміни вхідного коду до моменту досягнення вихідною напругою рівня 0,9 від встановленого значення), мінімальне значення
$TSWTY$	Те саме, типове значення
$TSWMX$	Те саме, максимальне значення

Опорна напруга визначає діапазон вихідної аналогової напруги. Між вузлом джерела опорної напруги і «землею» підключається опір  $1/GMIN$ .

Якщо який-небудь розряд вхідного цифрового сигналу не визначений, вихідна напруга дорівнює половині різниці двох напруг. Одна з них являє



собою вихідну напругу ЦАП, якщо всі невизначені стани «X» замінити на «1», друга – якщо ці стани замінити на логічний «0». Це характеризує перехідні процеси в ЦАП (рис. 4.6).

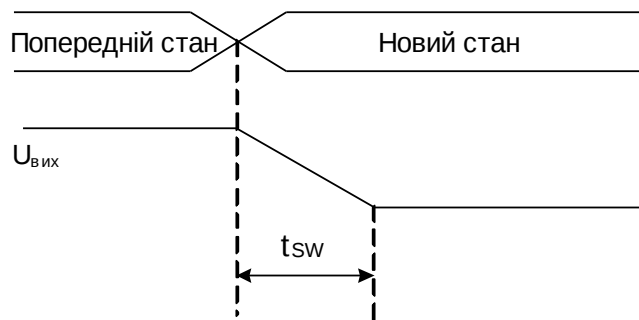


Рисунок 4.6 – Перехідні процеси в ЦАП

При зміні станів всіх розрядів вихідна напруга лінійно змінюється протягом інтервалу перетворення [13].

#### 4.4 Вентилі

Вентилі підрозділяються на елементарні і складні. Елементарні вентилі мають один або кілька входів і тільки один вихід. Складні вентилі (збірки) містять в одному корпусі декілька простих вентилів (рис. 4.7).

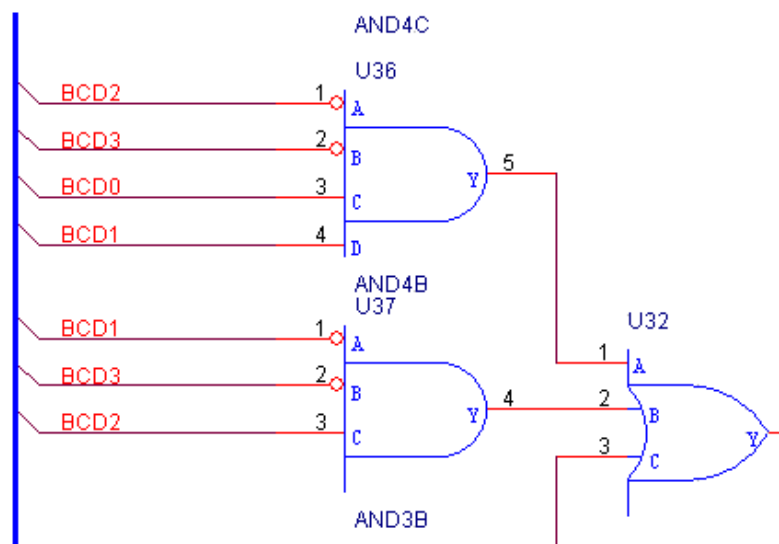


Рисунок 4.7 – Складні вентилі (збірки)

Крім того, вентилі підрозділяються на два типи: стандартні вентилі і вентилі з трьома станами. Вентилі з трьома станами управляються

сигналами дозволу. Коли цей сигнал має рівень «0», вихідний сигнал вентиля має невизначений рівень «X» при високому вихідному опорі Z.

Параметри моделей стандартних вентилів перераховані в табл. 4.4.

Таблиця 4.4 – Параметри моделей стандартних вентилів

Тип	Параметр	Порядок перерахування	Функціональне призначення
<i>BUF</i>	немає	Вх., вих.	буфер
<i>INV</i>	немає	Вх., вих.	інвертор
<i>AND</i>	<i>N</i>	Вх.1, вх.2, ..., вих.	логічне І
<i>NAND</i>	<i>N</i>	Вх.1, вх.2, ..., вих.	логічне І-НЕ
<i>OR</i>	<i>N</i>	Вх.1, вх.2, ..., вих.	логічне АБО
<i>NOR</i>	<i>N</i>	Вх.1, вх.2, ..., вих.	логічне АБО-НЕ
<i>XOR</i>	немає	Вх.1, вх.2, вих.	виключне АБО
<i>NXOR</i>	немає	Вх.1, вх.2, вих.	виключне АБО-НЕ
<i>BUFA</i>	<i>L</i>	Вх.1, вх.2, ..., вих.1, вих. 2, ...	збірка буферів
<i>INVA</i>	<i>L</i>	Вх.1, вх.2, ..., вих.1, вих. 2, ...	збірка інверторів
<i>ANDA</i>	<i>N, L</i>	Вх.1, вх.2, ..., вих.1, вих. 2, ...	збірка логіки І
<i>NANDA</i>	<i>N, L</i>	Вх.1, вх.2, ..., вих.1, вих. 2, ...	збірка логіки І-НЕ
<i>ORA</i>	<i>N, L</i>	Вх.1, вх.2, ..., вих.1, вих. 2, ...	збірка логіки АБО
<i>NORA</i>	<i>N, L</i>	Вх.1, вх.2, ..., вих.1, вих. 2, ...	збірка логіки АБО-НЕ
<i>XORA</i>	<i>L</i>	Вх.1, вх.2, ..., вих.1, вих. 2, ...	збірка логіки виключне АБО
<i>NXORA</i>	<i>L</i>	Вх.1, вх.2, ..., вих.1, вих. 2, ...	збірка логіки виключне АБО-НЕ
<i>AO</i>	<i>N, L</i>	Вх.1, вх.2, ..., вих.	збірка логіки І-АБО
<i>OA</i>	<i>N, L</i>	Вх.1, вх.2, ..., вих.	збірка логіки АБО-І
<i>OAI</i>	<i>N, L</i>	Вх.1, вх.2, ..., вих.	збірка логіки АБО-І-НЕ
<i>AOI</i>	<i>N, L</i>	Вх.1, вх.2, ..., вих.	збірка логіки І-АБО-НЕ

Примітка. *N* – кількість входів, *L* – кількість вентилів.

Модель динаміки вентилів має формат *.MODEL <Ім'я моделі> UGATE [{параметри}]*. Параметри моделей стандартних вентилів типу *UGATE* наведені в табл. 4.5 (значення за замовчуванням – 0, одиниця виміру – с).

Таблиця 4.5 – Параметри моделей стандартних вентилів

Ідентифікатор	Параметр
---------------	----------

<i>TPLHMN</i>	Затримка при переході від низького рівня до високого, мінімальне значення
<i>TPLHTY</i>	Те саме, типове значення
<i>TPLHMX</i>	Те саме, максимальне значення
<i>TPHLMN</i>	Затримка при переході від високого рівня до низького, мінімальне значення
<i>TPHLY</i>	Те саме, типове значення
<i>TPHLMX</i>	Те саме, максимальне значення

Вентилі з трьома станами перераховані в табл. 4.6.

Таблиця 4.6 – Вентилі з трьома станами

Тип	Параметр	Порядок перерахування виводів
<i>BUF3</i>	немає	Вх., розр., вих.
<i>INV3</i>	немає	Вх., розр., вих.

Вентилі з трьома станами є буфером *BUF3* та інвертором *INV3* [12].

#### 4.5 Пристрої живлення

У програмі PSpice прийнято, що макромодель джерела живлення має ім'я *DIGIFPWR* і на неї автоматично робиться посилання за наявності в схемі макромоделей аналого-цифрових інтерфейсів. Конкретний зміст макромоделі джерела живлення визначається користувачем, а її текст поміщається в бібліотеку цифрових пристроїв. Наведемо макромодель джерела напруги, що набуває за замовчуванням значення 5 В:

```
.subckt DIGIFPWR AGND optional: DPWR=$G_DPWR DGND=$G_DGND
+ params: VOLTAGE=5.0v REFERENCE=0v VDPWR DPWR DGND
{VOLTAGE} R1 DPWR AGND 1MEG VDGND DGND AGND
{REFERENCE} R2 DGNDAGND 1MEG
.ends
```

Тут вузлам цифрового живлення і цифрової «землі» присвоєно глобальні імена \$ *G\_DPWR*, \$ *G\_DGND* (за бажання їх можна змінити тут і одночасно в моделях всіх цифрових пристроїв). Вузол *AGND* – загальний вузол джерела живлення, який при підключенні в загальну схему з'єднується з вузлом аналогової «землі» 0. Для створення макромоделі джерела іншої напруги необхідно в завдання на моделювання (у файлі \*.*CIR*) внести виклик макромоделі джерела живлення, задавши за

допомогою параметра *VOLTAGE* потрібне значення напруги живлення, і вказати після номера вузла «землі» імена (номери) вузлів, наприклад: *XYPOWER O MY\_PWR MY\_GND DIGIFWR params: VOLTAGE=9.0v*.

Тут викликається джерело напруги 9 В *MY\_PWR* і *MY\_GND*. Ускладнивши макромодель, можна створити джерело декількох напруг [12].

## 4.6 Генератори цифрових сигналів

Генератори цифрових сигналів можна задати двома способами: визначення форми цифрового сигналу за форматом (пристрої *STIM*) та зчитування тимчасової діаграми цифрового сигналу з файлу (пристрої *FSTREAM*).

### 4.6.1 Визначення форми цифрового сигналу за форматом (пристрої *STIM*)

Визначення форми цифрового сигналу в завданні на моделювання за форматом (пристрої *STIM*): *Uxxx STIM* (<кількість сигналів>, <формат>) + <+ Вузол джерела живлення> <-вузол джерела живлення> + <Список вузлів> \* <ім'я моделі вхід / вихід> + [*IO\_LEVEL*- <номер макромоделі інтерфейсу вхід / вихід>] + [*STIMULUS* = <ім'я впливу>] [*TIMESTEP* = <крок за часом>] + <Команди опису форми сигналу> \*.

Змінна <кількість сигналів> дорівнює кількості генерованих різних цифрових сигналів.

Змінна <формат> – це специфікація формату змінної <дані>, в якій наведено логічні рівні сигналів генератора. Ця змінна є послідовністю цифр, загальне число яких дорівнює значенню змінної <кількість сигналів>. Кожна цифра набуває значення 1, 3 або 4, що означає двійкову, вісімкову і шістнадцяткову систему числення.

Підключення джерела живлення задається номерами вузлів <+ вузол джерела живлення>, <-вузол джерела живлення>.

Номери підключення виходів генератора до схеми задаються <списком вузлів>.

Ім'я моделі вхід/вихід задається параметром <ім'я моделі вхід / вихід>, як для будь-якого цифрового пристрою.

*STIMULUS* – необов'язковий параметр, що позначає ім'я сигналу.

*IO\_LEVEL* – необов’язковий параметр для вибору однієї з чотирьох макромоделей інтерфейсу вхід/вихід (за замовчуванням 0).

*TIMESTEP* – необов’язковий параметр для задання періоду квантування (або кроку). При заданні моментів часу номером кроку (мають суфікс «C») він домножається на величину кроку. За замовчуванням встановлюється *TIMESTEP*=0. Цей параметр не береться до уваги, якщо задано абсолютні значення моментів часу (мають суфікс «S»).

Параметр <опис форми-сигналу> являє собою довільну комбінацію одного або декількох рядків:

<t>, <логічний рівень>

*LABEL* = <UMH мітки>

<t> *GOTO* <ім’я мітки> <n> *TIMES*

<t> *GOTO* <ім’я мітки> *UNTIL GT* <дані>

<t> *GOTO* <ім’я мітки> *UNTIL GE* <дані>

<t> *GOTO* <ім’я мітки> *UNTIL LT* <дані>

<t> *GOTO* <ім’я мітки> *UNTIL LE* <дані>

<t> *INCR BY* <дані>

<t> *DECR BY* <дані>

*REPEAT FOREVER*

*REPEAT* <n> *TIMES*

*ENDREPEAT*

*FILE* = <ім’я файлу>

Кожна цифра змінної <дані> являє собою логічний рівень відповідного вихідного сигналу, який наведений в системі числення  $2m$ , де  $m$  – відповідна цифра змінної <формат>.

Кількість вузлів в <списку вузлів> має дорівнювати значенню змінної <кількість сигналів>.

Змінна <t> визначає моменти часу, в які задаються логічні рівні сигналу. Якщо перед значенням змінної <t> є символ «+», то ця змінна задає прирощення відносно попереднього моменту часу, в іншому випадку вона визначає абсолютне значення відносно початку відліку часу  $t=0$ . Суфікс «S» вказує розмірність часу в секундах (допускається суфікс «nS» – наносекунди і т. п.). Суфікс «C» означає вимір часу в кількості циклів, розмір яких визначається параметром *TIMESTEP* (змінна <крок за часом>).

Змінна <дані> складається із символів «0», «1», «X», «R», «F» або «Z», інтерпретованих у заданому форматі.

Змінна  $\langle n \rangle$  задає кількість повторюваних циклів *GOTO*, причому значення  $n=-1$  задає нескінченне повторення циклу.

Змінна  $\langle \text{ім'я мітки} \rangle$  використовується при організації циклу за допомогою оператора переходу *GOTO*, який передає управління на рядок за оператором *LABEL* =  $\langle \text{ім'я мітки} \rangle$ .

*REPEAT FOREVER* – початок нескінченно повторюваного циклу (еквівалентн конструкції *REPEAT -1 TIMES*).

*REPEAT*  $\langle n \rangle$  *TIMES* – повторення  $n$  раз.

*ENDREPEAT* – кінець циклу *REPEAT*.

*FILE* – вказання імені файлу, в якому знаходиться опис одного або декількох вхідних сигналів.

Наведемо приклад:

```
USIGNAL STIM (2,11) $ G_DPWR $ G_DGND 1 2 IO_STM TIMESTEP = 1ns
```

```
+ OC 00; у момент часу  $t = 0$  стани обох вузлів дорівнюють «0»
```

```
+ LABEL = M
```

```
+ 1C 01; при  $t = 1$  не стан вузла 1 дорівнює «0», вузла 2 – «1»
```

```
+ 2C 11; при  $t = 2$  не стани обох вузлів дорівнюють «1»
```

```
+ 3C GOTO M 3 TIMES; у момент часу  $t = 3$  не перехід на мітку M і виконується перша інструкція без додаткової затримки
```

Цикл повторюється 3 рази [12].

#### 4.6.2 Зчитування тимчасової діаграми цифрового сигналу з файлу (пристрої FSTREAM)

Пристрої FSTIM задаються за форматом:

```
Uxxx FSTIM ( $\langle \text{кількість виходів} \rangle$ )  $\langle +$  вузол джерела живлення  $\rangle$ 
```

```
+  $\langle -$ вузол джерела живлення  $\rangle$   $\langle \text{список вузлів} \rangle$  *
```

```
+  $\langle \text{Ім'я моделі вхід / вихід} \rangle$  FILE =  $\langle \text{ім'я файлу сигналів} \rangle$ 
```

```
+ [IO_LEVEL =  $\langle \text{номер макромоделі інтерфейсу вхід/вихід} \rangle$ ]
```

```
+ [SIGNAMES =  $\langle \text{ім'я впливу} \rangle$ ].
```

Параметр  $\langle \text{кількість виходів} \rangle$  визначає кількість вузлів, до яких підключаються сигнали. Варто звернути увагу, що задання за допомогою параметра *FILE* імені файлу, в якому описані сигнали, є обов'язковим.

Файл цифрового сигналу може бути отриманий шляхом редагування файлу результатів моделювання або складений вручну за допомогою будь-якого текстового редактора.

Файл цифрового сигналу має дві секції: заголовок (header), що містить список імен сигналів, та список значень сигналу (transitions), що містить на одному або більше рядках моменти часу змін сигналів і колонки їх значень.

Наведемо спочатку приклад файлу цифрових сигналів:

*Заголовок; містить імена сигналів CLOCK, RESET, IN1, IN2*

*\* Початок опису сигналів; відділяється від заголовка пустим рядком*

*0 0000; двійкові коди сигналів*

*10ns 1100*

*20ns 0101*

*30ns 1110*

*40ns 0111*

Заголовок має такий формат:

[TIMESCALE=<значення>]<ім'я сигналу 1>...< ім'я сигналу n>

OCT(<3-й біт сигналу> ... <1-й біт сигналу>) ...

HEX(<4-й біт сигналу> ... <1-й біт сигналу>) ...

Імена сигналів можуть розділятися комами або пропусками. Вони розташовуються на одному або декількох рядках, але рядки продовження не мають на початку знака «+». Максимальна кількість сигналів 255. На одному рядку можуть розміщуватися не більше 300 символів. Імена сигналів перераховуються в тому ж порядку, в якому їх значення наводяться в тимчасових діаграмах.

Якщо перед ім'ям сигналу не вказується назва системи числення, він вважається двійковим. Імена сигналів, записаних у вісімковому коді, наводяться за допомогою коду OCT і групуються по 3 сигнали (імена груп сигналів беруть в круглі дужки). Імена шістнадцяткових сигналів мають код HEX і групуються по 4 сигнали.

Наведемо ще один приклад файлу сигналів:

*Clock Reset In1 In2*

*HEX (Addr7 Addr6 Addr5 Addr4) HEX (Addr3 Addr2 Addr1 Addr0)*

*ReadWrite*

*0 0000 00 0; для наочності між групами можна вставляти пробіли*

*10n 1100 4E 0*

*20n 0101 4E 1*

*30n 1110 4E 1*

*40n 0111 FF 0*

Тут спочатку наведено імена чотирьох двійкових сигналів, потім дві групи сигналів, що задаються в шістнадцятковому вигляді, і потім ще один бінарний сигнал. У подальшому списку значень сигналів подано 7 колонок, відповідних наведеним у заголовку сигналам.

**Список значень** має формат:

<Час> <значення сигналу> \*.

Список значень відділяється від заголовка пустим рядком. Моменти часу і список значень сигналів мають бути розділені принаймні одним пропуском. Для моментів часу змін сигналів вказуються їх абсолютні значення (в секундах), наприклад  $25ns$ ,  $1.2E-9$ ,  $5.8$  або приріст відносно попереднього моменту часу. Ознака збільшення – знак «+» перед значенням моменту часу, наприклад  $+5ns$ . Кожне значення сигналу відповідає одиничному бінарному сигналу або групі сигналів, імена яких занесені до груп з покажчиками системи числення ОСТ або HEX. Загальна кількість значущих цифр у значеннях сигналів має дорівнювати загальній кількості бінарних сигналів і груп сигналів, позначених символами ОСТ і HEX. Розряди даних сигналу можуть набувати значень, наведених в табл. 4.7.

Таблиця 4.7 – Значення розрядів сигналів

Сигнали	Двійкові	Вісімкові	Шістнадцяткові
Логічні / цифрові	0, 1	0...7	0...F
Невизначені	X	X	X
Високого імпедансу	Z	Z	Z
Передній фронт	R	R	–
Задній фронт	F	F	–

Зауважимо, що задній фронт не може в шістнадцятковій системі позначатися як  $F$ , тому що цей символ зайнятий під чисельне значення. Наведемо приклади генераторів цифрових сигналів типу *FSTIM*.

Приклад 1. Нехай є файл *dig1.stm*, в якому описана тимчасова діаграма сигналу *IN1*. Тоді до завдання на моделювання можна внести такий опис генератора:

```
U1 FSTIM (1) $G_DPWR $G_DGND IN1 IO_STM FILE = DIG1.STM
```



Приклад 2. Нехай є файл flipflop.stm такого змісту:

```
J K PRESET CLEAR CLOCK 0 0 0 010  
10ns 0 0 111
```

До завдання на моделювання внесемо опис генератора сигналу  
*U2 FSTIM(4) \$G\_DPWR \$G\_DGND CLK PRE J K IO\_STM + FILE=flipflop.stm  
SIGNAMES = CLOCK PRESET.*

У цьому прикладі першим двом вузлам *CLK* і *PRE* відповідають сигнали *CLOCK* і *PRESET* з файлу сигналів, це відповідність встановлюється за допомогою опції *SIGNAMES*. Останнім вузлам *J* і *K* відповідають однойменні сигнали з файлу сигналів, тому їх імена немає необхідності вносити до опції *SIGNAMES*. Сигнал *CLEAR* в цьому прикладі не використовується (але на нього можна послатися в іншому генераторі *FSTIM*) [12].

#### 4.7 Програмовані логічні матриці

Програмовані логічні матриці (ПЛМ, PLC – Programmable Logic Arrays) мають ряд входів, які формують стовпці матриці, і ряд виходів, що утворюють рядки. Кожен вихід (рядок) керується одним логічним елементом. Сукупність керуючих сигналів складають програму для ПЛМ, яка визначає, які входи з'єднуються з логічними елементами. До складу примітивів ПЛМ входять тільки однотипні вентиля (І, АБО, І-НЕ і т. п.), тому реальні ІС ПЛМ складаються з декількох примітивів у вигляді макромоделей.

Програма ПЛМ вводиться в завдання на моделювання двояко:

- попередньо записавши її у файл у форматі *JEDEC* і вказавши його ім'я в описі примітиву конкретної ПЛМ;
- внісши дані програми безпосередньо в опис ПЛМ (використовуючи конструкцію *DATA = ...*), що менш зручно.

Однак при використанні бібліотек, в яких ПЛМ оформлені у вигляді макромоделей, користувачеві не потрібно розбиратися в деталях їх моделей – досить вказати ім'я моделі ПЛМ, список вузлів включень, за допомогою опції *TEXT* вказати ім'я *JEDEC*-файлу, що містить опис програми ПЛМ, як показано в прикладі:

```
X1 IN1 IN2 IN3 IN4 IN5 IN6 IN7 IN8 IN9 IN10 IN11 IN12 IN13 IN14  
+ OUT1 OUT2 OUT3 OUT4 PAL14H4 TEXT: JEDEC_FILE = «myprog.jed».
```

Тут в схему підключена ПЛМ типу *PAL14H4*, програмована з файлу *murprog.jed*. Примітиви ПЛМ вводяться до завдання на моделювання процедурою:

*Uxxx* <тип ПЛМ> (<кількість входів>, <кількість виходів>)  
+ <+вузол джерела живлення> <-вузол джерела живлення>  
+ <вхідний вузол> \* <вихідний вузол> \*  
+ <ім'я моделі динаміки> <ім'я моделі вхід / вихід>  
+ [*FILE* = <ім'я файлу>]  
+ [*DATA* = <прапор системи числення> <дані програми> \$]  
+ [*MNTYMXDLY* = <вибір значення затримки>]  
+ [*IO\_LEVEL* = <рівень моделі інтерфейсу>].

Існують ПЛМ таких типів:

- *PLAND* – матриця логіки І;
- *PLOR* – матриця логіки АБО;
- *PLXOR* – матриця логіки виключне АБО;
- *PLNAND* – матриця логіки І-НЕ;
- *PLNOR* – матриця логіки АБО-НЕ;
- *PLNXOR* – матриця логіки виключне АБО-НЕ;
- *PLANDC* – матриця логіки І, що містить для кожного входу стовпці прямого і додаткового коду;
- *PLORC* – матриця логіки АБО, що містить для кожного входу стовпці прямого і додаткового коду;
- *PLXORC* – матриця логіки виключне АБО, що містить для кожного входу стовпці прямого і додаткового коду;
- *PLNANDC* – матриця логіки І-НЕ, що містить для кожного входу стовпці прямого і додаткового коду;
- *PLNORC* – матриця логіки АБО-НЕ, що містить для кожного входу стовпці прямого і додаткового коду;
- *PLNXORC* – матриця логіки **виключне** АБО, що містить для кожного входу стовпці прямого і додаткового коду.

Після ключового слова *FILE* вказується ім'я файлу у форматі *JEDEC*, в якому записана програма ПЛМ.

Воно може бути зазначено як текстова константа (і тоді **береться** в лапки «») або як тек**стовий** вираз (розташовується між вертикальними рисками | |).

Якщо наведено опцію *FILE*, то будь-які дані, наведені після опції *DATA*, ігноруються.

Спосіб адресації, прийнятий у файлі *JEDEC* для розташування даних, визначається параметрами моделі динаміки.

Файл системи числення набуває значень:

- *B* – двійкова система числення;
- *O* – вісімкова система числення (біт старшого розряду розташований за молодшою адресою);
- *X* – шістнадцяткова система числення (біт старшого розряду розташований за молодшою адресою).

Дані для програми записуються між знаками долара \$ і можуть розташовуватися як разом, так і розділитися одним або декількома пропусками.

Потік даних починається з нульової адреси, за якою програмується з'єднання першого входу з вентилем, керуючим першим виходом.

Наявність «0» означає, що вхід не підключений до вентиля, а «1» – наявність такого з'єднання (спочатку всі входи не під'єднані до жодних виходів).

Дані за наступною адресою керують з'єднанням доповнення першого входу до вентиля, керуючого першим виходом (тільки для ПЛМ з прямими і додатковими входами), або другого входу, підключеного до вентиля, керуючого першим виходом.

Кожна додаткова «1» або «0» програмує з'єднання входу або його доповнення з вентилем, керуючим першим виходом, до тих пір, поки не переберуть всі входи (і їх доповнення).

Наступні дані програмують з'єднання входів з другим виходом і т. д.

Модель динаміки ПЛМ має формат

*MODEL* <ім'я моделі динаміки> *UPLD* {(параметри моделі динаміки)}.

Параметри моделі динаміки ПЛМ наведено в табл. 4.8.

Таблиця 4.8 – Параметри моделі динаміки ПЛМ

Ідентифікатор	Параметр	Значення за замовчуванням	Одиниця виміру
<i>TPLHMN</i>	Час затримки на вихід при перемиканні 0 -> 1, мінімальне значення	0	с
<i>TPLHTY</i>	Te same, типове значення	0	с
<i>TPLHMX</i>	Te same, максимальне значення	0	с

<i>TPHLMN</i>	Час затримки на вихід при перемиканні 1 -> 0, мінімальне значення	0	c
<i>TPHLTY</i>	Те саме, типове значення	0	c
<i>TPHLMX</i>	Те саме, максимальне значення	0	c
<i>OFFSET</i>	Адреса даних, керуючих підключенням першого входу до першого виходу (у файлі <i>JEDEC</i> )	0	–
<i>COMPOFFSET</i>	Адреса даних, керуючих підключенням доповнення першого входу до першого виходу (у файлі <i>JEDEC</i> )	1	–
<i>INSCALE</i>	Кількість адрес для програмування зміни стану кожного входу (у файлі <i>JEDEC</i> )	1	–
<i>OUTSCALE</i>	Кількість адрес у файлі <i>JEDEC</i> для програмування зміни стану кожного виходу (вентиля)	2	–

Наведемо приклад дешифратора 3×8, реалізованого на ПЛМ (рис. 4.7).

Вхідні вузли позначимо *IN1* (старший розряд), *IN2*, *IN3* (молодший розряд). Якщо всі входи знаходяться в стані «0», вихід *OUT1* = «1». Якщо *IN1* і *IN2* – у стані «0», а *IN3* – у стані «1», *OUT2* = «1» і т. д. Дані програми для зручності читання записані у вигляді масиву. У коментарях зверху від програми вказані імена вхідних вузлів, що знаходяться в стані «1» – *true* (*T*) і «0» – *false* (*F*, додатковий код). В коментарях в кінці рядків вказані імена вихідних вузлів, керованих вентилям.

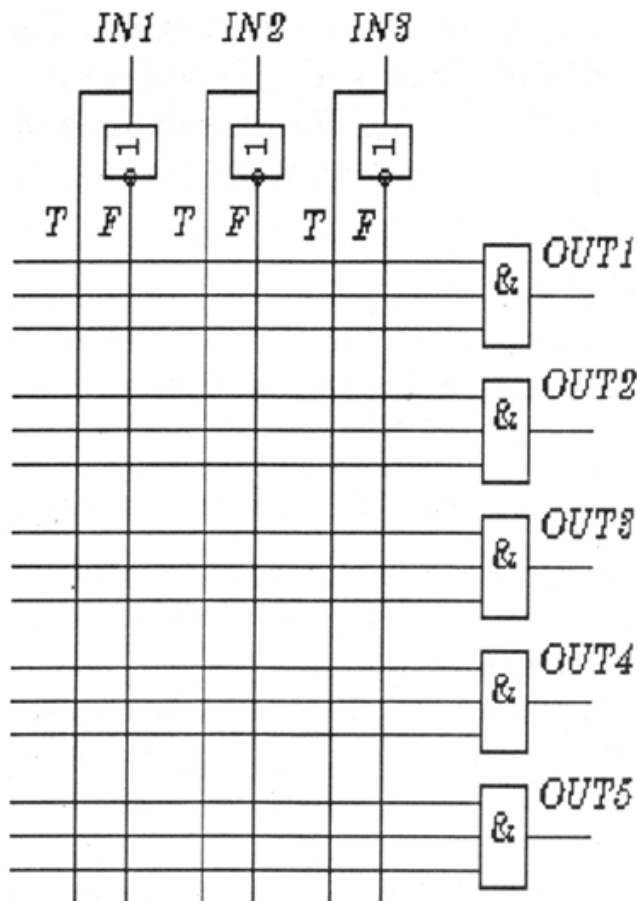


Рисунок 4.7 – Дешифратор 3×8, реалізований на ПЛМ

*UDECODER PLANDC (3,8); 3 входи, 8 виходів*

*+ \$ G\_DPWR \$ G\_DGND; вузли джерела живлення і «землі»*

*+ IN1 IN2 IN3; входи*

*+ OUT1 OUT2 OUT3 OUT4 OUT5 OUT6 OUT7 OUT8; виходи*

*+ PLD\_MDL; ім'я моделі динаміки ПЛМ*

*+ IO\_STD; ім'я моделі вхід / вихід ПЛМ*

*+ DATA = B\$; дані програми ПЛМ*

*\*IN1 IN2IN3*

*\*TF TF TF*

*+ 01 01 01 ; OUT1*

*+ 01 01 10; OUT2*

*+ 01 1001 ; OUT3*

*+ 01 1010; OUT4*

*+ 1001 01 ; OUT5*

*+ 1001 10; OUT6*

*+ 101001 ; OUT7*

+ 10 10 10\$; OUT8

.MODEL PLD\_MDL UPLD (...); визначення моделі динаміки ПЛМ

## 4.8 Запам'ятовувальні пристрої

Запам'ятовувальні пристрої (ЗП) поділяються на постійні (ROM, Read Only Memories) і оперативні (RAM, Random Access Read-Write Memories) [12].

### 4.8.1 Постійні запам'ятовувальні пристрої (ПЗП)

Існує два способи запису даних в ПЗП для подальшого моделювання:

- зазвичай дані попередньо записуються у файл у форматі *Intel Hex* і перед початком моделювання вони з нього зчитуються;
- дані записуються безпосередньо в описі ПЗП за допомогою конструкції *DATA = ...*

ПЗП (рис. 4.8) вводиться до завдання на моделювання так:

*Uxxx ROM* (<кількість адресних входів>, <кількість виходів>) + <+вузол джерела живлення> <-вузол джерела живлення> + <вхід дозволу читання>

+ <старший розряд адреси> . . . <молодший розряд адреси> + <старший розряд виходу> .. <молодший розряд виходу> +, <ім'я моделі динаміки> <ім'я моделі вхід / вихід> - + [*FILE =* <ім'я файлу>]

+ [*DATA =* <файл системи числення> \$<дані програми> \$] + [*MNTYMXDLY =* <вибір значення затримки>] + [*IO\_LEVEL =* <рівень моделі інтерфейсу>]

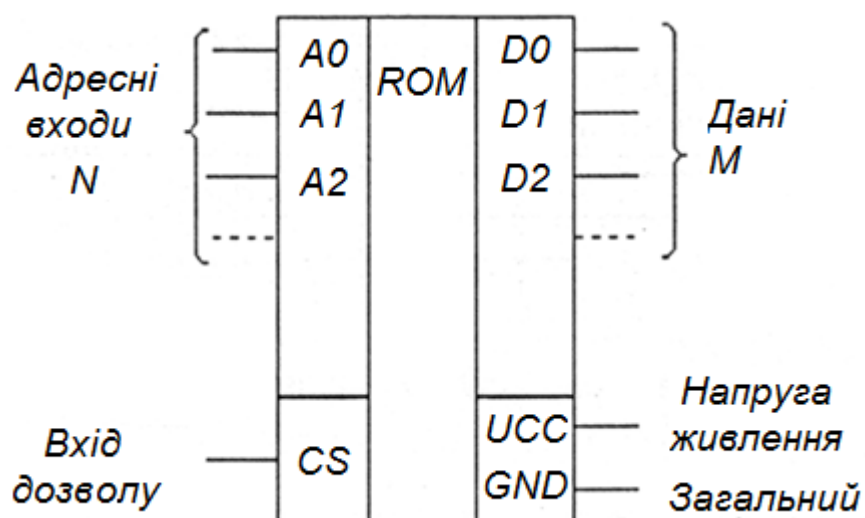


Рисунок 4.8 – Постійний запам'ятовувальний пристрій

Після ключового слова *FILE* вказується ім'я файлу у форматі *Intel Hex*, в якому записані дані ПЗП. Воно може бути зазначено як текстова константа (і тоді береться в лапки «») або як текстовий вираз (розміщується між вертикальними рисками | |). Якщо наведено опцію *FILE*, то будь-які дані, наведені після опції *DATA*, ігноруються.

Файл системи числення набуває значень:

- *B* – двійкова система числення;
- *O* – вісімкова система числення (біт старшого розряду розташований за молодшою адресою);
- *X* – шістнадцяткова система числення (біт старшого розряду розташований за молодшою адресою).

Дані до програми записуються між знаками долара \$ і можуть розташовуватися як разом, так і розділятися одним або декількома пропусками. Потік даних починається з нульової адреси, за якою розміщується перший розряд даних. Наступний біт відноситься до другого розряду даних і так до тих пір, поки не будуть визначені стани всіх розрядів за цією адресою. Після цього перераховуються дані за наступною адресою і т. д.

Модель динаміки ПЗП має формат:

*.MODEL <Ім'я моделі динаміки> UROM [(параметри моделі динаміки)]*

Параметри моделі динаміки ПЗП наведені в табл. 4.9 (значення за замовчуванням – 0, одиниця виміру – с).

Таблиця 4.9 – Параметри моделі динаміки ПЗП

Ідентифікатор	Параметр
<i>TPADHMN</i>	Час вибірки адреси при перемиканні вхідних даних 0 -> 1, мінімальне значення
<i>TPADHTY</i>	Те саме, типове значення
<i>TPADHMX</i>	Те саме, максимальне значення
<i>TPADLMN</i>	Час вибірки адреси при перемиканні вхідних даних 1 -> 0, мінімальне значення
<i>TPADLTY</i>	Те саме, типове значення
<i>TPADLMX</i>	Те саме, максимальне значення
<i>TPEDHMN</i>	Час вибірки дозволу при перемиканні виходів Z -> 1, мінімальне значення

<i>TPEDHTY</i>	Те саме, типове значення
<i>TPEDHMX</i>	Те саме, максимальне значення
<i>TPEDLMN</i>	Час вибірки дозволу при перемиканні виходів $Z \rightarrow 0$ , мінімальне значення
<i>TPEDLTY</i>	Те саме, типове значення
<i>TPEDLMX</i>	Те саме, максимальне значення
<i>TPEDHZMN</i>	Час вибірки дозволу при перемиканні виходів $1 \rightarrow Z$ , мінімальне значення
<i>TPEDHZTY</i>	Те саме, типове значення
<i>TPEDHZMX</i>	Те саме, максимальне значення
<i>TPEDLZMN</i>	Час вибірки дозволу при перемиканні виходів $0 \rightarrow Z$ , мінімальне значення
<i>TPEDLZTY</i>	Те саме, типове значення
<i>TPEDLZMX</i>	Те саме, максимальне значення

Перехідні процеси під час зчитування з ПЗП показані на рис. 4.9. На вхід дозволу читання необхідно подати «1», а стани вузлів вихідних даних змінюються від стану високого імпедансу  $Z$  до відповідного стану через деякий час  $t_{PED}$ . Протягом часу, поки сигнал дозволу читання знаходиться в стані «1», сигнали адреси можуть змінюватися, і якщо це так, то нові дані доступні на виходах через деякий час затримки  $t_{PAD}$ .

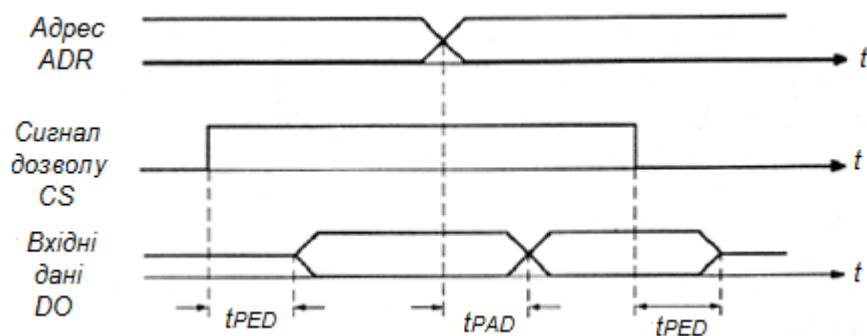


Рисунок 4.9 – Перехідні процеси в ПЗП

Наведемо приклад ПЗУ 8×8:

- UMULTIPLY ROM (8, 8); модель ПЗУ 256 8 разів рядів*
- + \$G\_DPWR \$G\_DGND; вузли джерела живлення і «землі»*
- + ENABLE; вхід дозволу читання*
- + AIN3 AIN2 AIN1 AINO; перші 4 біти адреси*
- + BIN3 BIN2 BIN1 BINO; другі 4 біти адреси*



+ *OUT?* *OUT6 OUT5 OUT4 OUT3 OUT2 OUT1 OUT0*; виходи  
 + *ROM\_MDL*; ім'я моделі динаміки  
 + *IO\_STD*; ім'я моделі вхід/вихід  
 + *DATA = X\$*

Дані в шістнадцятковому коді: \* 01 23456789ABCDEF  
 + 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ;  
 A=0 + 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F ;  
 A=1 + 00 02 04 06 08 0A 0C 0E 10 12 14 16 18 1A 1C 1E ;  
 A=2 + 00 03 06 09 0C 0F 12 15 18 1B 1E 21 24 27 2A 2D ;  
 A=3 + 00 04 08 0C 10 14 18 1C 20 24 28 2C 30 34 38 3C ;  
 A=4 + 00050A0F 14 19 1E 23 28 2D 32 37 3C 41 464B ;  
 A=5 + 00 06 0C 12 18 1E 24 2A 30 36 3C 42 48 4E 54 5A ;  
 A=6 + 00 07 0E 15 1C 23 2A 31 38 3F 46 4D 54 58 62 69 ;  
 A=7 + 00 08 10 18 20 28 30 38 40 48 50 58 60 68 70 78 ;  
 A=8 + 00 09 12 1B 24 2D 36 3F 48 51 5A 63 6C 75 7E 87 ;  
 A=9 + 00 0A 14 1E 28 32 3C 46 50 5A 64 6E 78 82 8C 96 ;  
 A=A + 00 0B 16 21 2C 37 42 4D 58 63 6E 79 84 8F 9A A5 ;  
 A=B + 00 0C 18 24 30 3C 48 54 60 6C 78 84 90 9C A8 B4 ;  
 A=C + 00 0D 1A 27 34 41 4E 5B 68 75 82 8F 9C A9 B6 C3 ;  
 A=D + 00 0E 1C 2A 38 46 54 62 70 7E 8C 9A A8 B6 C4 D2 ;  
 A=E + 00 0F 1E 2D 3C 48 5A 69 78 87 96 A5 B4 C3 D1 E1\$ ;  
 A=F .MODEL ROM\_MDL UROM (...)

#### 4.8.2 Оперативні запам'ятовувальні пристрої (ОЗП)

Зазвичай в початковий момент часу в ОЗП встановлюються довільні дані за всіма адресами. Існує два способи запису початкових даних в ОЗП при моделюванні:

- дані попередньо записуються у файл у форматі Intel Hex і до початку моделювання з нього зчитуються;
- дані записуються безпосередньо в описі ОЗУ за допомогою конструкції *DATA = ...*

ОЗП (рис. 4.10) вводиться до завдання на моделювання так:

*Uxxx RAM* (<Кількість адресних входів>, <кількість виходів>)

+ <+вузол джерела живлення> <-вузол джерела живлення>

+ <вхід дозволу читання> <вхід дозволу запису>

+ <старший розряд адреси> ... <молодший розряд адреси>

+ <старший розряд входу даних> ... <молодший розряд входу даних>

- + <старший розряд виходу> ... < молодший розряд виходу>
- + <ім'я моделі динаміки> <ім'я моделі вхід/вихід>
- + [FILE = <ім'я файлу>]
- + [DATA = << прапор системи числення>\$<дані програми> \$]
- + [MNTYMXDLY = <вибір значення затримки>]
- + [IO\_LEVEL = <рівень моделі інтерфейсу>]...

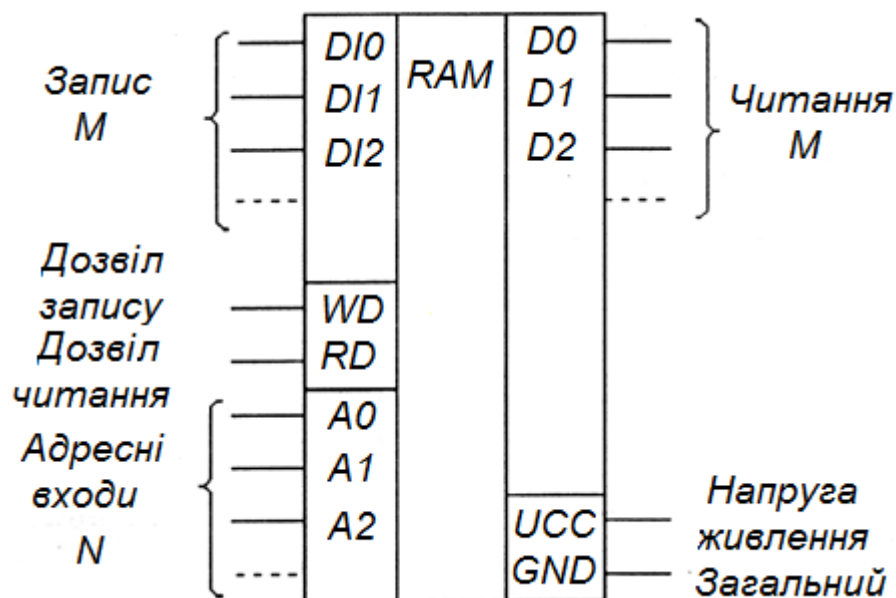


Рисунок 4.10 – Оперативний запам'ятовувальний пристрій

Після ключового слова *FILE* вказується ім'я файлу у форматі *Intel Hex*, в якому записані дані ЗП. Воно може бути зазначено як текстова константа (і тоді записується в лапки «») або як текстове вираз (укладається між вертикальними рисками | |). Якщо наведена опція *FILE*, то будь-які дані, наведені після опції *DATA*, ігноруються.

Файл системи числення набуває значень:

- *B* – двійкова система числення;
- *O* – вісімкова система числення (біт старшого розряду розташований за молодшою адресою);
- *X* – шістнадцяткова система числення (біт старшого розряду розташований за молодшою адресою).

Дані програми розташовуються між знаками долара \$ і можуть записуватися як разом, так і розділятися одним або декількома пропусками. Потік даних починається з нульового адреси, за якою розміщується перший розряд даних. Наступний біт відноситься до другого розряду даних, і так до тих пір, поки не будуть визначені стани всіх розрядів за цією адресою. Після цього перераховуються дані за наступною адресою і т. д., як і в ПЗП.

Під час записування даних в ОЗП необхідно спочатку подати сигнали на адресні входи і входи даних та не змінювати їх протягом певного часу – часу встановлення  $t_{SUAEW}$  і  $t_{SUDEW}$ , відповідно, після чого встановити «1» на вході дозволу запису. Цей сигнал повинен утримуватися протягом деякого мінімального інтервалу часу  $t_{WEW}$  і потім може бути скинутий в «0». За такої умови сигнали адреси і даних не повинні змінюватися протягом часу, коли сигнал дозволу запису знаходиться в стані «1» і утримується ще якийсь час  $t_{HAEW}$  і  $t_{HDEW}$ , перш ніж змінитися.

Для читання з ОЗП на вхід дозволу читання необхідно подати «1», після чого стану вузлів вихідних даних змінюються від стану високого імпедансу  $Z$  до відповідного стану через деякий час  $t_{PED}$ . Протягом часу, поки сигнал дозволу читання знаходиться в стані «1», сигнали адреси можуть змінюватися, і якщо це так, то нові дані доступні на виходах через деякий час затримки  $t_{PAD}$ .

У моделі ОЗП ніщо не перешкоджає одночасно встановити «1» на входах дозволу читання і запису, хоча в більшості реальних ОЗП це не допускається. Нові дані посилаються на виходи даних після переходу сигналу дозволу запису з «1» в «0».

Модель динаміки ОЗП має формат: *.MODEL <Ім'я моделі динаміки> URAM [(параметри моделі динаміки)].*

Параметри моделі динаміки ОЗП наведено в табл. 4.10

Таблиця 4.10 – Параметри моделі динаміки ОЗП

Ідентифікатор	Параметр	Ідентифікатор	Параметр
<i>TPADHMN</i>	Час вибірки адреси при перемиканні виходів 0 -> 1, мінімальне значення	<i>TSUDEWMN</i>	Час встановлення даних відносно переднього фронту сигналу дозволу, мінімальне значення
<i>TPADHTY</i>	Те саме, типове значення	<i>TSUDEWTY</i>	Те саме, типове значення
<i>TPADHMX</i>	Те саме, максимальне значення	<i>TSUDEWMX</i>	Те саме, максимальне значення
<i>TPADLMN</i>	Час вибірки адреси при перемиканні виходів 1 -> 0, мінімальне значення	<i>TSUAWMN</i>	Час встановлення адреси відносно переднього фронту сигналу дозволу, мінімальне значення
<i>TPADLTY</i>	Те саме, типове значення	<i>TSUAWTY</i>	Те саме, типове значення
<i>TPADLMX</i>	Те саме, максимальне значення	<i>TSUAWMX</i>	Те саме, максимальне значення

<i>TPERDHMN</i> 1..	Час вибірки дозволу при перемиканні виходів $Z \rightarrow 1$ , мінімальне значення	<i>TWEWHMN</i>	Тривалість сигналу дозволу при запису 1, мінімальне значення
<i>TPERDHTY</i>	Те саме, типове значення	<i>TWEWHTY</i>	Те саме, типове значення
<i>TPERDHMX</i>	Те саме, максимальне значення	<i>TWEWHMX</i>	Те саме, максимальне значення
<i>TPERDLMN</i>	Час вибірки дозволу при перемиканні виходів $Z \rightarrow 0$ , мінімальне значення	<i>TWEWLMN</i>	Тривалість сигналу дозволу при запису 0, мінімальне значення
<i>TPERDLTY</i>	Те саме, типове значення	<i>TWEWLTU</i>	Те саме, типове значення
<i>TPERDLMX</i>	Те ж саме, максимальне значення	<i>TWEWLMX</i>	Те саме ж, максимальне значення
<i>TPERDHZMN</i>	Час вибірки дозволу при перемиканні виходів $1 \rightarrow Z$ , мінімальне значення	<i>THDEWMN</i>	Час утримання вхідних даних відносно заднього фронту сигналу дозволу запису, мінімальне значення
<i>TPERDHZTY</i>	Те саме, типове значення	<i>THDEWTY</i>	Те ж саме, типове значення
<i>TPERDHZMX</i>	Те саме, максимальне значення	<i>THDEWMX</i>	Те ж саме, максимальне значення

Продовження таблиці 4.10

<i>TPERDLZMN</i>	Час вибірки дозволу при перемиканні виходів $0 \rightarrow Z$ , мінімальне значення	<i>THAEWMN</i>	Час утримання адреси відносно заднього фронту сигналу дозволу запису, мінімальне значення
<i>TPERDLZTY</i>	Те саме, типове значення	<i>THAEWTY</i>	Те саме, типове значення
<i>TPERDLZMX</i>	Те саме, максимальне значення	<i>THAEWMX</i>	Те саме, максимальне значення

Значення ідентифікаторів за замовчуванням – 0, а одиниця виміру – с.

#### 4.9 Джерела постійних логічних сигналів

Ці компоненти мають виходи, але не мають входів. Логічний рівень вихідного сигналу дорівнює «1» для джерел типу *PULLUP* і «0» для джерел типу *PULLDN*.

Внутрішні опори джерел задаються в описі моделі вхід/вихід.

Джерела постійних логічних сигналів наведено в табл. 4.11.

Таблиця 4.11 – Джерела постійних логічних сигналів

Тип	Параметр	Порядок перерахування виводів	Функціональне призначення
<i>PULLUP</i>	Кількість джерел логічного сигналу «1»	Вих.1, вих. 2, ...	Матриця джерел
<i>PULLDN</i>	Кількість джерел логічного сигналу «0»	Вих.1, вих. 2, ...	Матриця джерел

Наведемо приклад задання джерел логічних сигналів зі своїми моделями вхід/вихід:

*U4 PULLUP (2) \$G\_DPWR \$G\_DGND PINO PIN1 R2K; Два джерела «1»*

*U5 PULLDN (4) \$G\_DPWR \$G\_DGND BUS0, BUS1, BUS2, BUSS R50; чотири джерела «0»*

*.MODEL R2K UIO (drvh = 2K)*

*.MODEL R50 UIO (drvl = 50)*

#### 4.10 Цифрові лінії затримки

Вони здійснюють затримку вхідного сигналу будь-якої тривалості (зауважимо, що вентиля не пропускають імпульси, тривалість яких менша за час затримки). Лінії затримки мають тип *DLYNE*, у списку вузлів після перерахування вузлів підключення джерела живлення вказуються <вузол входу> і <вузол виходу>.

Модель динаміки лінії затримки має форму

*.MODEL <Ім'я моделі> UDLY [(параметри)]*

Параметри цифрових ліній затримки наведено в табл. 4.12 (значення за замовчуванням – 0, одиниця виміру – с).

Таблиця 4.12 – Параметри цифрових ліній затримки

Ідентифікатор	Параметр
<i>DLYMN</i>	Мінімальна затримка
<i>DLYTY</i>	Типова затримка
<i>DLYMX</i>	Максимальна затримка

Значення ідентифікаторів за замовчуванням – 0, а одиниця виміру – с.



1. Які основні поняття описують цифрові компоненти?
2. Який загальний опис цифрових компонентів PSpice OrCAD?
3. Як можна описати багаторозрядні АЦП та ЦАП?
4. Який опис моделей вентилів?
5. Як описуються моделі пристроїв живлення?
6. Як можна описати моделі генераторів цифрових сигналів?
7. Як визначити форми цифрового сигналу за форматом (пристрої STIM)?
8. Як відбувається зчитування тимчасової діаграми цифрового сигналу з файлу (пристрої FSTREAM)?
9. Як можна описати програмовані логічні матриці?
10. Як описуються моделі запам'ятовувальних пристроїв?
11. Як можна описати постійні запам'ятовувальні пристрої?
12. Як описуються оперативні запам'ятовувальні пристрої?
13. Як описується модель джерела постійних логічних сигналів?
14. Як можна описати моделі цифрових ліній затримки?



## 5 МОДЕЛІ СИГНАЛІВ

### 5.1 Незалежні джерела сигналів

#### 5.1.1 Стандартні сигнали

Незалежні джерела напруги ( $V$ ) і струму ( $I$ ) стандартного виду описуються виразами:

-  $V_{xxx} <+ввузол> <-ввузол> [[DC]<значення>][AC<модуль>[<фаза>]] + [STIMULUS = <ім'я сигналу>] + [специфікація сигналу]$

-  $I_{xxx} <+ ввузол> <-ввузол> [[DC]<значення>][AC<модуль>[<фаза>]] + [STIMULUS = <ім'я сигналу>] + [специфікація сигналу]$

Після ключового слова *STIMULUS* вказується ім'я сигналу, створеного за допомогою програми Stimulus Editor. Під час аналізування перехідних процесів можна використовувати один із стандартних сигналів за допомогою конструкції *<специфікація сигналу>*, що має вигляд:

- *PULSE <параметри>* – імпульсний сигнал;
- *SIN <параметри>* – синусоподібний сигнал;
- *EXP <параметри>* – сигнал експоненційної форми;
- *PWL <параметри>* – кусково-лінійний сигнал;
- *SFFM <параметри>* – гармонічний сигнал з синусоподібною частотною модуляцією.

Джерела можуть використовуватися у всіх видах аналізу. За замовчуванням всі параметри набувають нульових значень. Параметр *DC* визначає постійну складову джерела напруги або струму. Для режиму *AC* задаються модуль і фаза (в градусах) джерела гармонічного сигналу. Наведемо приклади:

*IBIAS 1302.3MA*

*VAC 2 3 AC .001*

*VACPHS 2 3 AC .001 90*

*13 26 77 DC 0.002 AC 1 SIN (.002 0.05 1.5MEG)*

### 5.1.2 Експоненціальна функція

Експоненціальна функція задається списком параметрів

$$EXP (y_1 y_2 t_d t_{cr} t_r t_{fr})$$

і описується виразом:

*y*, при  $0 < t < t_d$ ;

$y_1 + (y_2 - y_1) \{1 - \exp[-(t - t_d) / t_{cr}]\}$ , при  $t_d < t < t_r$ ;

$y_1 + (y_2 - y_1) \{1 - \exp[-(t - t_d) / t_{cr}] - 1 + \exp[-(t - t_r) / t_{fr}]\}$ , при  $t_r < t < TSTEP$ .

Графік експоненціальної функції наведено на рис. 5.1, а її параметри показані в табл. 5.1.

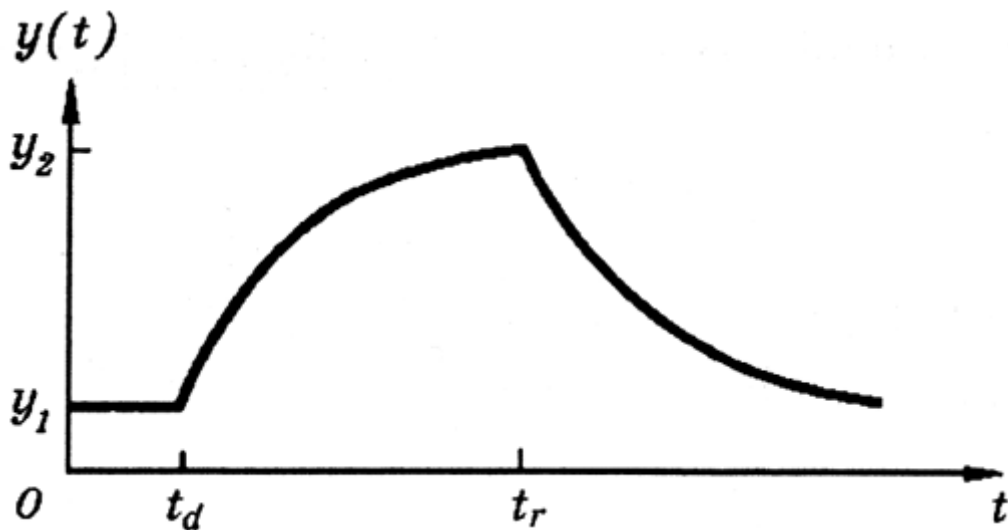


Рисунок 5.1 – Експоненціальна функція

Таблиця 5.1 – Параметри експоненціального сигналу

Позначення	Параметр	Розмірність	Значення за замовчуванням
$y_1$	Початкове значення	В або А	–
$y_2$	Максимальне значення	В або А	–
$t_d$	Початок переднього фронту	с	0
$t_{cr}$	Стала часу переднього фронту	с	$TSTEP^*$
$t_r$	Початок заднього фронту	с	$t_d + TSTEP$
$t_{fr}$	Стала часу заднього фронту	с	$TSTEP$

$TSTEP^*$  – крок за часом виведення на друк результатів розрахунків перехідних процесів (задається в директиві *.TRAN*).

Наведемо приклад опису джерела напруги експоненційної форми: *V1 3 4 EXP (0.02 2NS 20NS 40NS 20NS)*.

Імпульсна функція (рис. 5.2) задається списком наведених в табл. 5.2 параметрів

*PULSE (y1 y2 t\_d t\_r t\_f mT)*.



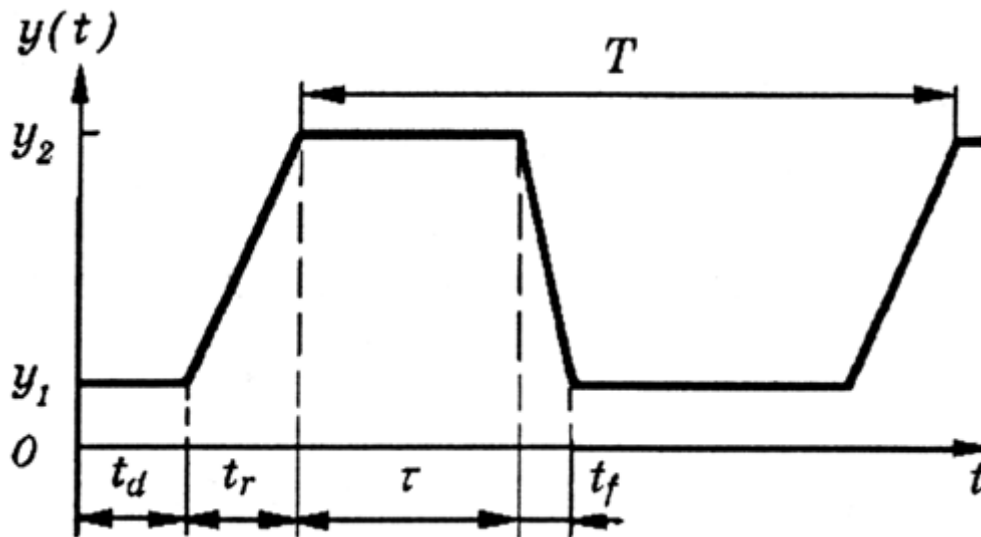


Рисунок 5.2 – Імпульсна функція

Таблиця 5.2 – Параметри імпульсного сигналу

Позначення	Параметр	Розмірність	Значення за замовчуванням
$y_1$	Початкове значення	В або А	–
$y_2$	Максимальне значення	В або А	–
$t_d$	Початок переднього фронту	с	0
$t_r$	Тривалість переднього фронту	с	$TSTEP^*$
$t_f$	Тривалість заднього фронту	с	$TSTEP$
$m$	Тривалість плоскої частини імпульсу	с	$TSTOP^{**}$
$T$	Період повторення	с	$TSTOP$

Згідно таблиці  $TSTEP^*$  – крок виведення на друк;  $TSTOP^{**}$  – кінцевий час аналізу перехідного процесу (параметри  $TSTEP$  і  $TSTOP$  задаються в директиві `.TRAN`) [12].

### 5.1.3 Кусково-лінійна функція

Кусково-лінійна функція задається списком параметрів  
`PWL [TIME_SCALE_FACTOR = <значення>] [VALUE_SCALE_FACTOR = <значення>] + (точки_відліку) *`

До секції (точки\_відліку) заносяться такі дані:

- ( $\langle t_n, y_n \rangle$ ) – координати точок;
- `FILE <ім'я файлу>` – читання координат точок з файлу;

- *REPEAT FOR*  $\langle n \rangle$  (точки\_відліку) \* *ENDREPEAT* – повторення  $n$  разів;

- *REPEAT FOREVER* (точки\_відліку) \* *ENDREPEAT* – нескінченне повторення;

- *PWL* є кусково-лінійною функцією  $y(t)$ .

Кусково-лінійна функція наведена на рис. 5.3.

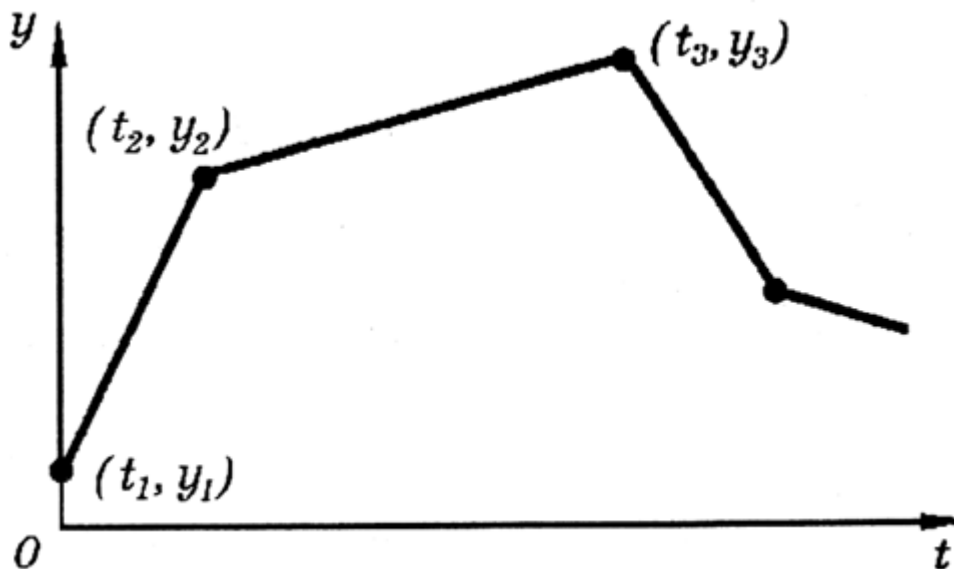


Рисунок 5.3 – Кусково-лінійна функція

Наведемо приклади.

Приклад 1. Східчаста напруга – 10 періодів по 1 мкс:

```
.PARAM N = 1
```

```
.STEPPARAMN 1.5.1
```

```
V1 1 0 PWL TIME_SCALE_FACTOR = 1e-6; час задано в мкс
```

```
+ REPEAT FOR 10
```

```
+ (.25, 0) (.26, {N}) (.99, {N}) (1,0)
```

```
+ ENDREPEAT
```

Приклад 2. Зчитування сигналу з файлу:

```
V2 1 0 PWL FILE triangle.in
```

У текстовому файлі *triangle.in* містяться значення відліків, наприклад  $(1\text{ms}, 1)$   $(2\text{ms}, 0)$   $(3\text{ms}, 1)$   $(4\text{ms}, 0)$  [12].

### 5.1.4 Синусоподібна функція з частотною модуляцією

Синусоподібна функція з частотною модуляцією задається списком параметрів  $SFFM(y_o, y_a, f_c, m, f_m)$ .

Параметри частотно-модульованого сигналу наведено в табл. 5.3.

Таблиця 5.3 – Параметри частотно-модульованого сигналу

Позначення	Параметр	Розмірність	Значення за замовчуванням
$y_o$	Постійна складова	В або А	–
$y_a$	Амплітуда	В або А	–
$f_c$	Частота несучої	Гц	$1/TSTOP$
$m$	Індекс частотної модуляції	–	0
$f_m$	Частота модуляції	Гц	$1/TSTOP$

Синусоподібна функція з частотною модуляцією описується виразом  $y(t) = y_o + y_a \sin [2\pi f_c t + m \sin (2\pi f_m t)]$

### 5.1.5 Сигнали довільної форми

Оскільки програма PSpice має досить бідний набір вбудованих джерел стандартних сигналів, цікаво обговорити можливість генерації сигналів довільної форми  $y(t)$ . З цією метою можна використовувати генератор цифрових сигналів, логічні стани якого при підключенні до аналогової частини схеми автоматично перетворюються на імпульсну електричну напругу, що може слугувати джерелом сигналу. Таким чином, зокрема, можна створити генератор псевдовипадкових сигналів. За необхідності імпульсна напруга згладжується аналоговим фільтром. Форма цифрового сигналу задається двояко:

- шляхом задання дискретних відліків  $(y_i, t_i)$  безпосередньо в описі цифрового сигналу (конструкція STIM);
- шляхом попереднього запису відліків  $(y_i, t_i)$  у файл, ім'я якого вказується в описі цифрового сигналу (конструкція FSTIM).

Другий спосіб кращий при формуванні довгих відрізків реалізацій сигналів складної форми, зокрема випадкових сигналів. Відзначимо, що формовані таким чином імпульсні сигнали мають постійне значення на інтервалі між дискретними відліками  $y(t) = y(t) \text{ при } t_i < t < t_{i+1}$ . Тим самим при постійному кроці дискретизації  $t = t_{i+1} - t_i$  сигнал формується з запізненням на  $t/2$  [12].

## 5.2 Залежні джерела сигналів

Залежні джерела напруги та струму можуть бути як лінійними, так і нелінійними. Існує чотири їх різновиди:

- $v=e(v)$  – джерело напруги, кероване напругою (ДНКН);
- $i=f(i)$  – джерело струму, кероване струмом (ДСКС);
- $i=g(v)$  – джерело струму, кероване напругою (ДСКН);
- $v=h(i)$  – джерело напруги, кероване струмом (ДНКС).

Лінійні керовані джерела описуються залежностями  $v=ev$ ,  $i=fi$ ,  $i=gv$ ,  $v=hi$ , де  $e$ ,  $f$ ,  $g$  і  $h$  – коефіцієнти передачі.

В описі нелінійних залежних джерел  $y=y(x_1, x_2, \dots, x_n)$  використовується поліноміальна функція *POLY* вигляду  $Y=P_0+P_1x_1+P_2x_2+\dots+P_nx_n \dots$ .

Тут  $x_1, x_2, \dots, x_n$  –  $n$  керуючих змінних. Вони можуть бути струмами або різницею потенціалів. Одночасне керування струмом і різницею потенціалів не допускається. Всі коефіцієнти полінома вводити не обов'язково, але вводити їх необхідно послідовно; один за одним без пропусків [12].

### 5.2.1 Джерело напруги, кероване напругою

Джерело напруги, кероване напругою, задається такими залежностями.

1. Лінійне джерело:

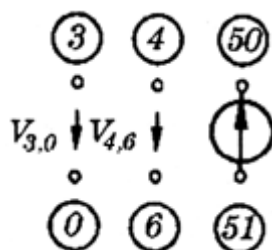
$E_{xxx} \langle +\text{вузол} \rangle \langle -\text{вузол} \rangle \langle +\text{керуючий вузол} \rangle \langle -\text{керуючий вузол} \rangle + \langle \text{коефіцієнт передачі} \rangle$

2. Нелінійне джерело:

$E_{xxx} \langle +\text{вузол} \rangle \langle -\text{вузол} \rangle \text{POLY}(\langle n \rangle) \langle \langle +\text{керуючий вузол} \rangle + \langle -\text{керуючий вузол} \rangle \rangle^* + \langle \text{коефіцієнт полінома} \rangle^*$

Наприклад, лінійне джерело  $ELIN = 2,5V_{10.11}$  описується залежністю  $ELIN121011 \ 2.5$ .

Нелінійне джерело напруги, кероване напругою,  $ENONLIN=0+13,6V_{3.0}+0,2V_{4.6}+0,0051V_{3.0}$  (рис. 5.4).



## Рисунок 5.4 – Нелінійне джерело напруги, кероване напругою

Нелінійне джерело *ENONLIN* описується як  
*ENONLIN 50 51 POLY(2) (3,0) (4,6) 0.0 13.6 0.2 0.005*

### 5.2.2 Джерело струму, кероване струмом

Джерело струму, керований струмом, задається таким чином.

1. Лінійне джерело:  $F_{xxx} \langle + \text{вузол} \rangle \langle -\text{вузол} \rangle \langle \text{ім'я керуючого джерела напруги} \rangle + \langle \text{коефіцієнт передачі} \rangle$

2. Нелінійне джерело:  $F_{xxx} \langle +\text{вузол} \rangle \langle -\text{вузол} \rangle \text{POLY} (\langle n \rangle) + \langle \text{ім'я керуючого джерела напруги} \rangle^* + \langle \text{коефіцієнт полінома} \rangle^*$

Керуючим струмом слугує струм незалежного джерела напруги. Наприклад, лінійне джерело струму, кероване струмом,  $F1 = 7,5I_{V4}$  (рис. 5.5) описується залежністю *F1 2 1 V4 7.5*.

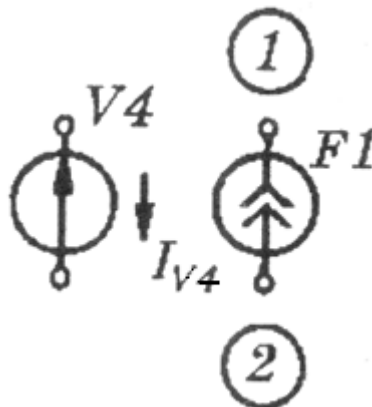


Рисунок 5.5 – Лінійне джерело струму, кероване струмом

Нелінійне джерело струму *FNONLIN* =  $0,01 + 13,6I_{VC1} + 0,2I_{VC2} + 0,0054 + 0,001I_{VC1} I_{VC2}$  описується залежністю *FNONLIN1011 POLY (2) VC1 VC2 0.01 13.60.20.0050.001*.

### 5.2.3 Джерело струму, кероване напругою

Джерело струму, керований напругою, задається виразами.

1. Лінійне джерело:

$G_{xxx} \langle +\text{вузол} \rangle \langle -\text{вузол} \rangle \langle +\text{керуючий вузол} \rangle \langle -\text{керуючий вузол} \rangle + \langle \text{коефіцієнт передачі} \rangle$ .

2. Нелінійне джерело:

$G_{xxx} <+взуол> <-взуол> POLY (<n>) <+керуючий вузол> + <-керуючий вузол>> * <коефіцієнт полінома> *.$

### 5.2.4 Джерело напруги, кероване струмом

Джерело напруги, керований струмом, задається таким чином.

1. Лінійне джерело:

$H_{xxx} <+взуол> <-взуол> <ім'я керуючого джерела напруги> + <коефіцієнт передачі>.$

2. Нелінійне джерело:  $H_{xxx} <+ вузол> <-взуол> POLY (<t>) + <Ім'я керуючого джерела напруги> * + <Коефіцієнт полінома> *.$

Зауваження. Через те, що в описах лінійних керованих джерел не допускається використання параметрів і функцій, в цих цілях можна скористатися залежними джерелами з нелінійними передавальними функціями.

Одне із застосувань нелінійних залежних джерел – генерація амплітудно-модульованих сигналів. Як приклад, на рис. 5.6 показаний формувач сигналу, отриманий в результаті виконання моделювання:

```
RADIOSIGNAL
VSIN 1 OSIN(0 1 100)
VPULSE 2 0 PULSE(0 1 0 0.25 0.25 0.5 1)
GSIGNAL 0 3 POLY(2)(1,0)(2,0) 00001
R1101MEG
R2201MEG
R3301MEG
.IRAN 0.01 1.6
.PROBE V(1)V(2)V(3)
.END
```

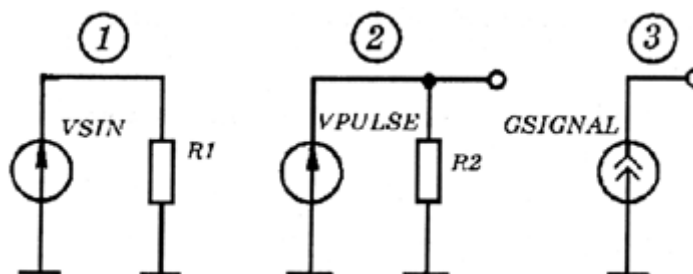


Рисунок 5.6 – Формувач сигналу

Тут *VSIN* – джерело гармонічного сигналу з частотою 100 Гц; *VPULSE* – джерело імпульсної напруги з періодом повторення 1 с. Перемноження цих сигналів за допомогою нелінійного джерела струму *GSIGNAL* створює послідовність радіоімпульсів.

Наявність залежних джерел дозволяє моделювати не тільки електричні принципові схеми, але й функціональні схеми динамічних систем зі зворотними зв'язками, а також вирішувати системи диференціальних рівнянь.

Розглянемо як приклад пристрій обчислення квадратного кореня, функціональна схема якого показана на рис. 5.7, а). При нескінченно великому коефіцієнті передачі підсилювача  $K \rightarrow$  нескінченності вихідна напруга  $V_{out} = (V_{in})^{1/2}$ .

Складемо еквівалентну принципову схему заміщення цього пристрою для PSpice, використовуючи компоненти, дозволені в програмі PSpice (рис. 5.7, б).

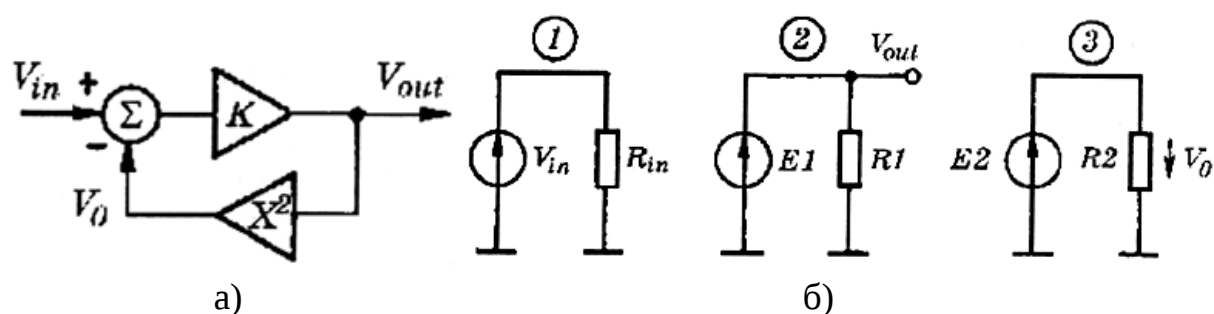


Рисунок 5.7 – Функціональні схеми: а – пристрій обчислення квадратного кореня; б – еквівалентна принципова схема для PSpice

За допомогою нелінійного джерела *E1* здійснюється віднімання напруг  $V_{in}$ ,  $V_0$  і посилення різницевого сигналу в  $K$  раз. Вихідна напруга  $V_{out}$ , підноситься до квадрата за допомогою нелінійного джерела *E2*. Кожне джерело напруги замкнуте на великий опір, щоб у схемі заміщення не було розімкнутих контурів. На вхідній мові програми PSpice схема заміщення описується таким чином:

*Rin 1 0 1MEG*

*E1 2 0 POLY (2) (1,0) (3,0) 0 1E6 -1E6; підсилювач-суматор*  
*R1201MEG*

*E2 3 0 POLY (2) (2,0) (2,0) 00001; квадратор*

Зауваження. Відзначимо, що наявність в редакторі PSpice Schematics бібліотеки символів функціональних блоків amb.slb дозволяє не складати електричні схеми заміщення функціональних схем типу рис. 5.7, б), а відразу складати функціональні схеми з наявних стандартних функціональних блоків.

Продемонструємо методику розв'язання диференціальних рівнянь на прикладі системи рівнянь

$$dx_1/dt = -0,3x_1 + x_2 + x_1^2 - 3x_1 x_2 - x_1$$

$$dx_2/dt = 0,24 - 0,6x_2 + 4x_1 x_2 - 6x_1^2 x_2$$

з початковими умовами  $x_1(0)=0$ ,  $x_2(0)=0,4$ . Використовуючи рівняння конденсатора  $i=Cdu/dt$ , змодельуємо систему диференціальних рівнянь за допомогою залежних джерел струму GX1, GX2, підключених до конденсаторів C1, C2, як показано на рис. 5.8.

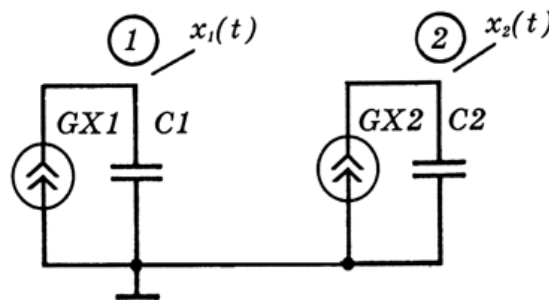


Рисунок 5.8 – Моделювання системи диференціальних рівнянь

Вхідною мовою програми PSpice моделювання складається таким чином:

*DIFFERENTIAL EQUATIONS*

*GX1 0 1 POLY (3) (1,0) (2,0) (0,0) 0-0.3 101-300001*

*GX2 0 2 POLY (3) (1,0) (2,0) (0,0) 0.6 0-0.6 00400000-6*

*C1 1 01*

*C2201*

*.ICV (1) = 0V (2) = 0.4*

*.IRAN 0.1s 40s SKIPBP*

### 5.3 Приклад обрахунку залежного джерела сигналу



Джерело напруги, кероване напругою, зображене на рисунку 5.9, містить незалежне джерело напруги  $V$  і залежне джерело напруги  $E$  з міткою  $2V_a$ . Його вихідна напруга є функцією напруги на резисторі  $R_1$ , яка позначається  $V_a$ . Множник 2 вказує, що напруга  $E$  дорівнює подвоєному значенню  $V_a$ . У загальному вигляді множник позначається як  $k$ .

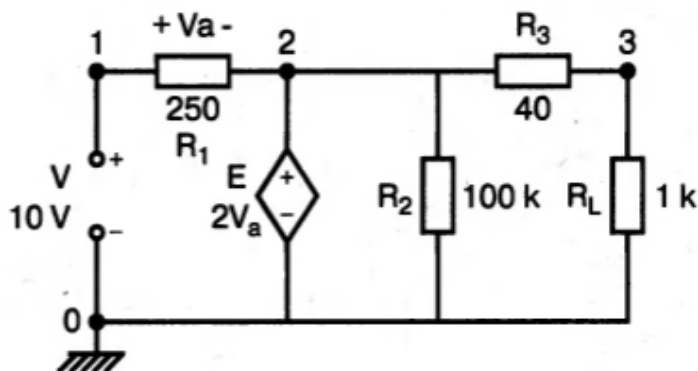


Рисунок 5.9 – Джерело напруги, кероване напругою

Струми і напруги в схемі можуть бути отримані за допомогою звичайних рівнянь. Для лівого контуру другий закон Кірхгофа

$$V = R_1 I_{12} + E = R_1 I_{12} + 2V_a,$$

де  $I_{12}$  – струм через резистор  $R_1$ .

Оскільки  $V_a = R_1 I_{12}$ , вираз можна записати як

$$V = R_1 I_{12} + 2R_1 I_{12} = 3R_1 I_{12};$$

$$10V = 3 \cdot (250\Omega) \cdot I_{12};$$

$$I_{12} = 13,33\text{mA};$$

$$V_{12} = V_a = R_1 I_{12} = (250\Omega) \cdot (13,33\text{mA}) = 3,333V;$$

$$E = 2V_a = 6,667V.$$

Оскільки ця напруга прикладена до  $R_2$ , то можна розрахувати струм через  $R_2$ , а також струм через вітку, яка містить  $R_3$  і  $R_L$ :

$$I_{R2} = \frac{6,667B}{100k\Omega} = 66,67 \mu A;$$

$$I_{R3} = \frac{6,667B}{1,04k\Omega} = 4,4 \mu A.$$

Струм через джерело визначається за першим законом Кірхгофа:

$$I_E = 13,33 \mu A - 66,67 \mu A - 4,4 \mu A = -57,74 \mu A.$$

Вхідним файлом для схеми буде лістинг програми (рис. 5.10).

```

00 Voltage-Controller Voltage Sources
01 V 1 0 10V
02 E 2 0 1 2 2
03 R1 1 2 250
04 R2 2 0 100k
05 R3 2 3 40
06 RL 3 0 1k
07 .OP
08 .OPT nopage
09 .TF V(3) V
10 .END

```

Рисунок 5.10 – Лістинг програми вхідного файла

Вихідним файлом для схеми буде лістинг програми (рис. 5.11).

```
**** 07/27/05 12:27:16 ***** Evaluation PSpice (Nov 1999) *****
Voltage-Controlled Voltage Sources
```

```
****          CIRCUIT DESCRIPTION
```

```
V 1 0 10V
E 2 0 1 2 2
R1 1 2 250
R2 2 0 100k
R3 2 3 40
RL 3 0 1k
.OP
.OPT nopage
.TF V(3) V
.END
```

```
****          SMALL SIGNAL BIAS SOLUTION          TEMPERATURE = 27.000 DEG C
```

NODE	VOLTAGE	NODE	VOLTAGE	NODE	VOLTAGE	NODE	VOLTAGE
( 1)	10.0000	( 2)	6.6667	( 3)	6.4103		

```
VOLTAGE SOURCE CURRENTS
```

NAME	CURRENT
------	---------

V	-1.333E-02
---	------------

```
TOTAL POWER DISSIPATION 1.33E-01 WATTS
```

```
****          OPERATING POINT INFORMATION          TEMPERATURE = 27.000 DEG C
```

```
**** VOLTAGE-CONTROLLED VOLTAGE SOURCES
```

NAME	E
V-SOURCE	6.667E+00
I-SOURCE	6.856E-03

```
****          SMALL-SIGNAL CHARACTERISTICS
```

```
V(3)/V = 6.410E-01
INPUT RESISTANCE AT V = 7.500E+02
OUTPUT RESISTANCE AT V(3) = 3.846E+01
```

Рисунок 5.11 – Лістинг програми вихідного файла

Новою командою у вхідному файлі є команда опису залежного джерела напруги  $E$ . Вихідний файл (рис. 5.11) дає значення  $V(2)=6,6667$  В і  $V(3)=6,4103$  В. Струм джерела напруги  $V$  також дорівнює розрахованому 13,333 мА. Струм джерела живлення  $E$  дорівнює 6,856 мА і направлений від позитивного полюса джерела.

Вихідний файл дає значення вхідного опору в 750 Ом. Це відношення  $V$  до  $I_{12}$ . Для визначення вихідного опору всі джерела живлення потрібно закортити. При закортчуванні вузлів 2 і 0 паралельне з'єднання  $R_3$  і  $R_L$  дає опір 8,46 Ом.



1. Як описуються незалежні джерела сигналів?
2. Як можна описати стандартні сигнали?
3. Який опис експоненціальної функції?
4. Як описується кусково-лінійна функція?
5. Як можна описати синусоподібну функцію з частотною модуляцією?
6. Який опис сигналів довільної форми?
7. Як описуються залежні джерела сигналів?
8. Як можна описати джерело напруги, кероване напругою?
9. Як описується джерело струму, кероване струмом?
10. Як можна описати джерело струму, кероване напругою?
11. Який опис джерела напруги, кероване струмом?

## **СПИСОК ЛІТЕРАТУРИ**

1. Семеренко М. М. Автоматизоване моделювання елементів електронної техніки / Семеренко М. М. – Вінниця : ВДГУ, 2001. – 85 с.
2. Кеон Дж. OrCAD Pspice. Анализ электронных цепей / Кеон Дж. – М. : ДМК Пресс, 2008. – 640 с.
3. Норенков И. П. САПР в радиоэлектронике / Норенков И. П. – М. : Радио и связь, 2002. – 480 с.
4. Ильин В. Н. Автоматизация схемотехнического проектирования / Ильин В. Н. – М. : Радио и связь, 2001. – 360 с.
5. Корячко В. П. Теоретические основы САПР / Корячко В. П., Курейчик В. М., Лоренков И. П. – М. : Энергоатомиздат, 1999. – 380 с.
6. Разевиг В. Д. Система проектирования OrCAD / Разевиг В. Д. – М. : Солон-Р, 2000. – 366 с.

7. Красилькова Г. А. Автоматизация инженерно-графических работ / Красилькова Г. А., Самсонов В. В., Тарелкин С. М. – СПб. : Питер, 2001. – 256 с.

8. Поляков А. К. Язык VHDL в САПР цифровых схем / Поляков А. К. – М. : МЭИ, 1999. – 342 с.

9. Грошев Д. Е. Применение пакета OrCad для компьютерного проектирования электронных схем / Д. Е. Грошев, В. К. Макуха. – Новосибирск : НГТУ, 1999. – 426 с.

10. Нефедов А. В. Интегральные микросхемы и их зарубежные аналоги / Нефедов А. В. – М. : КУБК, 1996. – 996 с.

11. Болотовский Ю. С. OrCAD. Моделирование. «Поваренная» книга / Болотовский Ю. С. – М. : Солон-Пресс, 2013. – 200 с.

12. Кузнецова С. А. OrCad 10. Проектирование печатных плат / Кузнецова С. А., Нестеренко А. В., Афанасьев А. О. – М. : Телеком, 2005. – 454 с.

13. Смирнов Ю. А. Электронные и микропроцессорные системы управления / Ю. А. Смирнов, А. В. Муханов. – М. : Лань, 2012. – 624 с.

*Навчальне видання*

**Огородник Костянтин Володимирович  
Книш Богдан Петрович  
Ратушний Павло Миколайович  
Лазарев Олександр Олександрович**

## **МОДЕЛЮВАННЯ В ЕЛЕКТРОНІЦІ**

**Навчальний посібник**

Редактор Т. Старічек

Оригінал-макет підготовлено Б. Книшом

Підписано до друку .....  
Формат 29,7×42¼. Папір офсетний.  
Гарнітура Times New Roman.  
Друк різнографічний. Ум. друк. арк. ....  
Наклад .... пр. Зам. № 2017-

Видавець та виготовлювач  
інформаційний редакційно-видавничий центр.  
ВНТУ, ГНК, к. 114.  
Хмельницьке шосе, 95,  
м. Вінниця, 21021.  
Тел. (0432) 59-85-32, 59-87-38.  
**press.vntu.edu.ua;**  
*E-mail:* kivc.vntu@gmail.com.  
Свідоцтво суб'єкта видавничої справи  
серія ДК № 3516 від 01.07.2009 р.