

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Запорізький національний технічний університет

Кафедра ПЗ

МЕТОДИЧНІ ВКАЗІВКИ
до лабораторних робіт
з дисципліни
“ТЕХНОЛОГІЇ КОМП’ЮТЕРНОГО ПРОЕКТУВАННЯ”
для студентів освітнього ступеня бакалавр
за спеціальністю
122 Комп’ютерні науки та інформаційні технології
денної форми навчання

2016

Методичні вказівки до лабораторних робіт з дисципліни “Технології комп’ютерного проектування” для студентів освітнього ступеня бакалавр за спеціальністю 122 Комп’ютерні науки та інформаційні технології, денної форми навчання /Укл.: А.В. Пархоменко, М.О. Андреев. – Запоріжжя: ЗНТУ, 2016. – 76 с.

Укладачі: А.В. Пархоменко, к.т.н., доцент кафедри ПЗ,
М.О. Андреев, асистент кафедри ПЗ.

Рецензент: Г.М. Шило, к.т.н., доцент кафедри ІТЕЗ.

Відповідальний
за випуск: С.О. Субботін, зав. каф. ПЗ, д.т.н., професор.

Затверджено
на засіданні кафедри
програмних засобів
Протокол № 1
від “16” серпня 2016 р.

Рекомендовано до видання
НМК факультету
комп’ютерних наук і
технологій
Протокол № 1
від “22” серпня 2016 р.

ЗМІСТ

1 Лабораторна робота №1. Моделювання крайових задач за допомогою електричних сіток	6
1.1 Теоретичні відомості	6
1.1.1 Види рівнянь в часткових похідних	6
1.1.2 Побудова електричного аналога диференційного рівняння з частковими похідними	8
1.1.3 Розрахунок параметрів електричного аналога диференційного рівняння.....	13
1.2 Завдання до виконання лабораторної роботи	15
1.3 Зміст звіту	15
1.4 Контрольні питання	15
2 Лабораторна робота №2. Застосування методу скінчених різниць до розв'язку стаціонарних задач на компонентному рівні	16
2.1 Теоретичні відомості	16
2.1.1 Вивід скінченно-різницевих схем.....	16
2.1.2 Розв'язування задач методом скінчених різниць.....	19
2.1.3 Розв'язок систем лінійних алгебраїчних рівнянь	20
2.1.4 Алгоритм чисельного розв'язку задачі Діріхле (метод Лібмана).....	22
2.2 Завдання до виконання лабораторної роботи	23
2.3 Зміст звіту	23
2.4 Контрольні питання	23
3 Лабораторна робота №3. Застосування методу скінчених елементів до розв'язку стаціонарних задачах на компонентному рівні.....	24
3.1 Теоретичні відомості	24
3.1.1 Сутність методу скінчених елементів	24

3.2 Завдання до виконання лабораторної роботи	29
3.3 Зміст звіту	29
3.4 Контрольні питання	30
4 Лабораторна робота №4. Розв’язання задачі параметричної ідентифікації об’єкта	31
4.1 Теоретичні відомості	31
4.1.1 <i>Опис методики побудови математичних моделей</i>	<i>31</i>
4.1.2 <i>Опис топологічної моделі біполярного транзистора ..</i>	<i>32</i>
4.1.3 <i>Методика ідентифікації первинних параметрів моделі транзистора</i>	<i>34</i>
4.1.3.1 <i>Визначення початкових значень параметрів моделі емітерного переходу транзистора</i>	<i>34</i>
4.1.3.2 <i>Визначення параметрів моделі емітерного переходу транзистора</i>	<i>37</i>
4.1.3.3 <i>Визначення параметрів моделі колекторного переходу транзистора</i>	<i>38</i>
4.2 Завдання до виконання лабораторної роботи	40
4.3 Зміст звіту	41
4.4 Контрольні питання	41
5 Лабораторна робота №5. Розробка та дослідження імітаційних моделей виробничого процесу	42
5.1 Теоретичні відомості	42
5.1.1 <i>Системи масового обслуговування</i>	<i>42</i>
5.1.2 <i>Призначення, особливості та функціональні можливості програми</i>	<i>43</i>
5.1.3 <i>Опис інтерфейсу програми</i>	<i>44</i>
5.1.4 <i>Основні поняття системи моделювання</i>	<i>46</i>
5.1.5 <i>Опис досліджуваної системи масового обслуговування ..</i>	<i>48</i>
5.1.6 <i>Послідовність моделювання в системі Service Model</i>	<i>49</i>
5.1.7 <i>Вихідні дані програми</i>	<i>58</i>
5.2 Завдання до виконання лабораторної роботи	60

5.3 Зміст звіту	61
5.4 Контрольні питання	61
Рекомендована література	62
Додаток А – Завдання до лабораторної роботи №1	63
Додаток Б – Методика побудови та аналізу електричних принципових схем за допомогою системи CircuitMaker 6 Pro .	64
Додаток В – Завдання до лабораторної роботи №2	68
Додаток Д – Завдання до лабораторної роботи №3	69
Додаток Ж – Завдання до лабораторної роботи №4	70
Додаток К – Завдання до лабораторної роботи №5	71

1 ЛАБОРАТОРНА РОБОТА №1

Моделювання крайових задач за допомогою електричних сіток

Мета роботи: ознайомитися з розв'язком фізичних задач на компонентному рівні за допомогою електричних сіток.

1.1 Теоретичні відомості

1.1.1 Види рівнянь в часткових похідних

Більшість фізичних задач, які вирішують інженери на компонентному рівні (моделювання легованих областей у напівпровіднику, теплових і електричних полів, поширення хвиль та ін.), можуть бути описані за допомогою рівнянь в часткових похідних (РЧП).

На відміну від звичайних диференціальних рівнянь, в яких невідома функція залежить лише від однієї незалежної змінної, в РЧП невідома функція залежить від декількох змінних (наприклад, концентрація $N(x, y, z, t)$ залежить від координат x, y, z та часу t).

Рівняння в часткових похідних можна класифікувати за шістьма основними ознаками:

– **порядок рівняння** – найвищий ступінь часткових похідних, які входять до рівняння. Наприклад:

$$\frac{\partial N}{\partial t} = a \frac{\partial^2 N}{\partial x^2} \quad \text{– рівняння 2-го порядку,} \quad (1.1)$$

$$\frac{\partial^4 w}{\partial x^4} + 2 \frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial^4 y} = 0 \quad \text{– рівняння 4-го порядку.} \quad (1.2)$$

– **кількість змінних** – число незалежних змінних. Наприклад, рівняння (1.1) має дві незалежні змінні, а рівняння (1.3) – три:

$$\frac{\partial N}{\partial t} = a \frac{\partial^2 N}{\partial x^2} + a \frac{\partial^2 N}{\partial y^2} \quad (1.3)$$

– **лінійність**. У лінійні рівняння залежна змінна та усі її часткові похідні входять лінійно, зокрема, вони не множаться одна на одну, не підносяться до квадрату і т.д. Наприклад, рівняння (1.1)-(1.3) – лінійні, а рівняння (1.4) та (1.5) – нелінійні:

$$N \frac{\partial^2 N}{\partial x^2} + \frac{\partial N}{\partial t} = f(x), \quad (1.4)$$

$$x \frac{\partial N}{\partial x} + y \frac{\partial N}{\partial y} + N^2 = 0 \quad (1.5)$$

Для прикладу, лінійним рівнянням другого порядку з двома незалежними змінними називається рівняння виду:

$$A \frac{\partial^2 N}{\partial x^2} + B \frac{\partial^2 N}{\partial x \partial y} + C \frac{\partial^2 N}{\partial y^2} + D \frac{\partial N}{\partial x} + E \frac{\partial N}{\partial y} + FN = G, \quad (1.6)$$

де A,B,C,D,E,F,G – константи або задані функції незалежних змінних.

– **однорідність.** Рівняння (1.6) називають однорідним, якщо права частина G дорівнює нулю для усіх x та y в заданій області; інакше рівняння називають неоднорідним.

– **види коефіцієнтів.** Якщо коефіцієнти рівняння (1.6) A,B,C,D,E,F – константи, то (1.6) – рівняння з постійними коефіцієнтами. Якщо коефіцієнти – функції незалежних змінних, то маємо рівняння зі змінними коефіцієнтами.

– **стаціонарність.** Якщо розв’язок РЧП не змінюється у часі, то такі рівняння називаються стаціонарними (в іншому випадку – нестаціонарними). Наприклад, подальші рівняння є стаціонарними:

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} = 0 \text{ – рівняння Лапласа,} \quad (1.7)$$

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} = f(x, y, z) \text{ – рівняння Пуассона.} \quad (1.8)$$

Окремо взяте РЧП має безліч розв’язків. Тому, щоб отримати конкретний розв’язок, необхідно додати початкові та краєві умови. Найчастіше зустрічаються такі способи завдання умов на границі:

– задають значення функції $h(s)=f(s)$, де $f(s)$ – відома функція від координат точки s контуру. Цей випадок називають першою красою задачею, або задачею Діріхле;

– задають значення нормальної похідної від шуканої функції:

$$\frac{\partial h(s)}{\partial n} = f(s). \quad (1.9)$$

Це так звана друга краєва задача, або задача Неймана;

– при третій (змішаній) краєвій задачі задають лінійну комбінацію:

$$f_1(s)h(s) + f_2(s)\left(\frac{\partial h(s)}{\partial n}\right) = f_3(s). \quad (1.10)$$

1.1.2 Побудова електричного аналога диференційного рівняння з частковими похідними

На сьогоднішній день розроблено багато прикладних програм (МАЕС-П, Circuit Maker, PSPICE, MICRO), які призначені для розрахунку електричних схем. Ці програми можна з успіхом використовувати для розв'язку інженерних задач, які описуються рівняннями з частковими похідними (задачі механіки, гідро- та газодинаміки, тепло- та масопереносу) з використанням наближених електричних моделей.

Як приклад розглянемо задачу розв'язку рівняння теплопровідності (нестационарне рівняння Фур'є):

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = B(x, y) \frac{\partial T}{\partial t}, \quad (1.11)$$

де x, y – просторові координати;

t – час;

T – температура (шукана функція аргументів).

Проаналізуємо фізичний зміст другої похідної від температури по просторовій координаті. Друга похідна – границя відношення різниці значень перших похідних в двох сусідніх точках області моделювання, розміщених в напрямку вісі аргументу, за яким беруть похідну, до відстані між точками, за умови, що ця відстань прямує до нуля. Перші похідні від температури в сусідніх точках області за деякою віссю необхідно розглядати як *границю відношення різниці* між значеннями температури у відповідних точках області і в деякій нульовій точці, яка знаходиться між цими точками, *до відстані* від розглянутих точок до нульової точки за умови, що ця відстань прямує до нуля.

Різниця температур між сусідніми точками, тобто перша похідна, прямо пропорційна потоку тепла за одиницю часу між цими

точками. Це є тепловий аналог закону Ома, а коефіцієнт пропорційності – величина теплового опору. Друга похідна від температури за будь-якою просторовою віссю – це різниця між тепловим потоком за одиницю часу, що тече в дану точку в напрямку розглянутої осі, і потоком, що витікає з неї в напрямку цієї ж осі. Будь-яка нерівність призводить до пропорційного збільшення (зменшення) температури в даній точці. Коефіцієнтом пропорційності є величина обернена теплоємності.

Лапласіан, тобто ліва частина рівняння теплопровідності, відображає суму приростів теплових потоків в даній точці за всіма координатами. Вираз в правій частині рівняння теплопровідності пропорційний до приросту температури в даній точці. Отже, рівняння теплопровідності у вигляді (1.11) є тепловий аналог закону Кірхгофа (алгебраїчна сума струмів у будь-якому вузлі дорівнює нулю).

Зі сказаного видно, що для побудови електричного аналога рівняння теплопровідності необхідно моделювати тепловий опір, теплоємність, температуру та потік тепла. Теплопровідне середовище можна представити подібним до резистивного. Далі виникає думка про можливість наближеної заміни середовища, в якому моделюється поле, сіткою дискретних елементів. Суцільну неперервну область моделювання покривають сіткою з елементарних дискретних опорів, ємностей і т. д., які розміщені в напрямку осей координат. До такої сітки в місцях з'єднання елементів (вузлів) підводять потенціали та струми, які виражають початкові та краєві умови, а потенціали в інших вузлах відображають шукану функцію.

Наведені міркування характерні для так званого фізичного метода (часто його називають методом заміщення). Цей метод застосовують при чистому електричному моделюванні – електролітичні ванни, резистивний папір, а інколи – при створенні електричних сіток. При цьому, суцільне неперервне середовище замінюють “еквівалентним” на основі аналогії рівнянь і далі замінюють “еквівалентними” зосередженими елементами. Така заміна має переваги фізичної наочності. Але питання похибки при такій заміні лишається відкритим.

Другий метод – математичний, базується на тому, що схема складається так, щоб задовольнити алгебраїчні рівняння наближеної скінченно-різницевої апроксимації диференційного рівняння. При такому способі формування заступних електричних схем фізична

наочність втрачається, але з'являється наочність математична, яка дозволяє оцінити похибки.

Розглянемо фізичні процеси у плоскій сітці опорів (рис. 1.1). Запишемо перший закон Кірхгофа для даної схеми:

$$\sum I_{0-i} = 0, \quad (1.12)$$

де I_{0-i} – алгебраїчне значення струму, який тече/витікає з вузла.

Для I_{0-i} , згідно з законом Ома, можна отримати співвідношення:

$$I_{0-1} = \frac{V_1 - V_0}{R_1}, \quad (1.13)$$

$$I_{0-2} = \frac{V_2 - V_0}{R_2}, \quad (1.14)$$

$$I_{0-3} = \frac{V_3 - V_0}{R_3}, \quad (1.15)$$

$$I_{0-4} = \frac{V_4 - V_0}{R_4} \quad (1.16)$$

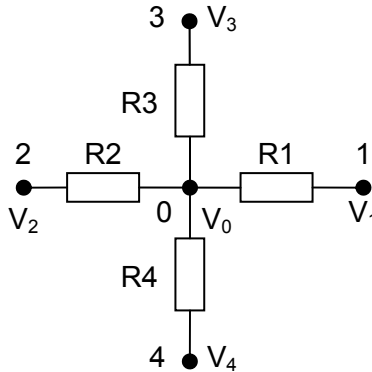


Рисунок 1.1 – Схема вузла плоскої сітки опорів

Тоді, враховуючи, що $R_1 = R_2 = R_3 = R_4$, отримаємо рівняння:

$$V_1 + V_2 + V_3 + V_4 - 4V_0 = 0 \quad (1.17)$$

Цей вираз є скінченно-різницевою апроксимацією двомірного стаціонарного рівняння Лапласа:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \quad (1.18)$$

У випадку тривимірного рівняння Лапласа вираз (1.17) перепишеться у формі:

$$V_1 + V_2 + V_3 + V_4 + V_5 + V_6 - 6V_0 = 0 \quad (1.19)$$

При сітковій апроксимації, кожен резистор моделі відповідає опору деякого об'єму суцільного середовища, що моделюють. Однак, в резисторі струм протікає в напрямку деякої вісі, хоча в елементі модельованого середовища струм може поширюватися в усіх напрямках. Опір кубічного елемента модельованого середовища обчислюють за формулою:

$$R = \rho \frac{l}{S}, \quad (1.20)$$

де ρ – питомий опір, Ом $\cdot$ м;

l – довжина елемента в напрямку лінії поширення струму, м;

S – елементарна площа, через яку проходить струм, м².

Тоді, якщо диференційний кубічний елемент симетричний, опори шістьох еквівалентних резисторів попарно рівні у кожному з трьох напрямків та визначаються за такими виразами (рис. 1.2):

$$R_1 = \rho \frac{\Delta X}{4 \Delta Y \Delta Z}, \quad R_3 = \rho \frac{\Delta Y}{4 \Delta X \Delta Z}, \quad R_5 = \rho \frac{\Delta Z}{4 \Delta X \Delta Y} \quad (1.21)$$

Аналогічна симетрія для плоскої задачі (рис. 1.1):

$$R_1 = \rho \frac{\Delta X}{2 \Delta Y}, \quad R_3 = \rho \frac{\Delta Y}{2 \Delta X} \quad (1.22)$$

Коефіцієнт ρ можна замінити будь-якою іншою постійною величиною, яку вибирають, виходячи з умов зручності масштабування (наприклад, $1/\lambda$ – для розрахунку теплових опорів).

При розв'язку нестационарного рівняння Фур'є (1.11), коли рівновага між тепловими потоками, що течуть до вузла і витікають з нього порушується, додатковий потік призводить до накопичення (або втрати) тепла в елементарному об'ємі протягом певного часу. Для електричної моделі таким елементом може бути ємність, яку включають між вузлом сітки та землею (рис. 1.3). У цьому випадку алгебраїчна сума струмів, які течуть до вузла, та струмів, що течуть від нього буде дорівнювати:

$$\sum I_0 = \frac{V_1 - V_0}{R_1} + \frac{V_2 - V_0}{R_2} + \frac{V_3 - V_0}{R_3} + \frac{V_4 - V_0}{R_4} + \frac{V_5 - V_0}{R_5} + \frac{V_6 - V_0}{R_6} \quad (1.23)$$

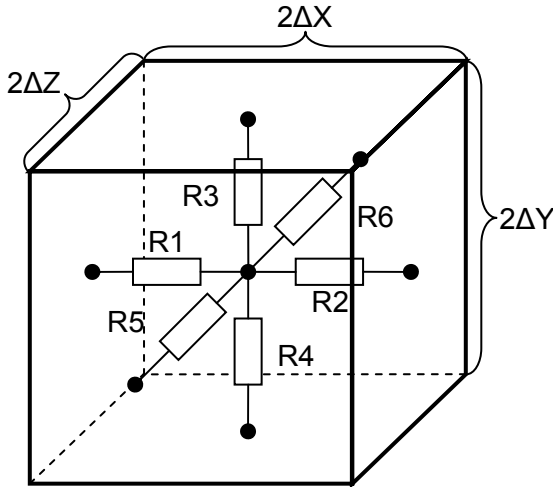


Рисунок 1.2 – Диференційний кубічний елемент

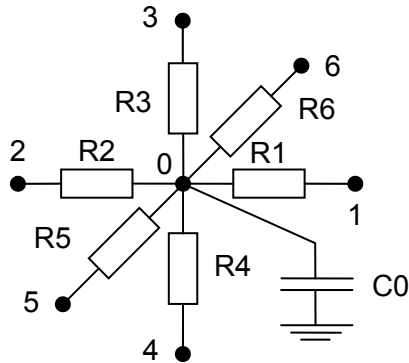


Рисунок 1.3 – Вузол об'ємної резистивно-ємнісної сітки

Додатковий струм викличе накопичення заряду на ємності C_0 . Величина заряду конденсатора пов'язана з потенціалом так:

$$V_0 = \frac{q_0}{C_0}, \text{ тобто } q_0 = C_0 \cdot V_0 \quad (1.24)$$

Диференціюючи це рівняння за часом, отримаємо:

$$\frac{dq_0}{dt} = C_0 \frac{dV_0}{dt} \quad (1.25)$$

Величина dq_0/dt , яка характеризує зміну заряду за одиницю часу, є струмом через конденсатор. Підставивши $C_0 \cdot dV_0/dt$ на місце ΣI_0 у виразі (1.23), ми отримаємо:

$$C_0 \frac{dV_0}{dt} = \frac{V_1 - V_0}{R_1} + \frac{V_2 - V_0}{R_2} + \frac{V_3 - V_0}{R_3} + \frac{V_4 - V_0}{R_4} + \frac{V_5 - V_0}{R_5} + \frac{V_6 - V_0}{R_6} \quad (1.26)$$

Враховуючи, що $R = R_1 = R_2 = R_3 = R_4 = R_5 = R_6$, скінченно-різницевою апроксимацію рівняння (1.26) запишемо так:

$$V_{1,t} + V_{2,t} + V_{3,t} + V_{4,t} + V_{5,t} + V_{6,t} - 6V_{0,t} = R \cdot C \cdot (V_{0,t+1} - V_{0,t}), \quad (1.27)$$

де $V_{i,t}$ – напруга в i -му вузлі в момент часу t .

Величини опорів для RC-сітки обирають пропорційними до відповідних елементарних ділянок середовища, значення ємностей необхідно вибирати пропорційними до розмірів цих ділянок.

1.1.3 Розрахунок параметрів електричного аналога диференційного рівняння

Для прикладу розрахуємо параметри комірки нерегулярної електричної сітки для моделювання двовимірного рівняння Фур'є. Фрагмент сітки зображений на рис. 1.4.

Прийнемо h – величина кроку між вузлами (наприклад h_i^{i+1} – величина кроку між i -м та $i+1$ -м вузлами, мм); $\lambda = 1,31$ Вт/(м·°К) – коефіцієнт теплопровідності середовища. Визначимо опори резисторів:

$$R_1 = \frac{h_j^{j+1}}{\lambda \cdot (h_i^{i+1} + h_{i+1}^{i+2})} = \frac{2 \cdot 10^{-3}}{1,31 \cdot (1 \cdot 10^{-3} + 3 \cdot 10^{-3})} = 0,382 \text{ °К/Вт},$$

$$R_3 = \frac{h_i^{i+1}}{\lambda \cdot (h_{j+1}^{j+2} + h_j^{j+1})} = \frac{1 \cdot 10^{-3}}{1,31 \cdot (1 \cdot 10^{-3} + 2 \cdot 10^{-3})} = 0,254 \text{ °К/Вт}.$$

Аналогічно $R_2=0,305$ (°К/Вт), $R_4=0,763$ (°К/Вт),
 $R_5=0,509$ (°К/Вт), $R_6=0,191$ (°К/Вт), $R_7=0,153$ (°К/Вт).

Далі визначимо величини ємностей конденсаторів:

$$C_1 = \lambda \cdot (h_j^{j+1} + h_{j+1}^{j+2}) \cdot (h_i^{i+1} + h_{i+1}^{i+2}) \cdot 10^{-3} \cdot B(x, y),$$

$$B(x, y) = \frac{C(x, y) \cdot \rho(x, y)}{\lambda}, \text{ тоді}$$

$$C_1 = (h_j^{j+1} + h_j^{j+2}) \cdot (h_i^{i+1} + h_{i+1}^{i+2}) \cdot C(x, y) \cdot \rho(x, y) \cdot 10^{-3},$$

де $\rho(x, y)$ – густина середовища (2800 кг/м³);

$C(x, y)$ – теплоємність одиниці маси (860 Дж/(кг·°K));

$\lambda(x, y)$ – коефіцієнт теплопровідності середовища.

$$C_1 = (2 + 1) \cdot 10^{-3} \cdot (1 + 3) \cdot 10^{-3} \cdot 860 \cdot 2800 \cdot 10^{-3} = 0,029 \text{ Дж/°K},$$

Аналогічно $C_2 = 0,036 \text{ (Дж/°K)}$.

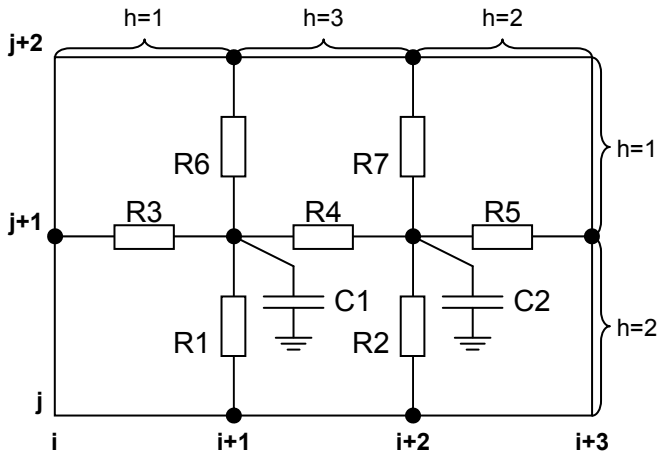


Рисунок 1.4 – Резистивно-ємнісна сітка

У розглянутому прикладі параметри сітки розраховуються для побудови електричного аналога рівняння теплопровідності, тобто це є тепловий опір, теплоємність. Відповідність електричних та теплових параметрів наведена в таблиці 1.1. При розрахунках електричних схем розмірності параметрів мають відповідати електричним величинам.

Таблиця 1.1 – Електричні та теплові величини-аналоги

Теплові	$R_T, \text{°K/Вт}$	$T, \text{°C}$	$P, \text{Вт}$	$C_T, \text{Дж/°K}$
Електричні	$R, \text{Ом}$	$U, \text{В}$	$I, \text{А}$	$C_e, \text{Ф}$

Напруги для краєвих умов визначають за допомогою виразу:

$$U_i = C \cdot f_i,$$

де f_i – значення шуканої функції в i -му вузлі;

C – постійний масштабний коефіцієнт.

У разі розв'язку рівняння теплопровідності краєвими умовами є значення температури у граничних вузлах.

1.2 Завдання до виконання лабораторної роботи

1.2.1 Отримати у викладача індивідуальне завдання (варіанти наведені у додатку А).

1.2.2 Побудувати електричний аналог заданого диференційного рівняння.

1.2.3 Розрахувати параметри елементів схеми.

1.2.4 Провести розрахунок отриманої електричної схеми за допомогою ППП “Circuit Maker” або подібного.

1.2.5 Обчислити у вузлах сітки значення шуканої функції (температури).

1.3 Зміст звіту

1.3.1 Мета роботи.

1.3.2 Теоретичні відомості у конспективній формі.

1.3.3 Індивідуальне завдання.

1.3.4 Побудова дискретної області моделювання.

1.3.5 Побудова електричного аналога диференційного рівняння в часткових похідних.

1.3.6 Результати розрахунку.

1.3.7 Висновки по роботі.

1.4 Контрольні питання

1.4.1 Види рівнянь в часткових похідних?

1.4.2 Назвіть основні види краєвих умов?

1.4.3 У чому полягає основний зміст першої похідної?

1.4.4 Який фізичний зміст другої похідної?

1.4.5 Як визначають параметри електричного аналога диференційного рівняння?

1.4.6 Особливості скінченно-різницевої апроксимації дво- та тривимірних стаціонарних та нестаціонарних рівнянь.

1.4.7 Особливості побудови диференційного кубічного елемента.

1.4.8 Методика побудови та розрахунку електричних схем за допомогою пакету Circuit Maker.

1.4.9 Електричні та теплові величини-аналоги.

2 ЛАБОРАТОРНА РОБОТА №2

Застосування методу скінчених різниць до розв'язку стаціонарних задач на компонентному рівні

Мета роботи: отримати практичні навички застосування методу скінчених різниць до розв'язку стаціонарних задач на компонентному рівні.

2.1 Теоретичні відомості

2.1.1 Вивід скінченно-різницевої схем

Обмежимося випадком двовимірних задач і рівномірними сітками. Ідея методу скінчених різниць полягає в заміні похідної її наближеним значенням, що виражається через розділені різниці функції в окремих дискретних точках. Координати точок (вузлів сітки) всередині області визначають значення індексів i та j . Різницеві рівняння записують для довільного вузла (i, j) , з використанням значень функції N у цьому та сусідніх вузлах. Прямокутна скінченно-різницева сітка показана на рис. 2.1.

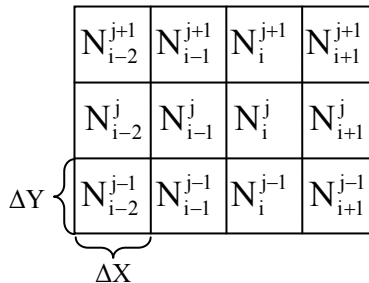


Рисунок 2.1 – Схема скінченно-різницевої сітки

Позначимо $N_i^j = N(x_0, y_0)$, тоді $N_{i+1}^j = N(x_0 + \Delta x, y_0)$, $N_{i-1}^j = N(x_0 - \Delta x, y_0)$. Похідна функції у точці (x_0, y_0) може бути визначена з виразу:

$$\frac{\partial N}{\partial x} = \lim_{\Delta x \rightarrow 0} \frac{N(x_0 + \Delta x, y_0) - N(x_0, y_0)}{\Delta x} \quad (2.1)$$

Якщо функція $N(x, y)$ неперервна, а Δx достатньо мале і

скінченне, то значення різниці $[N(x_0+\Delta x, y_0) - N(x_0, y_0)]/\Delta x$ буде близьким до значення похідної $\partial N/\partial x$. Оцінити похибку різницевої апроксимації можна шляхом розкладу функції N в ряд Тейлора. Якщо

$$N(x_0+\Delta x, y_0) = N(x_0, y_0) + \frac{\partial N}{\partial x} \Big|_{x_0, y_0} \Delta x + \frac{\partial^2 N}{\partial x^2} \Big|_{x_0, y_0} \frac{(\Delta x)^2}{2!} + \dots + \frac{\partial^n N}{\partial x^n} \Big|_{x_0, y_0} \frac{(\Delta x)^n}{n!} \quad (2.2)$$

то

$$\frac{\partial N}{\partial x} \Big|_{x_0, y_0} = \frac{N(x_0+\Delta x, y_0) - N(x_0, y_0)}{\Delta x} - \sum \frac{\partial^n N}{\partial x^n} * \frac{(\Delta x)^n}{n!}, n \geq 2 \quad (2.3)$$

Вираз (2.3) називають різницями вперед, або “правими” різницями. У компактнішому вигляді значення функції у вузлі (i, j) різницевої сітки має вигляд:

$$\frac{\partial N}{\partial x} \Big|_{i, j} = \frac{N_{i+1}^j - N_i^j}{\Delta x} + \Theta(\Delta x), \quad (2.4)$$

де $(N_{i+1}^j - N_i^j)/\Delta x$ – скінченно-різницева представлення похідної $(\partial N/\partial x)_{i, j}$;

$\Theta(\Delta x)$ – залишковий член.

Похибкою апроксимації називають різницю значень часткової похідної і її скінченно-різницевого представлення. За абсолютною величиною вона не перевищує $k|\Delta x|$ при $\Delta x \rightarrow 0$ (для достатньо малих Δx), причому $k > 0$ і є дійсною константою. Сам запис $\Theta(\Delta x)$ нічого не говорить про величину похибки, а лише вказує на її прямування до нуля. Впевнено можна сказати, що скінченно-різницеві схеми з похибкою апроксимації $\Theta/(\Delta x^2)$ мають вищу точність, ніж скінченно-різницеві схеми з похибкою апроксимації $\Theta(\Delta x)$. Хоча дане твердження для достатньо малих Δx безумовно правильне, але на практиці важко наперед визначити, яке Δx буде “достатньо малим”.

Взагалі можна побудувати велику множину скінченно-різницевих апроксимацій похідної $(\partial N/\partial x)_{i, j}$. Варіюючи кількість та розміщення вузлів шаблону, першу похідну можна представити з використанням різниць назад, яку називають “лівими” різницями:

$$N(x_0-\Delta x, y_0) = N(x_0, y_0) - \frac{\partial N}{\partial x} \Big|_{x_0, y_0} \Delta x + \frac{\partial^2 N}{\partial x^2} \Big|_{x_0, y_0} \frac{(\Delta x)^2}{2!} - \frac{\partial^3 N}{\partial x^3} \Big|_{x_0, y_0} \frac{(\Delta x)^3}{3!} + \dots \quad (2.5)$$

Після нескладних перетворень знайдемо вираз, який апроксимує похідну різницями назад:

$$\frac{\partial N}{\partial x} \Big|_{i,j} = \frac{N_i^j - N_{i-1}^j}{\Delta x} + \Theta(\Delta x) \quad (2.6)$$

Якщо від (2.3) відняти (2.6) отримаємо апроксимацію першої похідної центральними різницями:

$$\frac{\partial N}{\partial x} \Big|_{i,j} = \frac{N_{i+1}^j - N_{i-1}^j}{2 \Delta x} + \Theta(\Delta x) \quad (2.7)$$

У випадку суми виразів (2.2) і (2.5) знайдемо скінченно-різницеву апроксимацію другої похідної другого порядку:

$$\frac{\partial^2 N}{\partial x^2} \Big|_{i,j} = \frac{N_{i+1}^j - 2 N_i^j + N_{i-1}^j}{(\Delta x)^2} + \Theta(\Delta x)^2 \quad (2.8)$$

Більшість рівнянь в часткових похідних містять похідні 1-го та 2-го порядків. Для їх апроксимації намагаються використовувати не більше трьох вузлів сітки. Тому, на рівномірній сітці ($\Delta y = \Delta x = \text{const}$) найчастіше використовуються нижченаведені апроксимації. Для похідних 1-го порядку – (2.9)-(2.11), а для триточкових апроксимацій похідних 2-го порядку – (2.12)-(2.14). Наведені приклади не вичерпують усіх можливостей скінченно-різницевих апроксимацій похідних 1-го та 2-го порядків. В окремих випадках використовуються скінченно-різницеві схеми вищих порядків, при цьому необхідно витримувати постійний крок по всій скінченно-різницевій сітці.

$$\frac{\partial N}{\partial x} \Big|_{i,j} = \frac{N_{i+1}^j - N_i^j}{h} + \Theta(h) \quad (2.9)$$

$$\frac{\partial N}{\partial x} \Big|_{i,j} = \frac{N_i^j - N_{i-1}^j}{h} + \Theta(h) \quad (2.10)$$

$$\frac{\partial N}{\partial x} \Big|_{i,j} = \frac{N_{i+1}^j - N_{i-1}^j}{2 h} + \Theta(h) \quad (2.11)$$

$$\frac{\partial^2 N}{\partial x^2} \Big|_{i,j} = \frac{N_i^j - 2 N_{i+1}^j + N_{i+2}^j}{h^2} + \Theta(h^2) \quad (2.12)$$

$$\frac{\partial^2 N}{\partial x^2} \Big|_{i,j} = \frac{N_i^j - 2 N_{i-1}^j + N_{i-2}^j}{h^2} + \Theta(h^2) \quad (2.13)$$

$$\frac{\partial^2 N}{\partial x^2} \Big|_{i,j} = \frac{N_{i+1}^j - 2 N_i^j + N_{i-1}^j}{h^2} + \Theta(h^2) \quad (2.14)$$

2.1.2 Розв'язування задач методом скінченних різниць

Для ознайомлення з основними правилами використання методу скінченних різниць, розглянемо задачу Діріхле для рівняння Лапласа. Нехай необхідно розв'язати задачу Діріхле:

$$\frac{\partial^2 N}{\partial x^2} + \frac{\partial^2 N}{\partial y^2} = 0, \quad 0 < x < l, \quad 0 < y < l, \quad (2.15)$$

$N=0$ на верхній і бокових сторонах прямокутника.

$N(x, 0) = \sin(\pi x)$, $0 < x < l$, $y = 0$ на нижній стороні.

Побудуємо в площині (x, y) сітку, як показано на рис. 2.2.

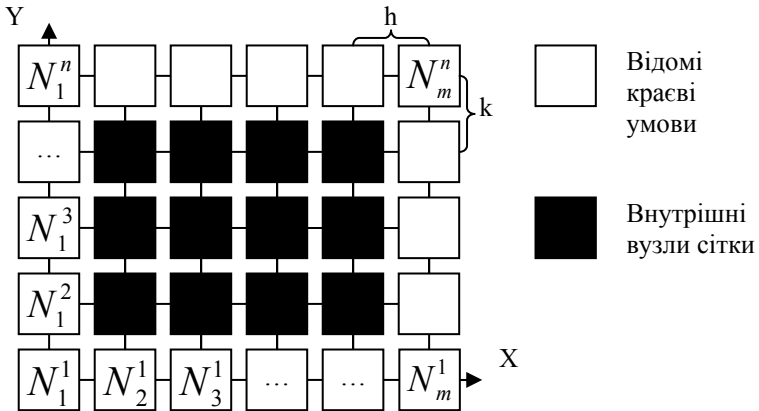


Рисунок 2.2 – Сітка для задачі Діріхле всередині прямокутника

Використаємо такі позначення:

$$N(x, y) = N_i^j \quad (2.16)$$

$$N(x, y - k) = N_i^{j-1} \quad (2.17)$$

$$N(x, y + k) = N_i^{j+1} \quad (2.18)$$

$$N(x + h, y) = N_{i+1}^j \quad (2.19)$$

$$N(x - h, y) = N_{i-1}^j \quad (2.20)$$

$$\frac{\partial N}{\partial x} = \frac{N_{i+1}^j - N_{i-1}^j}{2h} \quad (2.21)$$

$$\frac{\partial N}{\partial y} = \frac{N_i^{j+1} - N_i^{j-1}}{2k} \quad (2.22)$$

$$\frac{\partial^2 N}{\partial x^2} = \frac{N_{i+l}^j - 2N_i^j + N_{i-l}^j}{h^2} \quad (2.23)$$

$$\frac{\partial^2 N}{\partial y^2} = \frac{N_i^{j+l} - 2N_i^j + N_i^{j-l}}{k^2} \quad (2.24)$$

Основна ідея розв'язку задачі Діріхле базується на заміні часткових похідних у рівнянні Лапласа їх скінченно-різницеvim аналогом. Зробивши це і використавши представлені вище позначення, отримаємо таке різницеве рівняння для вузла (i,j):

$$\frac{N_i^{j+l} - 2N_i^j + N_i^{j-l}}{k^2} + \frac{N_{i+l}^j - 2N_i^j + N_{i-l}^j}{h^2} = 0, \quad (2.25)$$

де індекси i,j приймають значення індексів усіх внутрішніх вузлів.

У випадку, коли $h = k$ рівняння Лапласа зводиться до виразу:

$$N_i^{j+l} + N_i^{j-l} + N_{i+l}^j + N_{i-l}^j - 4N_i^j = 0 \quad (2.26)$$

Розв'язавши його відносно N_i^j , отримаємо:

$$N_i^j = \frac{N_i^{j+l} + N_i^{j-l} + N_{i+l}^j + N_{i-l}^j}{4} \quad (2.27)$$

У рівнянні (2.27) значення N_i^j беруться для усіх внутрішніх вузлів сітки. Відповідно до співвідношення (2.27) розв'язок N_i^j апроксимується середнім значенням розв'язку по чотирьох сусідніх точках. У результаті ми отримаємо систему лінійних алгебраїчних рівнянь, для розв'язку якої використовуються прямі методи (типу Гауса) або ітераційні.

2.1.3 Розв'язок систем лінійних алгебраїчних рівнянь

Методи розв'язання систем лінійних рівнянь є основою багатьох обчислювальних процедур. Для розв'язку таких систем використовуються прямі методи (наприклад метод виключення Гауса) або ітераційні (наприклад метод Лібмана). У даному випадку будемо розглядати один з прямих методів розв'язання – метод Гауса. Це один з кращих серед відомих алгоритмів, бо він базується на тому факті, що складання одного рівняння системи з іншим (можливо помноженим на константу), не змінює рішення системи. Метод передбачає здійснення двох етапів розв'язку: прямого ходу та зворотного.

Розглянемо матричне рівняння (2.28) та перепишемо його в координатній формі, позначивши елементи b_i вектора b через $a_{i,n+1}$. Це спрощує подальші позначення. Тоді система має вид:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = a_{1,n+1} \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = a_{2,n+1} \\ \dots\dots\dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = a_{n,n+1} \end{cases} \quad (2.28)$$

Розділимо перше рівняння на a_{11} та запишемо його:

$$x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 + \dots = a_{1,n+1}^{(1)},$$

де введено позначення $a_{12}^{(1)} = a_{12}/a_{11}$ і т.д. Помножимо це рівняння на $-a_{21}$ та складемо його з другим рівнянням. Коефіцієнти знову отриманого другого рівняння $a_{2j}^{(1)} = a_{2j} - a_{21} \cdot a_{1j}^{(1)}$ $j = 1, 2, \dots, n+1$. Такий вибір множника забезпечує рівність нулю коефіцієнта $a_{12}^{(1)}$. Аналогічно для інших рівнянь підстановка $n, i = 2, 3, \dots, a_{ij}^{(1)} = a_{ij} - a_{i1} \cdot a_{1j}^{(1)}$, $n+1, j = 1, 2, \dots$, забезпечує рівність нулю всіх коефіцієнтів у першому стовбці матриці A , за виключенням $a_{11}^{(1)}$, який дорівнює 1.

В результаті цих операцій рівняння будуть мати вигляд (індекс в дужках показує, що коефіцієнти були змінені один раз):

$$\begin{cases} x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 + \dots + a_{1n}^{(1)}x_n = a_{1,n+1}^{(1)} \\ a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2n}^{(1)}x_n = a_{2,n+1}^{(1)} \\ \vdots \\ a_{n2}^{(1)}x_2 + a_{n3}^{(1)}x_3 + \dots + a_{nn}^{(1)}x_n = a_{n,n+1}^{(1)} \end{cases}$$

Фактично, не потрібно обчислювати елемент, який стає нульовим. Елементи a_{i1} тепер не займають пам'ять ЕОМ, та обчислення виконуються, починаючи з $j = 2$.

На наступному кроці виключимо з розгляду перший рядок і стовбець та використаємо аналогічну процедуру для рівнянь від 2-го до n -го. Запишемо формули для обчислення нових значень коефіцієнтів:

$$a_{2j}^{(2)} = a_{2j}^{(1)} / a_{22}^{(1)}; \quad a_{ij}^{(2)} = a_{ij}^{(1)} - a_{i2}^{(1)} a_{2j}^{(2)}; \quad i = 3, 4, \dots, n, j = 3, 4, \dots, n+1.$$

Повторимо процедуру для всіх рядків матриці А. Якщо позначити $a_{ij}^{(0)} = a_{ij}$, то загальну формулу метода виключення Гауса можливо представити у такому вигляді:

$$a_{kj}^{(k)} = a_{kj}^{(k-1)} / a_{kk}^{(k-1)}; \quad a_{ij}^{(k)} = a_{ij}^{(k-1)} - a_{ik}^{(k-1)} a_{kj}^{(k)}; \quad (2.29)$$

$$k = 1, 2, \dots, n; i = k+1, \dots, n; j = k+1, \dots, n+1$$

В результаті система рівнянь приймає вигляд:

$$\left\{ \begin{array}{l} x_1 + a_{12}^{(1)} x_2 + a_{13}^{(1)} x_3 + \dots + a_{1n}^{(1)} x_n = a_{1,n+1}^{(1)} \\ x_2 + a_{23}^{(2)} x_3 + \dots + a_{2n}^{(2)} x_n = a_{2,n+1}^{(2)} \\ x_3 + \dots + a_{3n}^{(3)} x_n = a_{3,n+1}^{(3)} \\ \vdots \\ x_n = a_{n,n+1}^{(n)} \end{array} \right. \quad (2.30)$$

Для визначення вектора невідомих необхідно виконати зворотну підстановку. Остання складова x_n використовується в (n-1)-му рівнянні для визначення x_{n-1} і т.д. У загальному вигляді зворотну підстановку можливо записати так:

$$x_i = a_{i,n+1} - \sum_{j=i+1}^n a_{ij} x_j; \quad i = n-1, n-2, \dots, 1. \quad (2.31)$$

Тут верхні індекси не взяті до уваги.

В обчислювальній техніці прийнято оцінювати ефективність алгоритмів числом операцій, причому кожна операція являє собою комбінацію множення та віднімання. Можливо показати, що виключення по Гаусу потребує виконання $\sim n^3/3$ операцій (n – порядок матриці), а зворотну підстановку можна виконати за $\sim n^2/2$ операцій.

2.1.4 Алгоритм чисельного розв'язку задачі Діріхле (метод Лібмана)

КРОК 1. Присвоюємо величинам N_i^j у внутрішніх вузлах чисельне значення, яке дорівнює середньому арифметичному всіх значень межових умов (підвищення швидкості збіжності).

КРОК 2. Будемо перераховувати циклічно значення в усіх

внутрішніх вузлах сітки шляхом заміни старого значення середньозваженим по чотирьох сусідніх вузлах (див. вираз 2.25). Не дуже важливо, як буде організовано процес обчислень, проте зазвичай його проводять по рядках або стовбцях. Після декількох ітерацій значення у внутрішніх вузлах збіжаться з приблизним розв'язком задачі. Швидкість збіжності невисока, однак її можна підвищити декількома способами.

2.2 Завдання до виконання лабораторної роботи

2.2.1 Отримати у викладача індивідуальне завдання: тип задачі, номер варіанту та красві умови (див. додаток В).

2.2.2 Провести дискретизацію області розрахунку.

2.2.3 Побудувати скінченно-різницевий аналог диференційного рівняння в часткових похідних.

2.2.4 Програмно реалізувати розв'язок поставленої задачі методом Гауса та методом Лібмана.

2.2.5 Розв'язати отриману СЛАР у пакеті MathCad, порівняти отримані результати.

2.3 Зміст звіту

2.3.1 Мета роботи.

2.3.2 Теоретичні відомості у конспективній формі.

2.3.3 Індивідуальне завдання .

2.3.4 Побудова дискретної області моделювання.

2.3.5 Виведення скінченно-різницевого аналога диференційного рівняння в часткових похідних.

2.3.6 Побудова системи лінійних алгебраїчних рівнянь.

2.3.7 Схема алгоритму розв'язку системи різницевих рівнянь методом Гауса та методом Лібмана.

2.3.8 Текст програми.

2.3.9 Результати розрахунку.

2.4 Контрольні питання

2.4.1 У чому полягає основна ідея методу скінченних різниць.

2.4.2 У чому полягає основна ідея методу Лібмана.

2.4.3 Сутність методу Гауса.

2.4.4 Як розв'язати СЛАР у пакеті MathCad.

3 ЛАБОРАТОРНА РОБОТА №3

Застосування методу скінчених елементів до розв'язку стаціонарних задач на компонентному рівні

Мета роботи: отримати навички застосування методу скінчених елементів до розв'язку стаціонарних задач на компонентному рівні.

3.1 Теоретичні відомості

3.1.1 Сутність методу скінчених елементів

Метод скінчених елементів (МСЕ) є чисельним методом розв'язку диференційних рівнянь.

Основна ідея МСЕ полягає в тому, що будь-яку неперервну величину, таку, як температура, концентрація домішки, тиск та інші можна апроксимувати дискретною моделлю, яка будується на множині кусково-неперервних функцій, визначених на скінченній кількості підобластей. Кусково-неперервні функції визначаються за допомогою значень неперервної величини в скінченній кількості точок області моделювання.

В загальному випадку неперервна величина наперед невідома і необхідно поррахувати значення цієї величини в деяких точках області моделювання. Дискретну модель досить просто побудувати, якщо припустити, що числові значення цієї величини в кожній внутрішній точці області відомі. Після цього можна перейти до загального випадку. При застосуванні МСЕ до розв'язку фізичних задач на першому кроці моделювання необхідно побудувати дискретну модель неперервної величини. Дану процедуру проводять так:

- в області моделювання фіксується скінченна кількість точок. Ці точки називаються вузловими точками або просто вузлами;
- значення неперервної величини в кожній вузловій точці є змінною, яку необхідно визначити;
- область визначення неперервної величини розбивається на скінченну кількість під областей – елементів. Ці елементи мають спільні вузлові точки і в сукупності апроксимують форму області;
- неперервна величина апроксимується на кожному елементі поліномом, який визначається за допомогою вузлових значень цієї

величини. Для кожного елемента визначається свій поліном, але поліном підбирається таким чином, щоб зберігалась неперервність величини на границях елементів.

Основна ідея МСЕ може бути проілюстрована на одновимірному прикладі розподілу температури в стрижні (рис. 3.1). Розглядається неперервна величина $T(x)$, область визначення – відрізок OL в напрямку осі X .

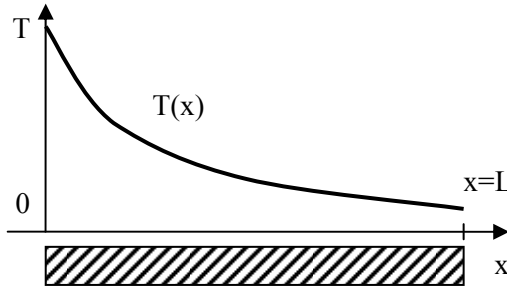


Рисунок 3.1 – Розподіл температури в одновимірному стрижні

Фіксовані та пронумеровані 5 точок на осі X (рис. 3.2). Це є вузлові точки (зовсім не обов'язково розміщувати їх на однаковій відстані одна від іншої). Очевидно, що можна розглядати і більше 5-ти точок, але їх достатньо, щоб проілюструвати основну ідею МСЕ.

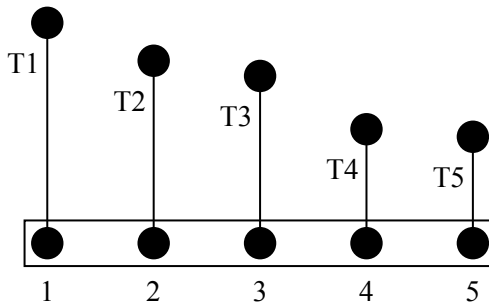
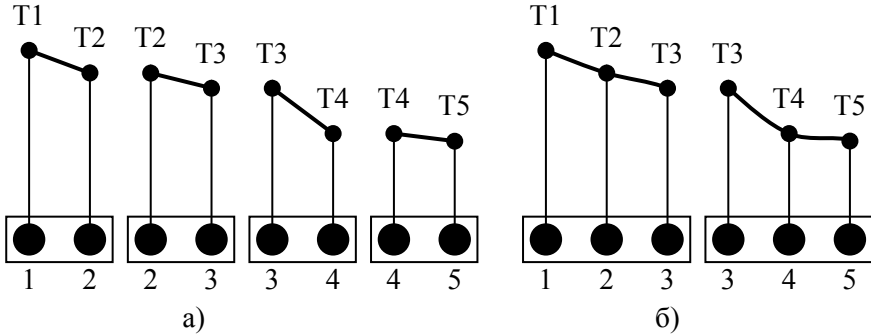


Рисунок 3.2 – Вузлові точки та передбачувані значення $T(x)$

Значення $T(x)$ в даному випадку відомі в кожній вузловій точці. Ці фіксовані значення приведені на рис. 3.2, б і позначені у відповідності з номерами вузлових точок через $T_1, T_2, \dots, T_5$. Розбиття області моделювання на елементи можна провести по

різному: перший спосіб - обмежити кожний елемент двома сусідніми вузловими точками, утворивши чотири елементи (рис. 3.3, а), або другий спосіб - розбити область моделювання на два елементи, кожний з яких містить три вузли (рис. 3.3, б).



а- перший спосіб; б- другий спосіб.

Рисунок 3.3 – Розбиття області моделювання на елементи

Відповідний елементу поліном визначається за значенням $T(x)$ у вузлових точках елемента. У випадку розбиття області моделювання на чотири елементи, коли на кожний елемент припадає по два вузли, функція елемента буде лінійною по X (дві точки однозначно визначають пряму лінію, $f = a_1 + a_2 \cdot X$). Апроксимація $T(x)$ по всьому стрижню буде складатися з чотирьох кусково-лінійних функцій, кожна з яких визначена на окремому елементі (рис. 3.4).

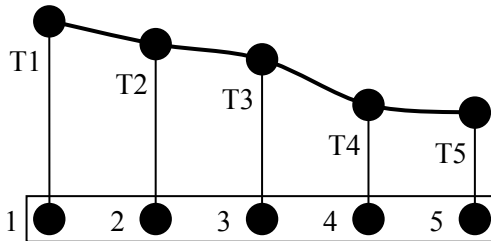


Рисунок 3.4 – Дискретна модель одновимірного температурного поля

Розбиття області моделювання на два елементи з трьома вузловими точками призводить до представлення функції елемента в формі поліному другого степеня ($f = a_1 + a_2 \cdot X + a_3 \cdot X^2$). В цьому випадку скінченною апроксимацією $T(x)$ буде сукупність двох кусково-неперервних квадратичних функцій.

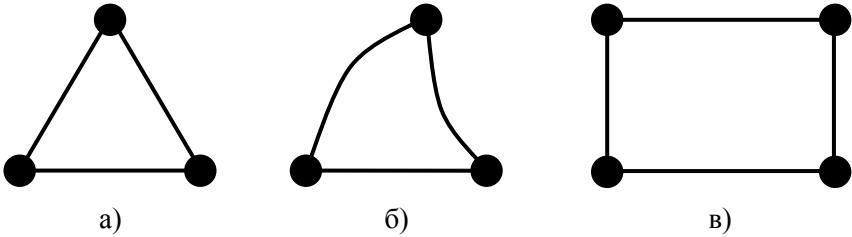
Необхідно зауважити, що це наближення буде тільки кусково-неперервним, оскільки кути нахилу графіків обох цих функцій можуть мати різні значення в третьому вузлі.

У загальному випадку розподіл температури є невідомим і нам необхідно визначити значення цієї величини в деяких вузлах. Методика побудови дискретної моделі залишається такою ж, як описано вище, але з додаванням ще одного додаткового кроку. Знову визначається множина вузлів і значення температури в цих вузлах $T_1, T_2, T_3, \dots$, які зараз є змінними, оскільки вони наперед невідомі. Область моделювання розбивається на елементи, на кожному з яких визначається відповідна функція елемента. Вузлові значення $T(x)$ мають бути тепер “відрегульовані” таким чином, щоб забезпечити найкраще наближення до реального розподілу температури. Це регулювання проводиться шляхом мінімізації деякої величини, пов’язаної з фізичним змістом задачі. Якщо розглядається задача розподілу тепла, то мінімізується функціонал, пов’язаний з відповідним диференціальним рівнянням. Процес мінімізації зводиться до розв’язку СЛАР відносно вузлових значень $T(x)$.

При побудові дискретної моделі неперервної величини визначеної в дво- або тривимірній області, основна концепція МСЕ використовується аналогічно. В двовимірному випадку елементи описуються такими функціями: $\varphi(X, Y) = a_1 + a_2 \cdot X + a_3 \cdot Y$ або $\varphi(X, Y) = a_1 + a_2 \cdot X + a_3 \cdot X + a_4 \cdot X^2 + a_5 \cdot X \cdot Y + a_6 \cdot Y^2$. Найчастіше розглядають трикутні (рис. 3.5 а), чотирикутні (рис. 3.5 в) або криволінійні (рис. 3.5 б) елементи. Функція елемента буде представлятися площиною, якщо для даного елемента взято мінімальну кількість вузлів, яка для трикутного елемента дорівнює трьом, а для чотирикутного – чотирьом.

Якщо використана кількість вузлів більша від мінімальної, то функції елемента буде відповідати криволінійна поверхня. До того ж, додаткова кількість вузлів дозволяє розглядати елементи з криволінійними границями. Скінченною апроксимацією двовимірної неперервної величини $\Phi(X, Y)$ буде сукупність кусково-неперервних поверхонь, кожна з яких визначається на окремому елементі за допомогою значень $\Phi(X, Y)$ у відповідних вузлових точках.

На рис. 3.6 наведені основні тривимірні скінчені елементи.



а- трикутний елемент; б- криволінійний елемент; в- чотирикутний елемент.

Рисунок 3.5 – Двовимірні скінчені елементи

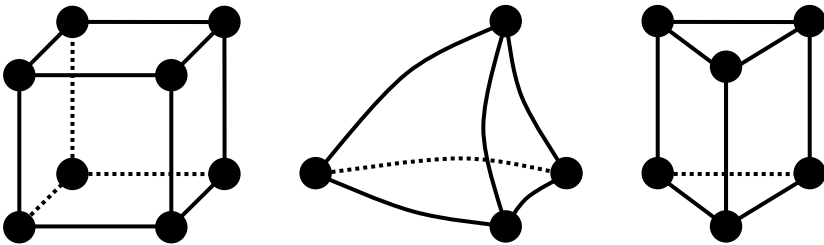


Рисунок 3.6 – Основні тривимірні скінчені елементи

Широке використання МСЕ обумовлене його перевагами над іншими чисельними методами. Зокрема до цих переваг можна віднести:

- властивості матеріалів суміжних елементів не обов'язково мають бути однаковими. Це дозволяє використовувати метод до тіл, складених з різних матеріалів;
- криволінійна область може бути апроксимована за допомогою прямолінійних елементів або описана точно за допомогою криволінійних елементів. Таким чином, метод можна використовувати не лише для областей з “хорошою” формою границь;
- розміри елементів можуть бути змінними. Це дозволяє збільшувати або зменшувати сітку розбиття області на елементи, якщо в цьому є необхідність;
- за допомогою МСЕ не виникає проблем при розгляді краєвих умов з розривним поверхневим навантаженням, а також змішаних краєвих умов.

НЕДОЛІК: Дуже громіздкі обрахунки навіть у випадку простих задач. Для розв'язку простих задач необхідно використовувати швидкодіючу ЕОМ з достатньо великим об'ємом оперативної пам'яті.

3.2 Завдання до виконання лабораторної роботи

3.2.1 Отримати у викладача індивідуальне завдання (див. додаток Д).

Дано:

- плоску 5-кутну область обмежену вузлами А, В, С, D, E;
- температуру в вузлах (°C): t_A, t_B, t_C, t_D, t_E ;
- координати точки F для пошуку невідомого значення

температури t_F .

3.2.2 Зобразити область в декартовій системі координат;

3.2.3 Розбити пластину на 3-лінійні трикутні елементи;

3.2.4 Пронумерувати глобальні вузли і отримані елементи з точки зору мінімізації ширини матриці коефіцієнтів;

3.2.5 Визначити порядок нумерації локальних вузлів кожного елемента;

3.2.6 Визначити ширину глобальної матриці коефіцієнтів;

3.2.7 Для дискретизованої області побудувати матрицю інтерполяційних поліномів вигляду $\varphi = N_i \cdot \Phi_i + N_j \cdot \Phi_j + N_k \cdot \Phi_k$;

3.2.8 Визначити температуру точки F із заданими координатами.

3.2.9 Реалізувати на ЕОМ розрахунок задачі (програма повинна здійснювати візуалізацію процесу розбиття заданої області на трикутники, визначати, до якого СЕ потрапила точка F, розраховувати температуру у точці F).

3.3 Зміст звіту

3.3.1 Мета роботи.

3.3.2 Теоретичні відомості в конспективній формі.

3.3.3 Індивідуальне завдання.

3.3.4 Виведення основних співвідношень для розрахунку поставленої задачі МСЕ.

3.3.5 Схема алгоритму та програма розрахунку задачі на ЕОМ.

3.3.6 Результати розрахунку.

3.4 Контрольні питання

- 3.4.1 В чому полягає основна ідея МСЕ?
- 3.4.2 Що таке глобальна і локальна нумерація вузлів?
- 3.4.3 Як впливає глобальна нумерація вузлів на ефективність подальших обчислень?
- 3.4.4 Що таке функція форми?
- 3.4.5 Які ви знаєте види скінченних елементів?
- 3.4.6 Що означає поняття алгебраїзація задачі МСЕ?
- 3.4.7 Як здійснюється об'єднання СЕ в ансамбль?
- 3.4.8 Сутність методу мінімізації функціоналу і методу Гальоркіна?
- 3.4.9 Алгоритм розв'язку фізичних задач методом скінченних елементів.

4 ЛАБОРАТОРНА РОБОТА №4

Розв'язання задачі параметричної ідентифікації об'єкта

Мета роботи: дослідження топологічної моделі біполярного транзистора та методики ідентифікації її первинних параметрів. Розробка програмного забезпечення для розрахунку параметрів моделі біполярного транзистора на основі даних ТУ.

4.1 Теоретичні відомості

4.1.1 Опис методики побудови математичних моделей

Загальна методика отримання математичних моделей складається з наступних етапів:

1) Визначення властивостей об'єкта, що підлягають відображенню в моделі. Вибір властивостей засновується на аналізі можливих застосувань моделі і визначає ступінь її унікальності.

2) Синтез структури математичної моделі об'єкта з точки зору його властивостей, що відображаються моделлю.

3) Розв'язання задачі ідентифікації чисельних значень внутрішніх параметрів моделі об'єкта для обраної структури.

4) Оцінка точності отриманої моделі об'єкта. Розв'язання цієї задачі виконується на тестових прикладах, що не використовувались при ідентифікації параметрів моделі.

5) Визначення області адекватності отриманої моделі. Розв'язання цієї задачі зводиться до визначення діапазону допустимих значень зовнішніх параметрів моделі, в межах зміни яких модель має допустиму похибку.

Застосування даної методики щодо математичного моделювання електричних процесів в біполярному транзисторі полягає в наступному:

1) Дослідження структури, фізичних процесів, що відбуваються в біполярному транзисторі, принципу його дії і схем включення, а також основних параметрів і характеристик.

2) На основі дослідженої структури транзистора і його властивостей, пов'язаних з електричним функціонуванням транзистора, може бути запропонована структура його топологічної моделі у вигляді еквівалентної заступної схеми, яку наведено на рис. 4.1. З точки зору універсальності ця топологічна модель

транзистора може застосовуватись для виконання проектних процедур одноваріантного аналізу електричних процесів в статичному, динамічному і частотному діапазонах.

В підрозділі 4.2 і на рис. 4.1 наведено опис параметрів віток моделі, а також визначені первинні параметри моделі.

3) Одним з способів визначення первинних параметрів топологічної моделі транзистора є методика ідентифікації параметрів моделі на основі даних технічних умов на транзистор. Цей спосіб є різновидом експериментального підходу до ідентифікації параметрів моделей і докладно викладений в підрозділі 4.3.

4) Для визначення точності і адекватності моделі транзистора, що розглядається, необхідно проведення експериментальних вимірів вихідних параметрів і характеристик його основних схем включення. При порівнянні експериментальних і розрахункових даних можна встановити максимальну похибку моделі і область її адекватності.

4.1.2 Опис топологічної моделі біполярного транзистора

На рис. 4.1 наведена топологічна модель біполярного транзистора і опис віток моделі. Параметрами лінійних віток моделі є:

R_E , R_B , R_K – активні опори ділянок емітера, бази, колектора транзистора.

Параметри і потокові змінні нелінійних віток моделі описуються співвідношеннями, наведеними на рис. 4.1, а їх первинними параметрами є:

- I_{TE} , I_{TK} – теплові струми (струми насичення) емітерного і колекторного переходів;
- $m\varphi_{TE}$, $m\varphi_{TK}$ – температурні потенціали переходів з урахуванням технологічних поправочних коефіцієнтів;
- $C_{БПЕО}$, $C_{БПКО}$ – бар'єрні ємності переходів при нульовому зміщенні на переході;
- U_{ZE} , U_{ZK} – контактні різниці потенціалів при нульовому зміщенні на переходах;
- τ_N , τ_I – час життя неосновних носіїв в базі для нормального і інверсного включення транзистора.

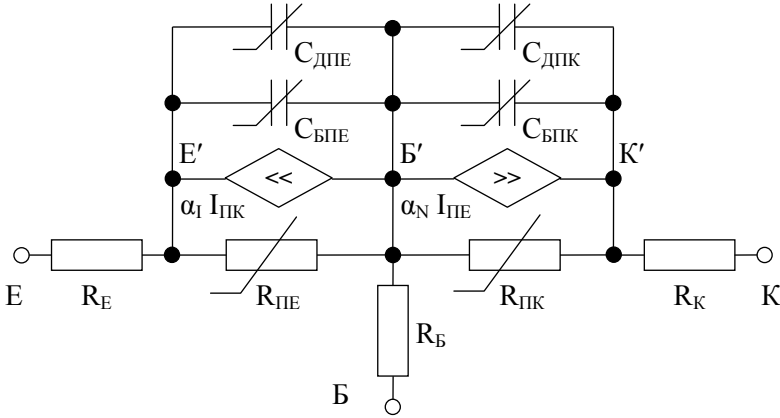


Рисунок 4.1 – Топологічна модель біполярного транзистора

$$I_{RПП} = I_{ПЕ}; I_{RПП} = I_{ПК}; \alpha_I = \frac{\beta_I}{1 + \beta_I}; \alpha_I = \frac{\beta_N}{1 + \beta_N};$$

$$I_{ПЕ} = A \cdot I_{TE} \cdot \left[e^{\frac{A \cdot v_{ПЕ}}{m \cdot \varphi_{TE}}} - 1 \right]; I_{ПК} = A \cdot I_{TK} \cdot \left[e^{\frac{A \cdot v_{ПК}}{m \cdot \varphi_{TK}}} - 1 \right];$$

$$C_{БПЕ} = \frac{C_{БПЕ0}}{\sqrt{1 - \frac{A \cdot v_{ПЕ}}{v_{ZE}}}}; C_{БПК} = \frac{C_{БПК0}}{\sqrt{1 - \frac{A \cdot v_{ПК}}{v_{ZK}}}};$$

$$C_{ДПК} = \frac{\tau_I \cdot I_{TK}}{2 \cdot m \cdot \varphi_{TK}} \cdot e^{\frac{A \cdot v_{ПК}}{m \cdot \varphi_{TK}}}; \beta_N = \beta_{N0} (1 + a \cdot v_{KE});$$

$$v_{ПЕ} = \varphi_{E'} - \varphi_{B'}; v_{ПК} = \varphi_{K'} - \varphi_{B'};$$

$A = +1$ для транзистора p-n-p типу;

$A = -1$ для транзистора n-p-n типу.

Первинними параметрами залежних джерел струму, які керують струмами переходів, є:

β_{NO} , β_I – коефіцієнти підсилення транзистора по струму для схеми зі спільним емітером (СЕ) при нормальному і інверсному його включенні;

a – коефіцієнт залежності від режиму (напруги U_{KE}).

В наведеній моделі транзистора урахування температурної залежності теплових струмів I_T і температурних потенціалів переходів здійснюється таким чином:

$$I_T = I_{TO} \cdot \frac{T}{T_{PO}} \cdot e^{k_{I_T} \cdot \frac{1}{T_{PO}} - \frac{1}{T_P}}, \quad (4.1)$$

$$m\varphi_T = m\varphi_{TO} \cdot \frac{T_P}{T_{PO}}, \quad (4.2)$$

де I_{TO} – тепловий струм при $T_{PO} = 298^\circ\text{K}$;

$m\varphi_{TO}$ – температурний потенціал при $T_{PO} = 298^\circ\text{K}$;

k_{I_T} – коефіцієнт температурної залежності I_T переходу.

Звідси, первинними параметрами топологічної моделі біполярного транзистора є: $R_E, R_B, R_K, I_{TEO}, I_{TKO}, m\varphi_{TEO}, m\varphi_{TKO}, C_{БПЕО}, C_{БПКО}, U_{ZK}, U_{ZK}, \tau_N, \tau_I, \beta_{N0}, a, \beta_B, k_{I_T}$.

4.1.3 Методика ідентифікації первинних параметрів моделі транзистора

Для визначення первинних параметрів топологічної моделі транзистора необхідні дані параметрів за ТУ і режими їх вимірювання, які приведені у додатку Е.

4.1.3.1 Визначення початкових значень параметрів моделі емітерного переходу транзистора

Визначення параметра $m\varphi_{TEO}$: значення параметра приймається рівним середньому значенню при $T = 298^\circ\text{K}$:

а) для германієвих транзисторів:

$$0.026B < m\varphi_{TEO} < 0.032B, m\varphi_{TEO} = 0.029B; \quad (4.3)$$

б) для кремнієвих і арсенідгалієвих транзисторів:

$$0.03B < m\varphi_{TEO} < 0.04B, m\varphi_{TEO} = 0.035B. \quad (4.4)$$

Визначення параметра β_{N0} : коефіцієнт підсилення по струму в схемі з СЕ приймається рівним:

$$\beta_{N0} = (\beta_{Nmin} + \beta_{Nmax})/2, \quad (4.5)$$

де $\beta_{Nmin}, \beta_{Nmax}$ – мінімальне і максимальне значення за ТУ.

Визначення параметра I_{TEO} : проводиться I_{TEO} на початковій

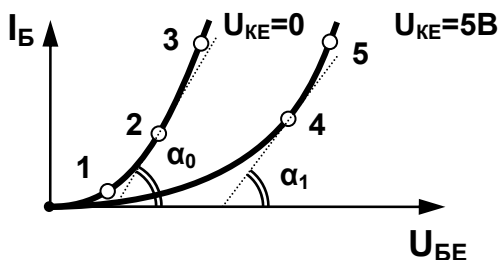
ділянці вхідної характеристики транзистора $I_B(U_{BE})$ при $U_{KE} = 0$ в точці 1 на середині нелінійної ділянки (рис. 4.2а). В цій точці

$$I_E = I_B \cdot (\beta_{NO} + 1), \quad (4.6)$$

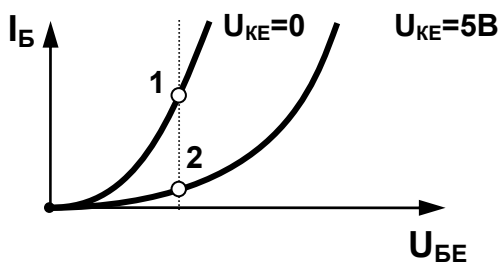
$$I_B \approx I_{PE} = I_{TEO} \cdot \left[e^{\frac{U_{PE}}{m \varphi_{TEO}}} - 1 \right], \quad (4.7)$$

де $U_{PE} \approx U_{BE}$. Тоді

$$I_{TEO} = \frac{I_B \cdot (\beta_{NO} + 1)}{e^{\frac{U_{BE}}{m \varphi_{TEO}}} - 1}, \quad (4.8)$$



а)



б)

Рисунок 4.2 – Визначення параметрів моделі транзистора за вхідними характеристиками

Визначення параметра R_E : об'ємний опір ділянки емітера визначається по вхідним характеристикам (див. рис. 4.2а).

$$R_E = \frac{ctg\alpha_1 - ctg\alpha_0}{\beta_{N0} + 1} \quad (4.9)$$

де

$$ctg\alpha_0 = \frac{\Delta v_{BE}}{\Delta I_B} = \frac{v_{BE3} - v_{BE2}}{I_{B3} - I_{B2}}, \quad (4.10)$$

$$ctg\alpha_1 = \frac{v_{BE5} - v_{BE4}}{I_{B5} - I_{B4}}, \quad (4.11)$$

де $v_{BE2}, \dots, v_{BE5}, I_{B2}, \dots, I_{B5}$ – значення напруги U_{BE} і струму I_B у відповідних точках на лінійних ділянках ВАХ.

Визначення параметра R_B – об’ємний опір базової ділянки визначається двома способами, а потім його остаточне значення визначається шляхом усереднення.

1) визначення R_B на основі напруги і струму насичення бази:

$$R_{B1} = \frac{U_{BE\text{ нас}}}{I_{B\text{ нас}}}, \quad (4.12)$$

де $U_{BE\text{ нас}}, I_{B\text{ нас}}$ – напруга і струм насичення бази.

2) визначення R_B на основі h-параметрів транзистора:

$$R_{B2} = h_{11E}, \text{ або } R_{B2} = \frac{h_{11B}}{1 - h_{21B}} \quad (4.13, 4.14)$$

де h_{11E}, h_{11B} – вхідні опори транзистора для режиму малого сигналу в схемі з СЕ і СБ;

h_{21B} – коефіцієнт підсилення транзистора в схемі з СБ.

Тоді

$$R_B = (R_{B1} + R_{B2})/2 \quad (4.15)$$

Визначення параметра a – параметр a лінійної залежності коефіцієнта підсилення (в схемі з СЕ) від режиму $\beta_N(v_{KE})$ визначається на основі вхідних характеристик транзистора (див. рис. 4.2б).

$$I_{E1} = I_{B1} \cdot (\beta_{N0} + 1), \quad I_{E2} = I_{B2} \cdot (\beta_N + 1) \quad (4.16, 4.17)$$

Оскільки $I_{E1} \approx I_{E2}$, тому:

$$I_{B1} \cdot (\beta_{N0} + 1) = I_{B2} \cdot (\beta_N + 1), \quad (4.18)$$

$$\beta_N = (\beta_{N0} + 1) \cdot \frac{I_{B1}}{I_{B2}} - 1, \quad (4.19)$$

де I_{B1} , I_{B2} – значення струму I_B при $U_{BE} = \text{const}$ в точках 1 та 2 відповідно, при $U_{KE} = 0$ і $U_{KE} \neq 0$.

Тоді $\Delta\beta_N = \beta_N - \beta_{N0}$, $\Delta v_{KE} = v_{KE}$ при $U_{BE} = \text{const}$, а отже параметр a дорівнюватиме:

$$a = \frac{\Delta\beta_N}{v_{KE}} \cdot \frac{I}{\beta_{N0}}, \quad (4.20)$$

4.1.3.2 Визначення параметрів моделі емітерного переходу транзистора

Як експериментальні дані для ідентифікації параметрів емітерного переходу транзистора, початкові значення яких були визначені в п. 4.1.3.1, використовується вхідна характеристика транзистора $I_B(U_{BE})$ при $U_{KE} \neq 0$.

Ідентифікація параметрів транзистора здійснюється шляхом параметричної оптимізації залежності $U_{BE} = f(I_B)$, яка визначається наступним розрахунковим співвідношенням:

$$U_{BE} = U_{RE} + U_{PE} + U_{RB} \quad (4.21)$$

або

$$U_{BE} = I_B \cdot (\beta_N + 1) \cdot R_E + m \varphi_{TEO} \cdot \ln \left[\frac{I_B \cdot (\beta_N + 1)}{I_{TEO}} + 1 \right] + I_B \cdot R_B, \quad (4.22)$$

де $\beta_N = \beta_{N0} \cdot (1 + a \cdot v_{KE})$.

Варійованими параметрами залежності (4.22) є параметри β_{N0} , $m \varphi_{TEO}$, R_B та I_{TEO} , діапазон допустимих значень яких визначається нерівностями:

$$\beta_{N_{min}} \leq \beta_{N0} \leq \beta_{N_{max}} \quad (4.23)$$

$$0.026V \leq m \varphi_{TEO} \leq 0.032V, \text{ для Ge-транзисторів} \quad (4.24)$$

або

$$0.03V \leq m \varphi_{TEO} \leq 0.04V, \text{ для Si, GaAs-транзисторів} \quad (4.25)$$

$$R_{B1} \leq R_B \leq R_{B2}, \text{ чи навпаки, якщо } R_{B1} > R_{B2}; \quad (4.26)$$

$$0.1 \cdot I_{TEO} < I_{TEO} < 10 \cdot I_{TEO} \quad (4.27)$$

Як критерій оптимізації використовується цільова функція, яка має вигляд:

$$F(\bar{q}) = \max_{i \in [1:n]} |\varepsilon_i|, \quad (4.28)$$

де

$$\varepsilon_i = \frac{v_{BE_i}^{дiйс} - v_{BE_i}^{розн}}{v_{BE_i}^{дiйс}} \quad (4.29)$$

$U_{BE_i}^{дiйс}, U_{BE_i}^{розн}$ – дійсне (задане в ТУ) та розрахункове i -те значення напруги U_{BE} ;

n – кількість точок на вхідній характеристиці транзистора $U_{BE}(I_{BE})$ при $U_{KE} \neq 0$ (від 6 до 10 точок);

$\bar{q}$ – вектор варійованих параметрів.

В процесі **параметричної оптимізації** здійснюється мінімізація цільової функції (4.28), а процес оптимізації завершується при максимальній відносній похибці $\varepsilon_{i \max} \leq 10\%$.

Для виконання оптимізації необхідно розробити підпрограму ОПТИМ, яка реалізує метод найскорішого спуску.

Вхідними даними ОПТИМ є: $m\varphi_{TE \min}, m\varphi_{TE \max}, \beta_{N \min}, \beta_{N \max}, I_{TE0}, R_E, a, R_{B1}, R_{B2}, U_{KE}, \varepsilon_{ДОП}, n, v_{BE_i}^{дiйс}, I_{B_i} (i = \overline{1, n})$.

Вихідними даними ОПТИМ є: $\beta_{N0}, m\varphi_{TE0}, I_{TE0}, R_B, \varepsilon_{\max}$.

4.1.3.3 Визначення параметрів моделі колекторного переходу транзистора

Визначення параметра R_K – об'ємний опір ділянки колектора визначається двома способами, а остаточне значення визначається шляхом їх усереднення.

1) визначення R_K по вихідним характеристикам через тангенс куту нахилу прямої 0А (рис. 4.3):

$$R_{K1} = \frac{I}{tg\alpha} = \frac{U_A}{I_A} \quad (4.30)$$

2) визначення R_K по струму і напрузі насичення колектора:

$$R_{K2} = \frac{U_{KE \text{ нас}}}{I_{K \text{ нас}}} \quad (4.31)$$

де $U_{KE \text{ нас}}, I_{K \text{ нас}}$ – напруга і струм насичення колектора.

Значення R_K розраховується як:

$$R_K = (R_{K1} + R_{K2})/2. \quad (4.32)$$

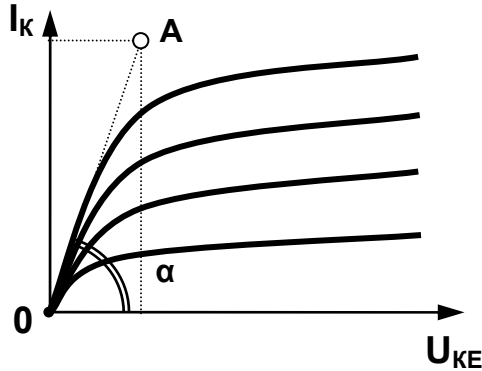


Рисунок 4.3 – Визначення параметрів моделі транзистора за вихідними характеристиками

Визначення параметра $m\varphi_{TKO}$ – значення параметра $m\varphi_{TKO}$ приймається рівним значенню $m\varphi_{TEO}$, розрахованому в п. 4.5.3.2.

Визначення параметра β_I – значення параметра приймається рівним типовому середньому значенню:

$$4 \leq \beta_I \leq 10, \beta_I = 7.$$

Визначення параметра I_{TKO} – параметр I_{TKO} визначається на основі рівності:

$$\beta_I I_{TKO} = \beta_{N0} I_{TEO} \quad (4.33)$$

Тоді

$$I_{TKO} = \beta_{N0} I_{TEO} / \beta_I \quad (4.34)$$

Визначення параметрів τ_N і τ_I – час життя неосновних носіїв в базі; τ_N – час життя неосновних носіїв в базі при нормальному включенні транзистора, визначається двома способами, а остаточне значення – шляхом усереднення:

1) визначення на основі граничної частоти:

$$\tau_{N1} = 1 / (2 \cdot \pi \cdot f_{TP}), \quad (4.35)$$

де f_{TP} – гранична частота передачі струму в схемі з СЕ.

2) визначення на основі модуля коефіцієнта передачі і частоти його визначення:

$$\tau_{N2} = 1 / (2 \cdot \pi \cdot |h_{21E}| \cdot f_m), \quad (4.36)$$

де $|h_{21E}|$ – модуль коефіцієнта передачі струму в схемі з СЕ;
 f_m – частота визначення $|h_{21E}|$.

Середнє значення τ_N дорівнює:

$$\tau_N = (\tau_{N1} + \tau_{N2})/2 \quad (4.37)$$

Час життя неосновних носіїв в базі при інверсному включенні τ_I :

$$\tau_I = 3 \cdot \tau_N \quad (4.38)$$

Визначення параметрів U_{ZE} і U_{ZK} – значення контактних різниць потенціалів емітерного і колекторного переходів; приймаються рівними типовим значенням:

$$U_{ZE} = U_{ZK} = 5.2\text{В.}$$

Визначення параметрів $C_{БПЕО}$ і $C_{БПКО}$ – бар'єрні ємності переходів при нульовому змаченні на них, визначаються із співвідношень:

$$C_{БПЕО} = C_E \sqrt{1 + \frac{v_{BE}}{v_{ZE}}}, \quad (4.39)$$

$$C_{БПКО} = C_K \sqrt{1 + \frac{v_{KB}}{v_{ZK}}}, \quad (4.40)$$

де C_E , C_K – значення ємностей переходів при напругах на переходах U_{EB} , U_{KB} .

Визначення параметра k_{I_T} – значення коефіцієнта температурної залежності теплового струму емітерного і колекторного переходів дорівнює:

для Ge транзисторів – 7700°K,

для Si транзисторів – 12900°K,

для GaAs транзисторів – 16170°K.

4.2 Завдання на виконання роботи

4.2.1 Вивчити методику побудови математичної моделі об'єкту.

4.2.2 Дослідити методику ідентифікації первинних параметрів топологічної моделі транзистора на основі даних ТУ та побудувати схему алгоритму її програмної реалізації.

4.2.3 Розробити програму ідентифікації первинних параметрів моделі транзистора на основі отриманого в п.4.2.2 алгоритму з

урахуванням наступних вимог:

- діалогове введення вхідних даних;
- розрахунок початкових параметрів моделі емітерного переходу, а також оптимізованих їх значень;
- розрахунок первинних параметрів моделі колекторного переходу;
- вхідні дані та результати розрахунків мають бути представлені у вигляді відповідних таблиць.

4.2.4 Для запропонованого викладачем варіанту (типу транзистора та даних ТУ з додатку Ж) виконати розрахунок первинних параметрів моделі транзистора.

4.3 Зміст звіту

4.3.1 Мета роботи.

4.3.2 Короткі теоретичні відомості.

4.3.3 Вхідні дані в залежності від варіанту.

4.3.4 Схема алгоритму програми ідентифікації параметрів топологічної моделі транзистора на основі даних ТУ.

4.3.5 Текст програми.

4.3.6 Результати розрахунків первинних параметрів топологічної моделі біполярного транзистора у вигляді таблиці.

4.3.7 Висновки по результатам роботи.

4.4 Контрольні питання

4.4.1 Особливості побудови математичних моделей COiC на макрорівні.

4.4.2 Компонентні та топологічні рівняння на макрорівні.

4.4.3 Топологічна модель біполярного транзистора і її опис.

4.4.4 Сутність методики ідентифікації первинних параметрів моделі транзистора на основі даних ТУ і її особливості.

4.4.5 Характеристика методів безумовної оптимізації.

4.4.6 Характеристика методів умовної оптимізації.

4.4.7 Порівняльна характеристика розрахункового та експериментального методів ідентифікації параметрів моделей.

4.4.8 Сутність методу найскорішого спуску.

5 ЛАБОРАТОРНА РОБОТА №5

Розробка та дослідження імітаційних моделей виробничого процесу

Мета роботи: вивчення системи імітаційного моделювання Service Model та практичне застосування цієї системи для моделювання реального технологічного процесу виготовлення деталей РЕА.

5.1 Теоретичні відомості

5.1.1 Системи масового обслуговування

Існує багато виробничих та економічних ситуацій, пов'язаних з процесами масового обслуговування клієнтів. Наприклад, за обмежений час треба обслуговувати покупців магазину, клієнтів сфери обслуговування, прийняти заявки на ремонтні роботи, виконати виробничу програму і т.д. При прийнятті рішень в таких ситуаціях, що залежать від випадкових факторів, часто виявляється неможливим встановити аналітичні залежності між різними факторами. В цих випадках доводиться штучно створювати випадкові процеси, що подібні до тих, які мають місце на практиці та можуть бути легко досліджені. Теорія масового обслуговування використовується в тих ситуаціях, де необхідне рішення визначається випадковими факторами чи обставинами. Тобто такими, які можуть приймати різні, раніше невідомі значення. Теорія масового обслуговування дає можливість врахувати ці випадковості в процесах, пов'язаних з потоками вимог (замовлень, обставин) на обслуговування.

В результаті розвитку теорії масового обслуговування були створені спеціальні програмні пакети, що дозволяють моделювати певні випадкові процеси, а на основі цих процесів створювати моделі різних систем масового обслуговування.

Система імітаційного моделювання – це система, що допомагає автоматизувати процес створення та дослідження моделі системи масового обслуговування. Найбільш сучасною версією такої програми є Microsoft Windows система **Service Model**, розроблена спеціально для моделювання сервісних систем, за допомогою якої можна ефективно вирішувати реальні задачі.

5.1.2 Призначення, особливості та функціональні можливості програми

Система *Service Model* дозволяє промодельовувати ситуацію, що виникне на підприємстві при реорганізації виробничого процесу; проаналізувати доцільність певних робочих потоків; правильно спланувати існуючі на підприємстві ресурси; визначити наскільки потрібна буде автоматизація офісу і чи варто її взагалі проводити при даному обсязі робіт, певній кількості клієнтів, обслуговуючого персоналу, а також керівного складу. За допомогою *Service Model* можна також провести аналіз ефективності виробничої діяльності, спланувати зміни на існуючому підприємстві, використовуючи найсміливіші проекти та рішення, а потім з отриманих варіантів вибрати найоптимальніший.

Реальні задачі, які дозволяє моделювати та аналізувати *Service Model*:

Process Reengineering – реорганізація виробництва;

Work Flow Analysis – аналіз робочих потоків;

Facility Planning – планування ресурсів;

Office Automation – автоматизація офісу;

Capacity Analysis – аналіз продуктивності;

Expansion Planning – планування розширення.

Service Model є системою, що призначена для імітації процесів на робочих місцях у вигляді обслуговування різноманітних об'єктів та при необхідності черг цих об'єктів.

Об'єктами обробки можуть бути певні документи, промислові вироби, клієнти і т. ін. Потоки вхідних об'єктів визначаються потужністю, часом та частотою появи об'єктів в моделі.

Процеси на робочих місцях підпорядковуються певній зазначеній при проектуванні логіці, яка може відображати, як взаємодію вхідних об'єктів, так і умови передачі цих об'єктів на подальші робочі місця. Системою моделювання підтримуються зв'язки між робочими місцями, що відображають їх відносини, а також враховуються ресурси, які необхідні для виконання робочих процесів. Ресурси – це не місця обробки і не об'єкти обробки вони потрібні для більш правильного відображення реальних процесів, що моделюються. Ресурсами найчастіше є люди, обладнання, програмне забезпечення і т. ін.

5.1.3 Опис інтерфейсу програми

Майже всі функції, що виконує програма, містяться в головному меню (рис. 5.1), яке з'являється у верхній частині екрану одразу після завантаження *Service Model* і залишається там весь час роботи програми.

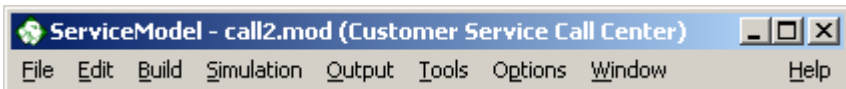


Рисунок 5.1 – Головне меню системи Service Model.

Пункт меню **File**:

- New* – новий файл для створення нової моделі;
- Open* – відкрити існуючу модель;
- Merge* – з'єднання моделей;
- Save* – запам'ятати модель;
- Save As...* – запам'ятати модель з новим ім'ям;
- View Text* – перегляд параметрів моделі;
- Print Text* – друк параметрів моделі;
- Print Setup* – настройка принтера;
- Create Model Package...* – створення модельного пакету;
- Install Model Package...* – установка модельного пакету;
- Exit* – вихід.

Пункт меню **Edit**:

- Delete* – видалити;
- Insert* – вставити;
- Append* – додати;
- Move* – перенести;
- Move to* – перенести в...

Пункт меню **Build**:

- Locations* – робочі місця;
- Entities* – об'єкти обробки;
- Path Networks* – шляхи переміщення;
- Resources* – ресурси;
- Processing* – процеси на робочих місцях;
- Arrivals* – параметри вхідних потоків;
- Shifts* – регламенти;
- Define* – опис регламенту;

Assign – призначення регламенту;
More elements – додаткові елементи;
Attributes – атрибути;
Variables(global) – глобальні змінні;
Macros – макроси;
Subroutines – підпрограми;
Arrival cycles – цикли вхідних потоків;
Table function – табличні розподілення;
User definitions – визначення користувача;
External files – зовнішні файли;
General Information – головна інформація про модель;
Background Graphics – фонові графіки;
Front of Grid – перед умовними позначеннями;
Behind Grid – за умовними позначеннями.

Пункт меню **Simulation:**

Run – запуск моделі;
Save & Run – запам'ятати та запустити модель;
Options – опції роботи моделі;
Model parameters – параметри моделі;
Scenarios – сценарії.

Пункт меню **Output:**

View statistics – перегляд статистики;
View trase – перегляд трасування.

Пункт меню **Tools:**

Graphc editor – графічний редактор;
Expression search – пошук;
Find – знайти;
Replase – замінити;
Search next – знайти наступне.

Пункт меню **Options:**

Autobild – режим автоматичного завдання параметрів моделі;
Show grid – відобразити сітку;
Snap to grid – прив'язка елементів моделювання до сітки;
Show networkcs – відобразити шляхи переміщення;
Show routing paths – відобразити напрямки руху;
Zoom – робота з вікном;
Views – відображення фіксованих елементів моделі;

Model defaults – настроювання параметрів моделі;

Editing preferenses – налаштування параметрів інтерфейсу користувача;

Directories – настроювання директорій;

Refrech layout – оновити поле моделі;

Reset window positions – скинути віконні настройки моделі;

Save settings now – зберегти настройки моделі зараз;

Save settings on exit – зберегти настройки моделі на виході.

Пункт меню **Window:**

Tile – розмістити вікна “черепицею”;

Cascade – розмістити вікна “каскадом”;

Arrange icons – систематизувати іконки.

5.1.4 Основні поняття системи моделювання

Основними поняттями системи моделювання ***Service Model*** є:

- об’єкти обробки (***entities***);
- параметри вхідних потоків(***arrivals***);
- робочі місця (***locations***);
- ресурси (***resources***);
- шляхи переміщення (***path networks***);
- процеси на робочих місцях (***processing***);
- опис процесу (***operation***);
- зв’язки між робочими місцями (***routing***);

Модель будь-якого виробничого процесу може бути представлена у вигляді сукупності робочих місць, пов’язаних з ними ресурсів, об’єктів, що обробляються на цих місцях та логічних відносин між робочими місцями. Створення моделі неможливе без побудови робочих місць, об’єктів обробки та логічних відносин між робочими місцями. Інші об’єкти моделювання, такі як ресурси, є необов’язковими, якщо їх наявність не вимагається умовами побудови моделі. Логічні відносини визначають алгоритми обробки об’єктів, а також послідовність обробки об’єктів на робочих місцях. Вони повинні реально відображати існуючі виробничі процеси, або виробничі процеси, що плануються.

Об’єкти обробки (entities). Об’єкти обробки – одні з найважливіших елементів моделі СМО, що розробляється за допомогою системи *Service Model*. Об’єкти обробки рухаються по

моделі і підлягають обробці під час моделювання. Вони моделюють об'єкти, що обслуговуються у реальній СМО. Наприклад якщо моделюється цех, у якому виробляють деталі, то ці деталі і будуть об'єктами обробки. Об'єкти обробки рухаються в моделі по певним шляхам переміщення і по закінченні циклу обробки виходять з моделі. Вхід об'єктів обробки (наприклад деталей) в модель моделюється появою об'єкта на першому місці обробки (наприклад станку), де він повинен оброблятися, або на спеціальному місці обробки, що моделює вхід в модель. Далі об'єкт обробки рухається в моделі по шляхам переміщення між наступними місцями обробки (станками чи приладами), а по закінченні процесу обробки виходить з моделі і більше не з'являється в ній. Процес обробки може бути індивідуальним для кожного виду об'єкта обробки. Об'єкти обробки, які обробляються на робочих місцях, є елементами вхідних потоків.

Параметри вхідних потоків(arrivals). Вхідним потоком для моделі СМО є множина об'єктів обробки, які з'являються в моделі для подальшої обробки протягом всього процесу моделювання. Правила появи об'єктів обробки в моделі визначаються законами розподілення параметрів потоків. Параметри вхідних потоків визначаються в залежності від особливостей модельованої СМО і відображають реальні закони чи принципи входу об'єктів обробки в модель. Кожний з основних параметрів вхідних потоків може визначатися формулою з використанням дистрибутивних законів розподілення. Окрім того, опис вхідного потоку може бути задано у вигляді таблиці, що відображає реальну статистику. Таблиці, що описують вхідні потоки можуть бути підготовлені за допомогою зовнішніх програм (наприклад, у *MS Excel*).

Робочі місця (locations). Робоче місце визначається функцією чи операцією конкретного виробничого або бізнес-процесу. Кожне робоче місце має виконувати певну функцію по обробці об'єктів. Причому, під обробкою мається на увазі не лише безпосередня обробка об'єкта, але також його обслуговування. Конкретний сенс в кожне робоче місце вкладається при моделюванні СМО з певними характеристиками. Якщо використовувати приклад цеху про обробці деталей, то можна сказати, що під робочим місцем розуміється кожний станок, на якому обробляються деталі. У банку, офісі, поліклініці робочим місцем буде робоче місце директора, менеджера, секретаря, лікаря і т. ін. Взагалі, робоче місце – це місце, де над об'єктами обробки виконуються будь-які дії.

Ресурси (resources). Ресурси – це все, що допомагає забезпечувати виконання операцій над об'єктами обробки. У реальній СМО роль ресурсів можуть виконувати обслуговуючий персонал (продавці, секретарі, менеджери, лікарі, медсестри, робочі, стюардеси), обчислювальна техніка, станки, оргтехніка, транспорт і т. ін.

Шляхи переміщення (path networks). Шляхи переміщення відображають переміщення ресурсів та об'єктів обробки під час виконання пов'язаних з ними операцій. Переміщення людських ресурсів може бути пов'язане з виконанням безпосередніх функцій: приходом на робоче місце від місць відпочинку, прийняття їжі, перекурів, і т.ін., а також відходів від робочого місця до місць відпочинку або іншої локалізації. Кожен такий шлях характеризується часом, який, в загальному випадку, є втраченим. Шляхи задаються графічно серед розташованих раніше робочих місць.

Процеси на робочих місцях (processing). Процеси на робочих місцях відображають алгоритми обробки об'єктів і є найголовнішими при моделюванні будь-якої моделі СМО. Процеси відображають реальні відносини між об'єктами моделі СМО. За їх допомогою моделюється процес обробки об'єкта моделі. Тут вказуються особливості обробки об'єктів на кожному робочому місці. Обов'язковою умовою правильного моделювання є те, що всі об'єкти, що оброблялися на конкретному робочому місці повинні вийти з нього.

5.1.5 Опис досліджуваної системи масового обслуговування

В одному з цехів підприємства є наступні види станків і обладнання: токарний станок; центрово-шліфувальний станок; свердлильний станок; горизонтально-протяжний станок; зубостругальний станок; вібробункер; мийна машина; гальванічна ванна. В цьому цеху виготовляються три види зубчатих коліс. Кожне колесо має свою технологію виготовлення. Технологія задає скільки станків необхідно для обробки деталі, а також час обробки на кожному станку. Ці дані наведені в табл. 5.1. Під час транспортування між станками витрачається деякий час, він вважається невиробничим. Для деталей першого та другого типу цей час однаковий. Для перевезень між усіма станками він становить 0.4 хв. Для деталі третього типу цей час становить: від токарного станка до свердлильного 0.5 хв., а від свердлильного до зубостругального 1.2 хв.

Частота появи деталей в цеху задана в табл. 5.2. Моделювання здійснюється протягом однієї зміни, тобто 8-годинного робочого дня.

Таблиця 5.1 – Час обробки деталей на станках, хв.

Вид обладнання	Вид деталі		
	Зубчате колесо 1 тип	Зубчате колесо 2 тип	Зубчате колесо 3 тип
Токарний станок	0.50	0.66	0.69
Центрово-шліфувальний станок	0.15	0.15	–
Свердлильний станок	0.28	0.3	0.47
Горизонтально-протяжний станок	0.10	0.10	–
Зубостругальний станок	1.2	2.1	0.67
Вібробункер	0.07	0.07	–
Мийна машина	0.05	0.05	–
Гальванічна ванна	0.10	0.10	–

Таблиця 5.2 – Частота появи деталей в цеху

Вид деталі	Інтервал появи деталі на вході, хв.
Зубчате колесо 1 тип	0.40
Зубчате колесо 2 тип	0.33
Зубчате колесо 3 тип	0.38

5.1.6 Послідовність моделювання в системі Service Model

Вхідними даними для програми *Service Model* є файл, у якому міститься створена за допомогою функцій та можливостей *Service Model* модель СМО. В цій моделі повинні бути правильно описані робочі місця, об'єкти обробки, вхідні потоки, шляхи переміщення і, якщо треба, ресурси, регламенти та сценарії. Для роботи моделі

необхідний також правильний опис взаємодій між об'єктами обробки та робочими місцями; між об'єктами обробки та ресурсами, тобто між усіма компонентами моделі. Таким чином, вхідними даними для системи імітаційного моделювання *Service Model* є модель реальної СМО правильно відображена та коректно описана з точки зору системи *Service Model*.

Запуск системи *Service Model* виконується при виборі файлу Smstu.exe. На екрані при цьому з'являється головна панель *Service Model*. Для вибору вже існуючої моделі треба навести вказівник миші на кнопку “*Open a model*” та натиснути ліву кнопку миші. З'явиться вікно “*Load model*”, у якому пропонуються назви моделей, що вже існують. Для вибору моделі треба лише виділити її назву та натиснути кнопку “*Открыть*”. Для створення нової моделі треба в меню вибрати пункт *File/New*. При спробі створення нової моделі на екран буде виведений запит основних даних про модель (*General Information*). У цьому вікні задається наступне:

- назва моделі(*Title*) (але не ім'я файлу),
- одиниці виміру часу і відстані(*Time units* та *Distanse units*),
- колір фону моделі(*Background color*),
- колір і крок масштабної сітки(*Grid/Scale*),
- файл із графічними образами(*Graphic library file*),
- початкова логіка і логіка завершення (*Initialisation logic* та *Termination logic*),
- примітки(*Model notes*).

Розглянемо процес створення моделі на прикладі цеху по обробці деталей. При створенні моделі цеху у вікні *General Information* модель названа “Модель цеху”. Одиницями виміру часу і відстані даної моделі є відповідно хвилини та метри. Це вказується встановленням відповідних позначок в лівому боці вікна *General Information*. Шлях до файлу, що містить у собі графічну бібліотеку моделі, вказаний в спеціальному полі (*Graphic library file*). Встановлюється також колір фону моделі (*Background color*). Інші параметри моделі залишилися по замовчанню. Після натискання <OK> Ви отримуєте змогу редагувати(або створювати) модель. Для подальшого створення моделі увійдіть в пункт меню *Build|Locations*. При цьому на екрані з'являється картинка. На ній можна побачити три вікна, необхідні для створення робочих місць: *Locations*, *Layout* та *Graphics*. Вікно *Layout* – це безпосередньо вікно моделі, в якому і

буде створюватись модель. У вікні **Locations** міститься опис характеристик та параметрів робочого місця. Вікно **Graphics** містить графічні образи робочих місць, приклади яких зображені на рис. 5.2.

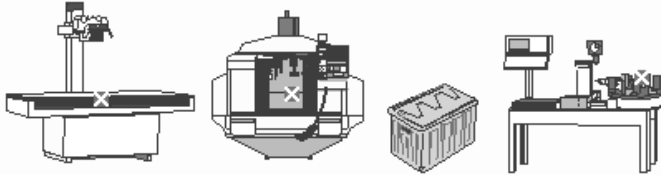


Рисунок 5.2 – Приклади робочих місць

Для визначення робочих місць (у нашому випадку – всі станки, а також вхід та вихід) у вікні **Graphics** вибираємо графічне представлення об'єкта і клікаємо на ньому лівою кнопкою миші, а потім клікаємо тією ж кнопкою у вікні **Layout** - там з'являється графічне зображення потрібного робочого місця. У той же час у вікні **Locations** з'являється рядок з описом щойно створеного робочого місця, у якому присутні наступні параметри:

- назва робочого місця (**Name**);
- зображення (**Icon**);
- місткість, що визначає кількість об'єктів, які можуть одночасно знаходитись у даному робочому місці (**Cap**);
- кількість одиниць даного робочого місця (**Units**);
- правила обробки об'єктів на даних робочих місцях, що складаються з:
 - правил вибору об'єктів для поточної обробки з потоку об'єктів (**Rules**);
 - правил формування вихідного потоку (**Dts**);
 - правил розподілення обробки вхідних об'єктів по робочим місцям, якщо місць даного типу більше одного (**Dts**);
 - текстового опису робочого місця (**Notes**).

Для робочих місць, що моделюють станки залишимо всі значення за замовчуванням. А от для входу та виходу змінимо ємність (**Cap**) на нескінченну, замінивши одиничку зарезервованим словом **inf**. Для створення об'єктів обробки потрібно вибрати пункт меню **Build\Entities**. Вигляд екрану при цьому зміниться. Тепер на екрані також три вікна: вже знайоме нам вікно **Layout**, вікно, яке містить

графічні зображення (рис. 5.3) для об'єктів обробки *Entity graphics* та вікно, у якому міститься опис параметрів об'єкта обробки *Entity*.



Рисунок 5.3 – Приклади об'єктів обробки

Визначаємо об'єкти обробки (у нашому випадку - деталі). У вікні *Entity Graphics* вибираємо зображення таким же чином, як і для робочого місця, але на відміну від нього це зображення не потрібно розмішувати в полі моделі. Це не робиться тому, що об'єкти обробки рухаються і в моделі може знаходитися одночасно декілька об'єктів того самого типу. При виборі зображення у вікні *Entities* автоматично з'являється рядок з описом типу даних об'єктів обробки. Для моделі цеху потрібно об'явити три види деталей, тому у вікні *Entities* будуть знаходитися три рядки опису видів деталей. Кожний рядок відображає не опис конкретної деталі, а опис виду деталей, кількість та характеристики яких будуть визначені при подальшому створенні моделі. Рядок опису типу об'єктів містить наступні поля:

- графічне представлення (*Icon*);
- ім'я (*Name*);
- швидкість (*Speed*, у дужках приведені одиниці її виміру);
- метод збору статистики по типу об'єктів (*Stats*);
- примітки (*Notes*).

У випадку моделювання цеху по обробці деталей, залишимо всі параметри об'єктів обробки (деталей) незмінними. Після об'явлення типів об'єктів обробки необхідно визначити для кожного типу об'єкта обробки вхідні потоки. Це можна зробити за допомогою виклику пункту меню *Build\Arrivals*. На екрані при цьому будуть розміщені наступні вікна: *Layout*, *Tools*, у якому розміщуються вже існуючі в системі (тобто об'явлені) об'єкти обробки та *Arrivals* – вікно, що містить опис вхідних потоків. Визначаємо вхідні потоки об'єктів обробки. Для цього у вікні *Arrivals* представлена таблиця з полями, в яких задаються характеристики потоків. Ці потоки в системі імітаційного моделювання *Service Model* мають наступні

характеристики:

- ім'я об'єкта обробки (**Entity**);
- початкове місце входу об'єкта в модель (**Location**), тобто назва місця обробки, де вперше з'явиться об'єкт обробки в процесі моделювання;
- кількість екземплярів об'єктів, що з'являються в моделі за одну порцію (**Qty each**);
- момент надходження першої порції (**First Time**);
- кількість таких порцій, що потраплять до моделі за весь час моделювання (**Occurrences**);
- час між появами означених порцій на місці обробки (**Frequency**);
- логіка появи об'єктів в моделі (**Logic**);
- опція включення/відключення потоку (**Disable**).

Задамо достатній набір параметрів для вхідного потоку об'єктів (деталей) до пункту входу в модель (місце обробки input). Для цього у вікні **Arrivals** клікаємо правою кнопкою миші на заголовку таблиці в полі **Entity**, після чого випадає віконце з переліком назв типів об'єктів обробки. Подвійним кліканням на імені потрібного типу вибирається тип об'єктів обробки, для якого у цьому рядку буде визначений вхідний потік. В полі **Location** таким же чином вибирається місце, на якому вперше з'явиться об'єкт цього типу. У полях **Qty each**, **First time**, **Occurrences**, **Frequency** безпосередньо задаються основні параметри вхідного потоку. Ці параметри однозначно задають вхідний потік деталей. Основні параметри потоків, які істотно впливають на процес моделювання є:

- кількість екземплярів об'єктів обробки в порції (n);
- кількість порцій (q);
- час появи першої порції (t_0);
- час між порціями (τ).

Кожний з основних параметрів вхідних потоків може визначатися формулою з використанням дистрибутивних законів розподілення.

Таким чином, якщо розглядати вхідний потік для деталей першого типу, то починаючи з моменту часу 0 (**First Time**), до цеху, а точніше на вхід моделі-input (**Location**) будуть поступати деталі (об'єкти обробки) через кожні 0.33 хв. (**Frequency**). Загальна кількість деталей не обмежена, тому що у полі **Occurrences** стоїть inf –

зарезервоване слово, що означає безкінечність.

Тепер необхідно визначити шляхи переміщення об'єктів обробки по моделі. Це можна зробити, вибравши пункт меню **Build|Path Networks**. На екрані у цьому випадку відображаються наступні вікна: **Layout, Path Networks**, у якому визначаються шляхи переміщення об'єктів обробки по моделі та **Paths**, у якому описуються відрізки, з яких складаються шляхи переміщення. У вікні **Path Networks** задаються наступні характеристики шляхів переміщення:

- колір графічного зображення, а також чи виводити зображення шляху переміщення під час запуску та роботи моделі (**Graphic**);
- назва шляху переміщення (**Name**);
- тип шляху переміщення (**Type**);
- визначення одиниць вимірювання переміщень об'єктів обробки під час моделювання на цьому шляху (час чи швидкість&дистанція) (**T/S**);
- кількість відрізків та визначення кожного з них (**Paths**);
- кількість вузлів, прив'язаних до місць обробки (**Interfaces**);
- кількість відрізків шляху, що моделюють реальну відстань (**Mapping**);
- загальна кількість вузлів даного шляху (**Nodes**).

Для створення шляху переміщення у вікні **Path Networks** подвійним кліканням лівої кнопки миші добиваємося появи нового рядка – це новий шлях переміщення. Тепер для нього потрібно визначити відрізки, з яких він складається, та вузли, в яких ці відрізки починаються та закінчуються. Відрізки визначаються за допомогою миші. У вікні **Layout** натисканням на ліву кнопку визначається початок відрізка, а на праву – кінець цього відрізка. Вузлами при цьому вважаються крапка початку та кінцю відрізка. Вони створюються автоматично. Також автоматично системою **Service Model** вираховується час, який буде втрачатися при переміщенні об'єкта обробки по цьому шляху (**Time**). Після визначення самого шляху необхідно прив'язати його вузли до місць обробки деталей. Це робиться у вікні **Path Networks** кліканням лівою кнопкою миші на заголовок **Interfaces** таблиці, що розміщена в цьому вікні. При цьому віконце **Paths** закривається, а на його місці з'являється інше вікно – **Interfaces**. В цьому вікні при одному кліканні лівою кнопкою миші на заголовок **Node** в таблиці видається віконце з переліком всіх вузлів даної сітки. В ньому можна вибрати вузол за допомогою подвійного

клікання лівою кнопкою миші. Таким же чином вибирається і робоче місце, до якого буде прив'язаний цей вузол, але це робиться в іншому стовпчику таблиці (**Location**). Після того, як буде вибраний вузол (**Node**) та робоче місце (**Location**), вважається, що цей вузол прив'язаний до робочого місця і об'єкти обробки, що будуть рухатися по шляху переміщення від вузла до вузла насправді будуть переміщуватися від одного робочого місця до іншого. Ті параметри, які не були розглянуті можна залишити без змін. Вони не є головними і їх значення по замовчанню нас влаштовують.

Найголовніше в процесі створення імітаційної моделі – задати всі дії, які будуть виконуватися над об'єктами обробки в процесі моделювання. Це можна зробити, вибравши пункт меню **Build|Processing**. На екрані будуть розміщені три віконця: **Tools**, в якому розміщені види об'єктів обробки та деякі кнопки, **Process**, де і знаходиться опис процесів на кожному робочому місці та вікно **Routing**, у якому розміщений опис подальшого направлення руху об'єктів з даного робочого місця.

Розглянемо процес визначення дій над об'єктами обробки у моделі. У нашому випадку ми можемо представити процес обробки однієї деталі на одному станку однією дією – безпосередньою обробкою деталі (затримкою). Усі переміщення по моделі прив'язуються до безпосередніх дій на місцях обробки. Для визначення будь-якої дії над будь-яким об'єктом обробки спочатку необхідно вибрати об'єкт обробки та місце, де він буде оброблятися. Це можна зробити у першому рядку вікна **Process**. Вікно **Process** містить інформацію, що безпосередньо описує процес обробки на даному робочому місці:

- тип об'єктів, що підлягають обробці на даному робочому місці (**Entity**);
- назва робочого місця, де відбувається даний процес (**Location**);
- опис алгоритму обробки з використанням елементів програмування (**IF-THEN, DO WHILE, GO_TO**) та інших вбудованих функцій або стверджень, з визначенням часу продовження процесу та переліку робочих місць для наступної обробки об'єктів (**Operation**).

Для прикладу, створимо процес обробки деталей першого типу (Zubch1) на токарному станку (Tokarn). У таблиці вікна **Process** це другий рядок. У цій таблиці клікнемо лівою кнопкою миші на заголовок поля **Entity** і у віконці **Entitis**, яке при цьому з'явиться,

подвійним кліканням лівої кнопки виберемо ім'я типу об'єкта обробки (Zubch1), для якого потрібно визначити процес обробки. Таким же чином у полі **Location** вибираємо ім'я робочого місця (Tokarn). У полі **Operation** описуємо, що буде відбуватися з деталлю на цьому станку. З підрозділу 1.5 візьмемо час обробки деталей першого типу на токарному станку. Цей час дорівнює 0.5 хв. Тому, затримка першого виду деталі на токарному станку дорівнює 0.5 одиниць часу (одиницями часу були вибрані хвилини). Використаємо для моделювання процесу обробки (затримки) стандартну процедуру затримки – Wait 0.5. Перейдемо у вікно **Routing**. Вікно **Routing** містить опис кожного наступного шляху передачі об'єктів з вказівкою правил та способів передачі об'єктів по кожному з них:

- вихідний об'єкт (**Output**);
- наступне місце обробки об'єкта (**Destination**);
- правила передачі об'єкта (**Rule**);
- логіка переміщення на наступне місце обробки (**Move logic**).

У ньому визначаємо, куди направиться об'єкт після обробки на токарному станку. Згідно з підрозділом 1.5, деталі першого типу після обробки на токарному станку переходять на подальшу обробку центрово-шліфувальним станком. Тому, у вікні **Routing** клікаємо лівою кнопкою миші на заголовок **Output** таблиці, що у ньому розміщена. У віконці, що з'явиться подвійним кліканням виберемо ім'я об'єкту обробки (Zubch1), що буде виходити з місця обробки (токарного станка Tokarn). У наступному стовпчику таблиці (**Destination**) виберемо місце наступної обробки деталі – центрово-шліфувальний станок (SentrShlif). Необхідно мати на увазі, що всі об'єкти обробки, які оброблялися на робочому місці повинні з нього вийти. В іншому випадку модель працювати не буде. Для коректного переміщення деталі по моделі необхідно задати також шлях переміщення. Це робиться у стовпчику **Move logic** цього ж вікна за допомогою зарезервованої команди “move on Net1”, де Net1 – назва шляху переміщення між вузлами яким об'єкт обробки буде переміщуватись між токарним та центрово-шліфувальним станками. Тепер деталь Zubch1 вже буде знаходитись на центрово-шліфувальному станку і далі при створенні моделі потрібно буде описувати її обробку на цьому станку (наступний рядок у таблиці вікна **Process**). Таким чином, кожному рядку у вікні **Process** буде відповідати своє вікно **Routing**, що містить подальший напрям руху

об'єкту. Таким чином, описуються всі процеси на робочих місцях. Після того, як екземпляр об'єкта обробки (у нашій моделі деталі) пройшов обробку на всіх станках він повинен вийти з моделі. Тому, останнім місцем його призначення повинен бути вихід з моделі. Для цього в полі **Destination** вікна **Routing**, яке відповідає процесу обробки деталі на виході (output), ставимо EXIT. Всі інші поля цього вікна в цьому випадку ролі не грають.

Для того, щоб під час моделювання зручно було слідкувати за кількістю вже оброблених деталей, можна створити змінні. Це робиться після вибору пункту меню **Build|More elements|Variables (Global)**. На екрані будуть знаходитись вікна **Layout** та **Variables (Global)**. Саме у вікні **Variables (Global)** і визначаються змінні. Таблиця цього вікна містить наступні стовпчики:

- значок “yes” або “no”, який визначає присутність значення цієї змінної у полі моделі (**Icon**);
- назва змінної (**ID**);
- її тип (**Type**);
- початкове значення (**Initial value**);
- тип збору статистики (**Stats**);
- примітки (**Notes**).

Для створення змінної лівою кнопкою миші необхідно двічі клікнути на вільному рядку таблиці, розміщеної у вікні **Variables (Global)**. Після появи нового опису змінної для відображення її значення у вікні **Layout** треба на потрібному місці клікнути лівою кнопкою миші, після чого з'явиться віконце, в яке під час моделювання буде виводитись значення змінної. Тепер стає можливим змінювати параметри змінної: ввести назву (**ID**), якщо надана системою не влаштовує, початкове значення (**Initial value**) та тип змінної (**Type**). Інші параметри можна залишити без змін. Для правильного відображення змінною існуючої ситуації в системі, її значення треба змінювати. Це можна зробити, знову відкривши пункт меню **Build|Processing**. Та у вікні **Processing** в той момент, коли об'єкт обробки виходить з моделі, за допомогою функції INC збільшити змінні на одиничку. Наприклад INC Detall, де Detall – назва змінної, що рахує вже оброблені деталі першого типу.

Після закінчення побудови моделі необхідно встановити час, який система буде моделювати процес обробки деталей. Це можна зробити, вибравши пункт меню **Simulation|Options**. На екрані

з'явиться вікно *Simulation Options*. У цьому вікні у полі *Run Hours* треба ввести “8” – тобто 8-годинний робочий день. Збережіть модель *File|Save* і запустіть її на виконання *Simulation|Run*. Виконання завершиться, коли буде вичерпаний час, за який проводиться моделювання, тобто коли системний час стане рівним встановленому – 8 годин. По закінченні роботи програма запитає, чи варто збирати статистику, і чи хочете ви продивитися результати роботи. При підтвердженні буде генерований звіт і запущений модуль перегляду звітів (*Service Model Output*).

5.1.7 Вихідні дані програми

Вихідними даними програми *Service Model* є різноманітна інформація про процес моделювання, а також статистика по кожному виду об'єктів, які присутні в моделі.

Переглянути вихідні результати моделювання дозволяється у двох випадках: після закінчення роботи моделі, або за допомогою модуля “Переглянути статистику” (*View statistics*) в меню *Output*.

Після закінчення моделювання результати моделювання можна продивитися в генеральному текстовому звіті (*General report*). Цей звіт складається з усіх даних, безпосередньо використовуваних при моделюванні. Однак, представлення інформації про вихідні результати в текстовій формі не є наочним.

Service Model надає декілька графічних звітів для зручності користувача. Цікавою можливістю *Service Model* є наявність опції створення вихідних файлів, які можна відкривати під час порівняння різних моделей. Для створення таких файлів необхідно вибрати модуль “*Open Multiple*” в меню *File* та додати поточний вихідний файл.

Розглянемо вихідні дані про модель, що містяться в генеральному текстовому звіті (*General report*). Звіт містить наступну інформацію:

Output – вказується шлях до файлу, в якому міститься модель СМО (назва моделі наведена в квадратних дужках);

Date – дата проведення моделювання;

Time – час проведення моделювання;

LOCATIONS – загальні дані про робочі місця, які складаються з:

Location Name – назва робочого місця;

Sheduled Hours – час роботи робочого місця;

Capacity – місткість робочого місця;

Total Entries – загальна кількість входів в робоче місце;

Average Minutes Per Entry – середній час входження;

Average Contents – середня завантаженість;

Maximum Contents – максимальна завантаженість;

Current Contents – поточна завантаженість;

% Util – процент використання.

LOCATIONS STATES BY PERCENTAGE (Multiple Capacity)–

стан робочих місць у відсотках в залежності від часу роботи (багатомісцеві робочі місця):

Location Name – назва робочого місця;

Sheduled Hours – час роботи робочого місця;

% Empty – робоче місце вільне;

% Partially Occupied – робоче місце частково зайняте;

% Full – робоче місце заповнене;

% Down – робоче місце не задіяне в роботі.

LOCATIONS STATES BY PERCENTAGE (Single Capacity) –

стан робочих місць у відсотках в залежності від часу роботи (одномісцеві робочі місця):

Location Name– назва робочого місця;

Sheduled Hours – час роботи робочого місця;

% Operation – робоче місце в операції;

% Setup – встановлення;

% Idle – час простою робочого місця;

% Waiting – робоче місце в стані очікування;

% Blocked – робоче місце заблоковане;

% Down – робоче місце не використовується.

FAILED ARRIVALS – невдалі спроби зайняти робоче місце:

Entity Name – назва об'єкту, що намагався зайняти робоче місце;

Location Name – назва робочого місця, яке він намагався зайняти;

Total Failed – загальна кількість невдалих спроб;

ENTITY ACTIVITY – дані про діяльність об'єктів обробки:

Entity Name – ім'я об'єкта обробки;

Total Exits – загальна кількість виходів з моделі;

Current Quantity In System – поточна кількість об’єктів обробки в моделі;

Average Minutes In System – середній час знаходження в системі;

Average Minutes In Move Logic – середній час, що витрачає об’єкт на рух між робочими місцями;

Average Minutes Wait For Res, etc. – середній час обслуговування об’єктів;

Average Minutes In Operation – середній час операції;

Average Minutes Blocked – середній час простою.

ENTITY STATES BY PERCENTAGE – стан об’єктів обробки у відсотках від всього часу, який об’єкт проводить в моделі:

Entity Name – ім’я об’єкта обробки;

% In Move Logic – витрачено на переміщення;

% Wait For Res, etc. – відсоток часу, який витрачається на обробку;

% In Operation – відсоток часу, який витрачається на очікування;

% Blocked – відсоток часу, який витрачається на простій.

VARIABLES – змінні:

Variables Name – ім’я змінної;

Total Changes – загальна кількість змін стану;

Average Minutes Per Change – середній час між змінами стану;

Minimum Value – мінімальне значення;

Maximum Value – максимальне значення;

Current Value – поточне значення;

Average Value – середнє значення.

5.2 Завдання до виконання роботи

5.2.1 Вивчити методику побудови та імітаційного моделювання СМО на прикладі тестової моделі “Модель цеха” за допомогою системи Service Model.

5.2.2 Виконати моделювання тестової моделі СМО “Модель цеха”, одержати генеральний звіт.

5.2.3 Скопіювати модель в свій робочий каталог під іншим іменем.

5.2.4 Виконати перетворення моделі згідно з даними для свого варіанту (див. додаток М). Виконати зміну наступних параметрів:

- кількість годин роботи моделі (***Simulation/Options Run Hours***);
- час обробки деталей на станках (***Build/Processing*** вікно Process (Operation));

- кількість деталей, що будуть оброблюватися (***Build/Arrivals Occurences***);

- частота надходження деталей в модель (***Build/Arrivals Frequency***).

5.2.5 Одержати результати для всіх випадків.

5.2.6 Дослідити та порівняти результати моделювання тестової та побудованих моделей, зробити висновки.

5.3 Зміст звіту

5.3.1 Мета роботи.

5.3.2 Теоретичні відомості у конспективній формі.

5.3.3 Вхідні дані для побудови тестової моделі СМО.

5.3.4 Генеральний звіт про результати моделювання для тестової моделі.

5.3.5 Звіти про результати моделювання для досліджуваних перетворених моделей.

5.3.6 Висновки по роботі з порівнянням результатів моделювання.

5.4 Контрольні питання

5.4.1 Призначення та різновиди систем масового обслуговування.

5.4.2 Аналітичні та імітаційні моделі СМО, умови їх застосування.

5.4.3 У яких станах можуть знаходитись обслуговуючі апарати (канали) та замовлення (транзакти)?

5.4.4 Види дисциплін обслуговування.

5.4.5 Сутність методу моделювання подій.

5.4.6 Види статистичних відомостей, що надає система імітаційного моделювання.

5.4.7 Побудова мережевої імітаційної моделі.

5.4.8 Алгоритм імітаційного моделювання.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Томашевський, В.М. Моделювання систем [Текст] / В.М. Томашевський. – К.: ВНУ, 2005. – 352 с.
2. Норенков И.П. Основы автоматизированного проектирования. Учебник для студ. ВУЗов.- М.: МГТУ им. Баумана, 2002.-336 с.
3. Колодницький М.М. Елементи теорії САПР складних систем. Навчальний посібник. – Житомир: ЖІТІ, 1999.- 512 с.
4. Болдин А.Н., Задиранов А.Н. Основы автоматизированного проектирования. Учебное пособие для студ. ВУЗов.- М.: МГИУ, 2006.-104 с.
5. Тимченко А.А. Основы системного проектування та системного аналізу складних об'єктів: Основы САПР та системного проектування складних об'єктів: Підручник/ За ред.В.І. Бикова.- К.: Либідь, 2003.- 272с.
6. Практическое моделирование динамических систем [Текст] / Е.С. Бенькович, Ю.Б. Колесов, Ю.Б. Сениченков. – СПб: БХВ-Петербург, 2002. – 464 с.
7. Колесов, Ю.Б. Моделирование систем. Динамические и гибридные системы [Текст] / Ю.Б. Колесов, Ю.Б. Сениченков. – СПб: БХВ-Петербург, 2006. – 224 с.
8. Кельтон, В. Имитационное моделирование [Текст] / В. Кельтон, А. Лоу. – СПб.: Питер, К.: ВНУ, 2004. – 847 с.
9. Вирішення практичних задач методами комп'ютерного моделювання [Текст] / В.М. Томашевський, О.Г. Данова, О.О. Жолдаков. – К.: Корнійчук, 2001. – 267 с.
10. Рыжиков, Ю. Имитационное моделирование. Теория и технологии [Текст] / Ю. Рыжиков. – М.: Альтекс-А, Корона Принт, 2004. – 384 с.
11. Полупроводниковые приборы: транзисторы. Справочник / В. А. Ароков, А. В. Баюков, А. А. Зайцев и др. Под общ. редакцией Н. Н. Горбунова. – М.: Энергоиздат, 1983. – 904 с.

Додаток А

Завдання до лабораторної роботи №1

Таблиця А.1 – Вихідні дані для розрахунку

№	Тип двовимірного рівняння	Кількість комірок по осі		Розмір комірки, мм		$C(x,y)$, Дж/кг·К	$P(x,y)$, кг/м ³	λ , Вт/м·К
		x	y	h_x	h_y			
1	Фур'є	2	3	1	1	730	2620	1,00
2	Лапласа	2	3	1	1	700	2840	1,50
3	Фур'є	3	3	2	1	860	2800	1,31
4	Лапласа	3	3	2	1	840	2702	0,98
5	Фур'є	3	2	1	2	800	2750	1,05
6	Лапласа	3	2	1	2	880	2100	30,00
7	Фур'є	3	4	2	2	870	2200	210,00
8	Лапласа	3	4	2	2	600	2000	15,00
9	Фур'є	4	3	2	3	865	2100	16,00
10	Лапласа	4	3	2	3	730	2620	1,00
11	Фур'є	4	4	3	2	860	2800	1,31
12	Лапласа	4	4	3	2	870	2200	210,00
13	Фур'є	3	2	2	1	860	2520	30,00
14	Лапласа	3	2	3	2	740	2300	70,00
15	Фур'є	4	4	3	2	720	2500	1,23
16	Лапласа	2	3	2	2	600	2640	180,00
17	Фур'є	2	3	3	2	650	2630	60,00
18	Лапласа	2	3	2	3	730	2820	18,00
19	Фур'є	3	2	2	2	710	2150	220,00
20	Лапласа	3	4	2	1	840	2330	0,96
21	Фур'є	4	2	3	1	820	2690	0,99
22	Лапласа	4	2	1	1	880	2400	1,60
23	Фур'є	3	3	1	2	730	2460	70,00
24	Лапласа	3	3	1	3	780	2210	65,00
25	Фур'є	2	3	2	4	810	2750	29,00

Додаток Б

Методика побудови та аналізу електричних принципових схем за допомогою системи CircuitMaker 6 Pro

Б.1 Призначення системи Circuit Maker

Впровадження в інженерну практику методів автоматизації проектування дозволяє перейти від традиційного макетування проекрованої апаратури до її моделювання за допомогою персональних комп'ютерів. Тому, потреба в засобах схемотехнічного аналізу дуже велика, що й призвело до розвитку ринку схемотехнічних програм (наприклад, PSPICE, Micro-CAP, МАЕС-П).

Одним з таких пакетів є Circuit Maker.

Система Circuit Maker дозволяє виконати такі проектні операції:

- створення умовних графічних позначень елементів електричної принципової схеми;
- графічне введення зображення і конструктивів проектованого пристрою;
- аналогове та цифрове моделювання електронних пристроїв;
- визначення напруги, сили струму, потужності елементів принципової схеми;
- визначення залежностей напруги, сили струму і миттєвих потужностей елементів електричної принципової схеми від часу, частоти та інших параметрів.

Б.2 Підготовка до роботи

2.1 Для виконання завдання необхідно завантажити в оперативну пам'ять програму Circuit Maker (запустити файл *C:\Program Files\MicroCode Engineering\Cirmaker.exe* або з головного меню Windows – *Programs\CircuitMaker 6 PRO\CircuitMaker*). На екрані монітора з'явиться вікно Circuit Maker PRO, яке містить вікно робочого документу, основне меню, панель інструментів, вертикальну та горизонтальну лінії прокрутки.

2.2 Створити новий файл за допомогою команди File/New.

Б.3 Побудова електричної принципової схеми

Б.3.1 Введення елементів схеми

Виберіть лівою кнопкою "миші" команду *Devices\Hotkeys2*. На екрані з'явиться список елементів. Виберіть елемент *Resistor*. Курсор прийме вигляд обраного елемента. Встановіть його на необхідну позицію на екрані та натисніть ліву кнопку "миші". Курсор прийме звичайний вигляд.

Обрати елемент також можна за допомогою кнопки на панелі інструментів: 

На екрані з'явиться вікно *Device Selection*.

Наприклад, необхідно вибрати резистор. З вікна *Major Device Class* оберіть *Passive Components*, з *Minor Device Class* – *Resistors*, а з *Device Symbol* – *Resistor:A*. Після цього натисніть кнопку *Place*.

Ви можете змінити положення елемента:

– встановити в інше місце (для цього потрібно зробити його активним – натиснути лівою кнопкою "миші" на самому елементі, який виділиться червоним кольором, та не відпускаючи кнопки перетягнути в необхідне місце);

– перевернути елемент – необхідно зробити його активним та натиснути на панелі інструментів кнопку: 

Аналогічним чином введіть інші елементи схеми (конденсатор, транзистор, заземлення, джерело живлення та інші) із *Devices\Hotkeys1* або *Devices\Hotkeys2*, або натиснувши на панелі інструментів кнопку з зображенням транзистора та конденсатора.

Б.3.2 Введення значень параметрів елементів схеми

Для введення нового значення параметра елемента необхідно подвійно натиснути ліву кнопку "миші". На екрані з'явиться діалогове вікно *Edit Device Data* з характеристиками елемента. В поле *Label-Value* введіть необхідне значення параметру (наприклад, якщо $R = 50\text{ Ом}$, введіть значення 50; якщо $R = 50\text{ кОм}$, введіть 50к).

Аналогічно задаються значення параметрів інших елементів (в тому числі значення напруг у граничних вузлах).

Б.3.3 З'єднання елементів

Вибрати на панелі інструментів кнопку у вигляді "+". Курсор прийме вигляд хрестика.

Елементи з'єднати таким чином: підвести курсор до виводу першого елемента (на перехрещенні їх з'явиться червоний прямокутник), натиснути ліву кнопку "миші" та, не відпускаючи її, підвести курсор до виводу другого елемента і знову натиснути ліву кнопку.

Виберіть команду *Options*, уберіть галочку біля *Show Pin Dots* (натисніть на неї лівою кнопкою "миші").

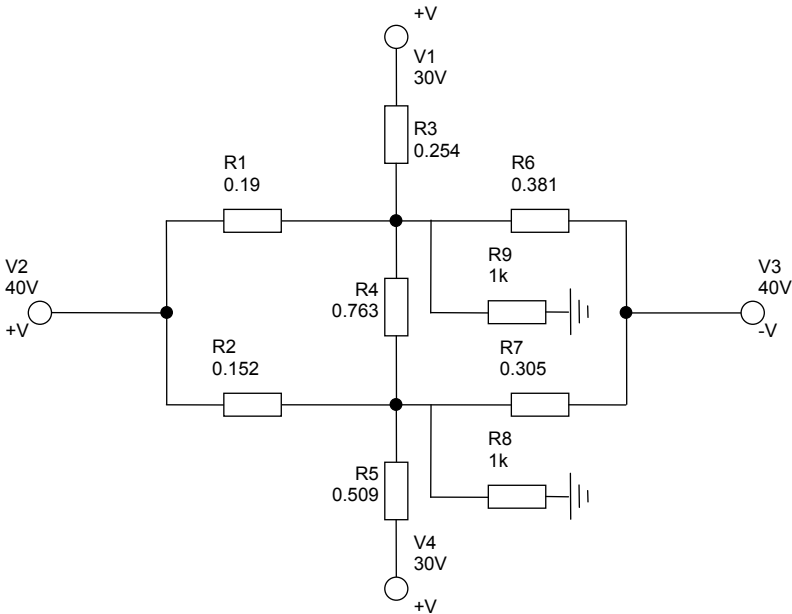


Рисунок Б.1 – Прилад побудови схеми

Б.3.4 Запис схеми на диск

Виберіть команду *File\Save*. Введіть назву файла.

Приклад електричної принципової схеми наведено на рис Б.1

Б.4 Аналіз електричної принципової схеми

Б.4.1 Вибір типу аналізу

Натисніть команду *Simulation* в головному меню системи Circuit Maker. З'явиться підменю. Зверніть увагу на те, щоб біля

Analog Mode стояла галочка. Якщо цієї команди немає в списку підменю, то замість неї ви побачите *Digital Mode*. Виберіть зі списку строку *Digital Mode* та натисніть на неї. Тепер у списку з'явиться *Analog Mode*. Біля *Display Variable Names* поставте галочку (натисніть на неї лівою кнопкою "миші").

Тепер виберіть *Analyses Setup*. Перед вами з'явиться вікно *Analyses Setup*. Відключіть усі опції (тобто уберіть галочки) окрім опції *Multimeter*.

Тепер ви можете здійснити аналіз. Натисніть кнопку *OK*, потім *Run Analyses*.

На екрані з'явиться нове вікно: *Multimeter* (відображає конкретне значення параметру).

Для того, щоб припинити аналіз (тобто закрити цю форму) натисніть на панелі інструментів кнопку *Stop*. Для запуску аналізу натисніть кнопку:



Б.4.2 Визначення значень параметрів схеми

Вам необхідно визначити значення напруги у вузлах схеми.

Для цього потрібно підвести курсор до тієї точки схеми, де необхідно виміряти значення напруги (курсор змінить своє зображення) та натисніть на ліву кнопку "миші".

У вікні *Multimeter* ви побачите значення параметру в вибраній точці.

Запишіть дані в таблицю Б.1

Таблиця Б.1 – Значення напруги в вузлах схеми

№ вузла	Напруга

Додаток В

Завдання до лабораторної роботи №2

Таблиця В.1 – Вихідні дані для побудови області розрахунку

№ вар.	n	m	k	h
1	4	4	1	2
2	4	5	2	3
3	5	4	1	3
4	4	4	2	4
5	4	5	1	4
6	5	4	2	5
7	4	4	1	5
8	4	5	1	2
9	5	4	2	3
10	4	4	2	6
11	4	5	1	6
12	5	4	2	7
13	5	4	1	2
14	5	5	2	3
15	4	4	2	3
16	5	5	1	2
17	4	5	2	4
18	4	5	1	5
19	5	4	1	7
20	4	5	2	2
21	5	4	2	4
22	4	4	1	3
23	5	4	1	4
24	5	4	2	5
25	4	4	2	5

Додаток Д

Завдання до лабораторної роботи №3

Таблиця Д.1 – Вихідні дані для розрахунків

Вар. №	Точки										
	А		В		С		D		Е		F
	(x,y)	t, °C	(x,y)	t, °C	(x,y)	t, °C	(x,y)	t, °C	(x,y)	t, °C	(x,y)
1	(1,4)	20	(3,4)	50	(4,2)	40	(3,0)	30	(0,0)	20	(3,1)
2	(0,2)	40	(2,3)	60	(4,3)	50	(5,0)	20	(0,0)	30	(3,2)
3	(0,2)	20	(1,4)	30	(5,4)	50	(5,0)	40	(0,0)	20	(4,2)
4	(0,4)	40	(4,4)	50	(5,2)	60	(5,0)	40	(0,0)	20	(3,2)
5	(0,4)	30	(4,3)	60	(5,1)	60	(4,0)	50	(0,0)	30	(3,2)
6	(0,3)	30	(2,3)	50	(5,4)	60	(5,0)	40	(0,0)	30	(3,2)
7	(1,3)	30	(3,3)	50	(5,4)	40	(5,0)	40	(0,0)	20	(4,2)
8	(1,3)	20	(3,4)	50	(5,2)	60	(5,0)	60	(0,0)	20	(3,2)
9	(0,4)	30	(5,4)	40	(3,2)	60	(5,0)	50	(0,0)	30	(3,1)
10	(0,3)	20	(3,4)	50	(5,4)	60	(5,0)	50	(0,0)	30	(3,2)
11	(0,4)	30	(4,3)	60	(5,1)	50	(4,0)	40	(0,0)	30	(3,2)
12	(2,4)	30	(4,3)	50	(4,2)	60	(3,0)	40	(0,0)	20	(3,2)
13	(0,3)	30	(3,4)	50	(5,3)	40	(5,0)	40	(0,0)	20	(3,2)
14	(0,3)	30	(3,3)	60	(5,2)	30	(3,0)	30	(0,0)	20	(3,2)
15	(1,3)	40	(3,3)	30	(5,4)	50	(5,0)	40	(0,0)	20	(4,2)
16	(1,2)	20	(0,4)	60	(5,4)	50	(5,0)	50	(0,0)	20	(3,2)
17	(0,2)	40	(1,4)	50	(5,4)	60	(5,0)	40	(0,0)	20	(4,2)
18	(0,4)	30	(4,4)	50	(5,2)	60	(5,0)	40	(0,0)	20	(3,2)
19	(0,4)	30	(3,2)	50	(5,2)	60	(5,0)	40	(0,0)	30	(4,1)
20	(0,3)	30	(2,3)	50	(5,4)	40	(5,0)	40	(0,0)	20	(3,2)
21	(1,3)	30	(3,3)	50	(5,4)	60	(5,0)	60	(0,0)	20	(4,2)
22	(1,3)	20	(3,4)	50	(5,2)	60	(5,0)	50	(0,0)	30	(3,2)
23	(0,4)	30	(5,4)	50	(3,2)	50	(5,0)	50	(0,0)	30	(3,1)
24	(2,3)	20	(3,3)	40	(3,1)	40	(3,0)	40	(0,0)	20	(4,1)

Додаток Ж

Завдання до лабораторної роботи №4

Таблиця Ж.1 – Вхідні дані для побудови імітаційної моделі СМО

Назва змінюваного параметра	№ варіанта									
	1	2	3	4	5	6	7	8	9	10
Кількість годин роботи моделі	100	200	300	400	500	600	700	800	900	1000
Час обробки деталей на станках, хв.										
– токарний						0,5	0,5	0,6	0,3	0,2
– свердлильний	0,3	0,5	0,5	0,4	0,5	0,6	0,8	0,7	0,9	0,7
– зубостругальний	0,4	0,6	0,5	0,6	0,3					
	0,2	0,4	0,4	0,2	0,4	0,4	0,2	0,8	0,2	0,9
Кількість деталей, що будуть оброблюватися	1000	2000	1000	3000	2000	1000	2000	4000	5000	3000
Частота надходження деталей до моделі, хв.	0,5	0,6	0,2	0,7	0,6	0,5	0,3	0,5	0,8	0,9

Додаток К

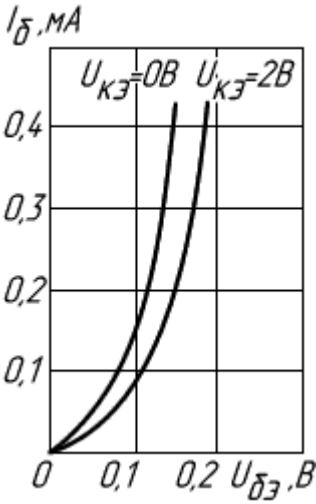
Завдання до лабораторної роботи №5

Таблиця К.1 – Дані ТУ на транзистори

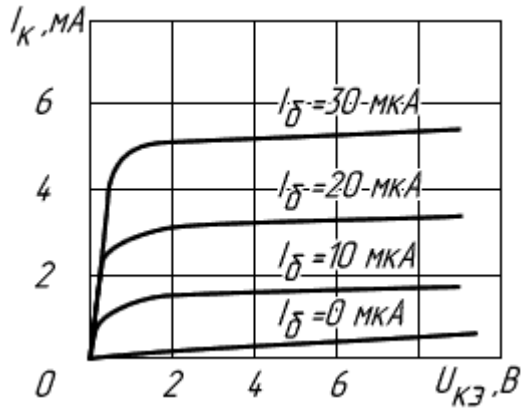
Вар. №	Тип	Параметри						
		$U_{BE \text{ нас,}}$ В	$I_{B \text{ нас,}}$ А	h_{21B} , відн. од.	h_{11E} , Ом	$U_{KE \text{ нас,}}$ В	$I_{K \text{ нас,}}$ А	$f_{гр}$, Гц
1	ГТ701А р	0,5	0,40	0,90	55,0	2,0	12,000	50Е3
2	КТ604А н	0,7	0,20	0,90	60,0	8,0	0,200	40Е6
3	ГТ403А р	0,8	0,40	0,95	100,0	0,5	1,250	8Е3
4	КТ312А н	1,1	0,30	0,90	38,0	0,8	0,030	80Е6
5	ГТ322А р	1,2	0,08	0,95	68,0	3,0	0,005	80Е6
6	ГТ311Е р	2,0	0,09	0,97	50,0	1,2	0,050	250Е6
7	ГТ310Б р	0,5	0,09	0,98	190,0	1,8	0,001	160Е6
8	ГТ308Б р	0,5	0,05	0,98	150,0	1,2	0,050	120Е6
9	ГТ305А р	0,7	0,20	0,97	66,6	0,5	0,040	140Е6
10	ГТ108А р	0,6	0,10	0,96	37,5	0,4	0,050	500Е3

Таблиця К.2 – Дані ТУ на транзистори

Вар. №	Тип	Параметри						
		$ h_{21E} $, відн. од.	f_m , МГц	C_E , пФ	U_{BE} , В	C_K , пФ	U_{KB} , В	B_{Nmin} ÷ B_{Nmax}
1	ГТ701А	11	20	40,0	25,0	8,0	65	8÷10
2	КТ604А	4	20	50,0	5,0	7,0	300	10÷40
3	ГТ403А	4	20	25,0	20,0	4,0	45	20÷60
4	КТ312А	4	20	20,0	4,0	5,0	20	10÷100
5	ГТ322А	4	20	40,0	4,5	1,8	15	20÷70
6	ГТ311Е	5	20	2,5	2,0	5,0	12	15÷80
7	ГТ310Б	8	20	15,0	3,5	4,0	12	60÷180
8	ГТ308Б	6	20	25,0	3,0	8,0	20	50÷120
9	ГТ305А	7	20	50,0	1,5	7,0	15	25÷40
10	ГТ108А	6	20	20,0	2,0	5,0	10	20÷50

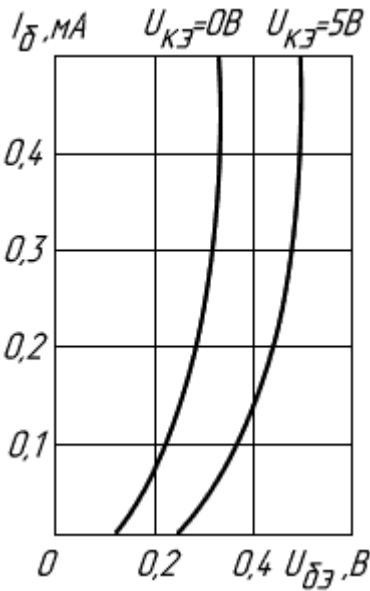


а) вхідна

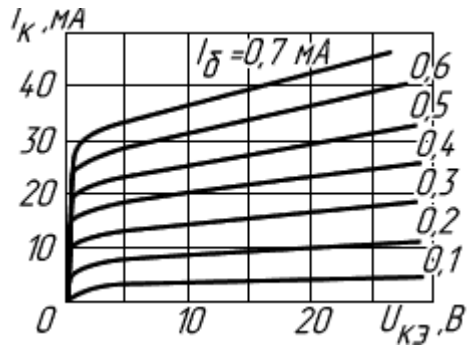


б) вихідна

Рисунок К.1 – Характеристики транзистора ГТ108А



а) вхідна



б) вихідна

Рисунок К.2 – Характеристики транзистора КТ312Б

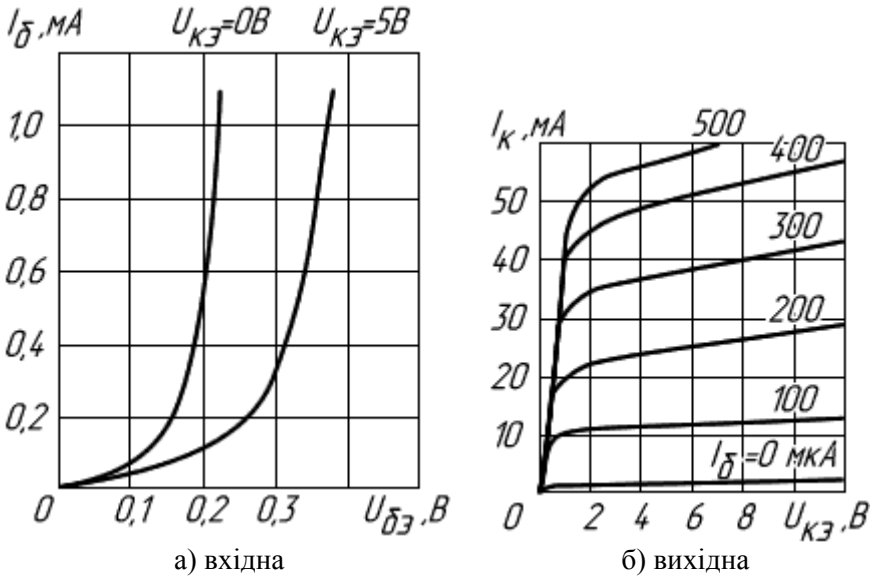


Рисунок К.3 – Характеристики транзистора ГТ305А

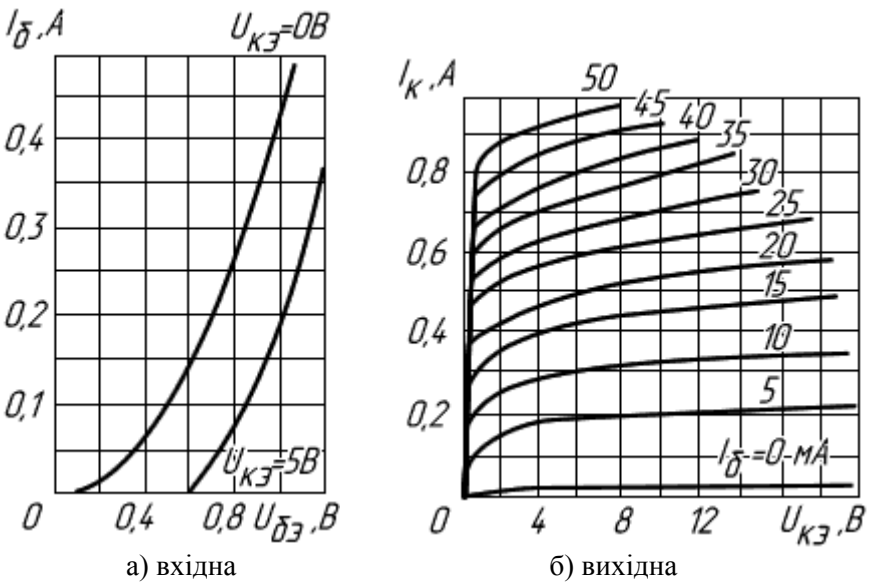
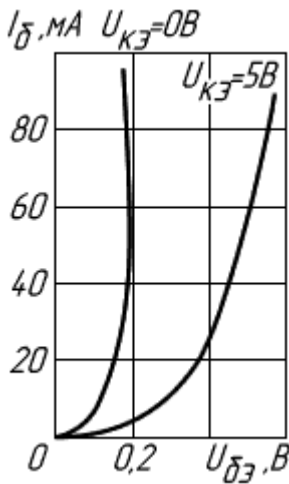
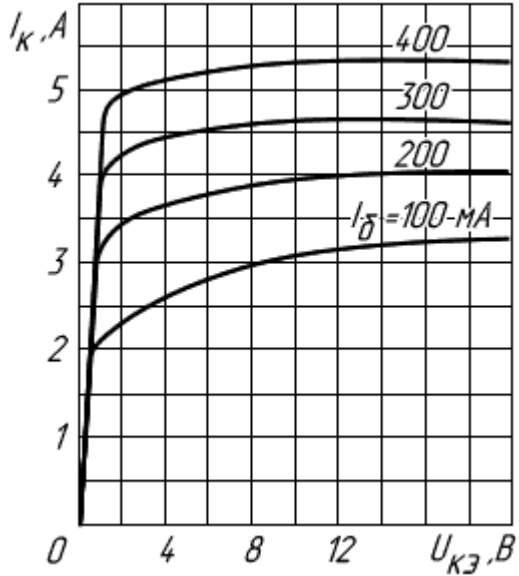


Рисунок К.4 – Характеристики транзистора ГТ403А

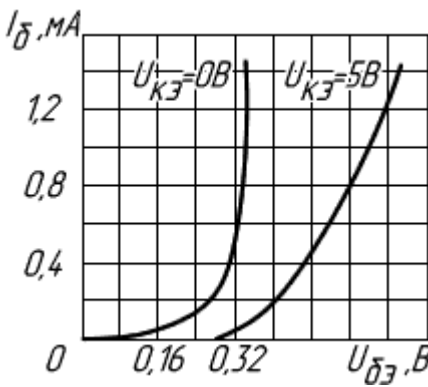


а) вхідна

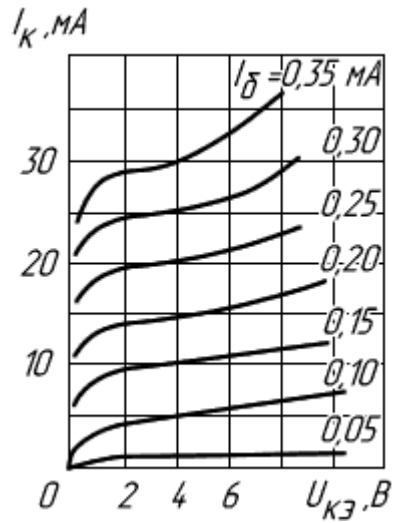


б) вихідна

Рисунок К.5 – Характеристики транзистора ГТ701А



а) вхідна



б) вихідна

Рисунок К.6 – Характеристики транзистора ГТ311Е

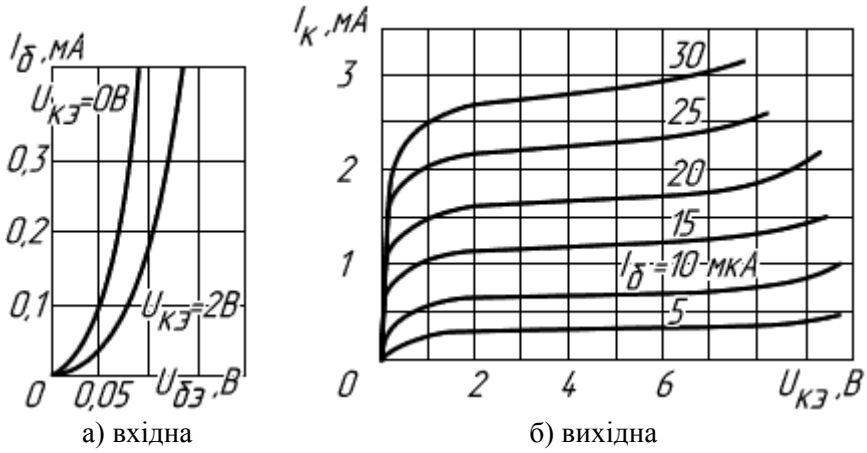


Рисунок К.7 – Характеристики транзистора ГТЗ10Б

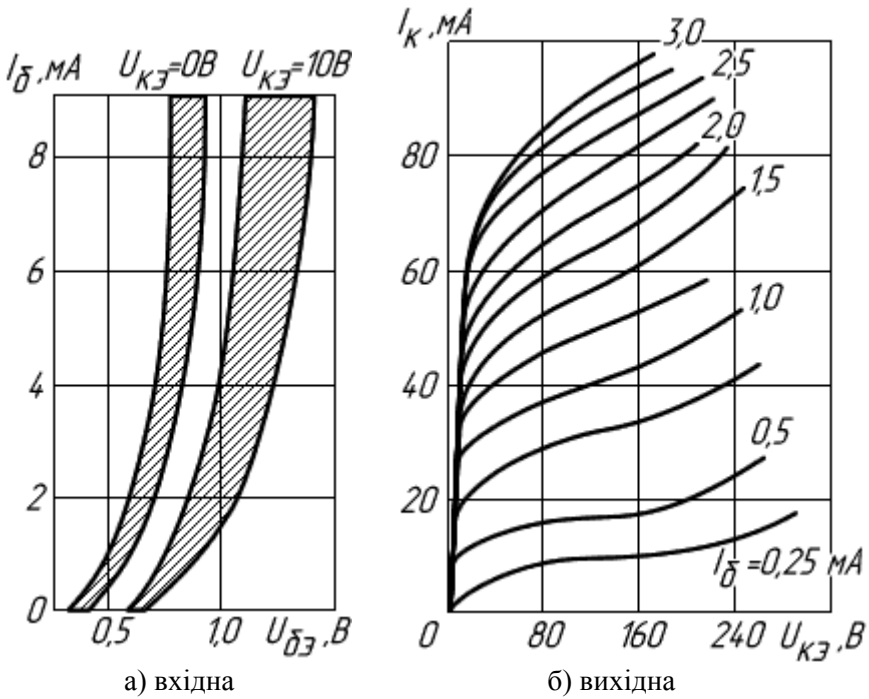


Рисунок К.8 – Характеристики транзистора КТ604А

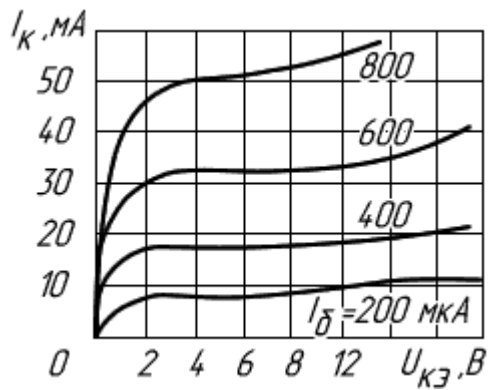
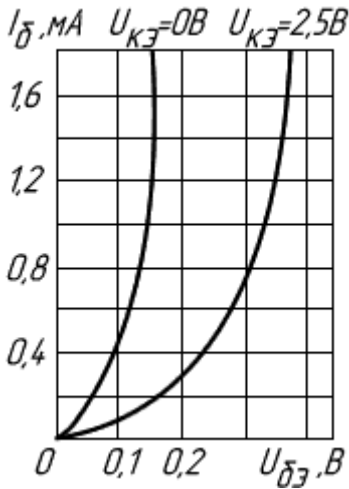


Рисунок К.9 – Характеристики транзистора ГТ308Б

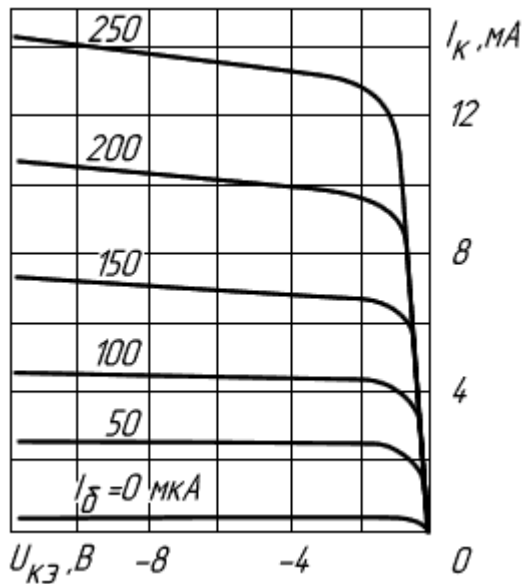
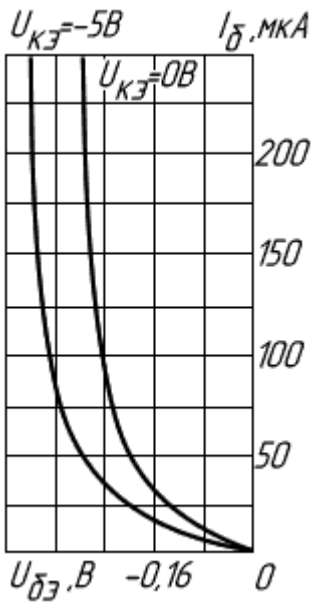


Рисунок К.10 – Характеристики транзистора КТ322А