

ФАКУЛЬТЕТ АВТОМАТИКИ, ТЕЛЕМЕХАНІКИ ТА ЗВ'ЯЗКУ

Кафедра спеціалізованих комп'ютерних систем

М.А. Мірошник

**СИСТЕМИ АВТОМАТИЗАЦІЇ ПРОЕКТУВАННЯ
ПРИСТРОЇВ І СИСТЕМ АВТОМАТИКИ.**

**ОСНОВИ СИСТЕМ АВТОМАТИЗАЦІЇ
ПРОЕКТУВАННЯ**

Конспект лекцій

Харків – 2014

Мірошник М.А. Системи автоматизації проектування пристроїв і систем автоматики. Основи систем автоматизації

проекування: Конспект лекцій. – Харків: УкрДАЗТ, 2014. – 102 с.

Даний конспект лекцій розкриває загальні теми робочої програми з дисциплін «Системи автоматизації проектування пристроїв і систем автоматики» та «Основи систем автоматизації проектування», містить лекційний матеріал, питання для самоконтролю, задачі, приклади та додатково-інформаційні дані для розв’язання задач.

Рекомендовано для студентів спеціальностей 7.091503 “Спеціалізовані комп’ютерні системи” та 7.092507 “Автоматика і автоматизація на транспорті” всіх форм навчання.

Лл. 39, бібліогр.: 10 назв.

Конспект лекцій розглянуто та рекомендовано до друку на засіданні кафедри спеціалізованих комп’ютерних систем 8 грудня 2011 р., протокол № 5/11.

Рецензент

проф. С.В. Листровой

М.А. Мірошник

СИСТЕМИ АВТОМАТИЗАЦІЇ ПРОЕКТУВАННЯ ПРИСТРОЇВ І
СИСТЕМ АВТОМАТИКИ.

ОСНОВИ СИСТЕМ АВТОМАТИЗАЦІЇ ПРОЕКТУВАННЯ

Конспект лекцій

Відповідальний за випуск Мірошник М.А.

Редактор Решетилова В.В.

Підписано до друку 06.02.12 р.

Формат паперу 60x84 1/16. Папір писальний.

Умовн.-друк.арк. 5,5. Тираж 50. Замовлення №

Видавець та виготовлювач Українська державна академія залізничного транспорту,
61050, Харків-50, майдан Фейербаха, 7.

Свідоцтво суб’єкта видавничої справи ДК № 2874 від 12.06.2007 р.

ЗМІСТ

	Вступ	5
1	Введення в автоматизоване проектування	7
1.1	Поняття проектування	7
1.2	Принципи системного підходу	8
1.3	Рівні проектування	10
1.4	Стадії проектування	13
1.5	Моделі і їх параметри в САПР	15
1.6	Проектні процедури	17
1.7	Життєвий цикл виробів	22
1.8	Структура САПР	23
1.9	Етапи проектування автоматизованих систем	26
	Питання, тести для самоконтролю	30
2	Технічне та математичне забезпечення САПР	31
2.1	Вимоги до технічного забезпечення САПР	31
2.2	Вимоги до математичних моделей і методів у САПР	40
2.3	Математичні моделі та методи аналізу на макрорівні	48
	Питання, тести для самоконтролю	49
	Індивідуальні практичні завдання	51
3	Автоматизація проектування в радіоелектроніці	52
3.1	Етапи та стадії проектування	52
3.2	Блочно-ієрархічний підхід при технічному проектуванні	53
3.3	Структура проектування схеми вузла	56
3.4	Труднощі традиційного проектування	59
3.5	Ручне та автоматизоване використання ЕОМ у процесі проектування	61
3.6	Комплекс засобів системи автоматизованого проектування	63
	Питання, тести для самоконтролю	66
	Типові завдання та приклади їх розв'язання	67
	Індивідуальні практичні завдання	70
4	Проектування електронних пристроїв та систем автоматики за допомогою спеціалізованих САПР	72
4.1	Огляд САПР друкованих плат	72

4.2	Етапи проектування друкованої плати в САПР P-CAD	74
4.3	Система проектування пристроїв за допомогою САПР OrCAD	77
4.4	Редактор топології друкованих плат OrCAD Layout...	
	Питання, тести для самоконтролю	88
	Типові завдання та приклади їх розв'язання	89
	Індивідуальні практичні завдання	97
	Висновки	101
	Список літератури	102

ВСТУП

Підвищення ефективності виробництва та якості продукції, що випускається, зростання числа виробів немислимо без автоматизації, впровадження і розвитку систем автоматизованого проектування (САПР). Автоматизовані системи проектування пристроїв та систем управління є досить широким класом систем, що забезпечують комплекс робіт, пов'язаний зі здійсненням проектування технологічних процесів, проектуванням оснащення технологічних процесів, складанням програм для автоматизованого технологічного обладнання. Такий склад визначає структуру курсу, пов'язану з створенням великого різновиду документів, кінець кінцем націлену на автоматизовану підготовку технології виробництва.

Автоматизація процесу проектування пристроїв та систем управління дозволяє скоротити терміни розробки, трудомісткість проектування, одержати вільну від помилок проектну документацію. Можливість аналізу великої кількості варіантів конструкцій та технологій дозволяє вибрати найбільш оптимальний і тим самим підвищити якість проектованого виробу. Здійснення росту в тій галузі сприяє підвищенню продуктивності праці, прискоренню прогресу в промисловості.

Метою цього конспекту лекцій є аналіз проблем і виклад наявного досвіду створення та використання автоматизованих систем розв'язання завдань виробництва. При цьому показана загальна концепція наскрізного конструктивно-технологічного проектування на базі комп'ютерних технологій.

Орієнтація на автоматизовані методи прийняття конструкторсько-технологічних рішень вимагає освоєння не тільки теоретичних основ, але й набуття навичок використання комп'ютерної техніки для прийняття оптимальних проектних рішень.

Предмет курсу: вивчення загальних відомостей про об'єкти, моделі і задачі автоматизованого проектування; основних понять САПР; призначення, складу, принципів та особливостей функціонування різних систем автоматизованого проектування; вивчення та вибір ефективних рішень щодо моделей комп'ютерних систем, процесів, що протікають в си-

стемах, методів, які дозволяють розв'язувати завдання аналізу характеристик і ідентифікації параметрів, методики розв'язання проектних завдань, математичного забезпечення процедур аналізу та синтезу проектних рішень, освітлення методики концептуального проектування, складних систем, питання інтеграції САПР із автоматизованими системами керування.

Дисципліна викладається з метою вивчення принципів побудови та організації САПР, методики використання САПР для розв'язання задач автоматизації проектування пристроїв та систем автоматизації (ПСА); вивчення загальних відомостей про об'єкти, моделі і задачі автоматизованого проектування; призначення, складу, принципів та особливостей функціонування різних систем автоматизованого проектування; надання цілісної картини процесу автоматизованого проектування ПСА, основних проблем і підходів до їх розв'язання.

Курс базується на загальному вмінні володіти комп'ютером та знаннях, отриманих при вивченні дисциплін, пов'язаних з електричними колами, електронікою, схемотехнікою та програмуванням, зокрема, дисциплін “Електроніка та мікросхемотехніка, “Математичні моделі в розрахунках на ЕОМ” і “Проектування програмного забезпечення”.

В результаті вивчення курсу студент повинен:

знати: основні принципи побудови та роботи сучасних САПР; методику автоматизованого проектування ПСА засобами САПР; галузі використання та особливості застосування САПР при розв'язанні задач автоматизації; особливості застосування САПР ПСА при розв'язанні задач автоматизації проектування друкованих плат, монтажних схем, електроустаткування допомогою систем OR-CAD, P-CAD та ACCEL EDA; основні принципи роботи сучасних систем автоматизованого проектування на прикладі Компас 3D, Solid Works.

Вміти: практично виконувати наскрізний цикл проектування друкованих модулів засобами САПР; здійснювати технологічну підготовку виробництва друкованих модулів засобами САПР; здійснювати підготовку та випуск конструкторської та технологічної документації засобами САПР;

мати уявлення про перспективи розвитку сучасних САПР пристроїв та систем автоматики.

1 ВВЕДЕННЯ В АВТОМАТИЗОВАНЕ ПРОЕКТУВАННЯ

1.1 Поняття проектування

Проектування технічного об'єкта – створення, перетворення й подання в прийнятій формі образу цього ще не існуючого об'єкта. Образ об'єкта або його складових частин може створюватися в уяві людини в результаті творчого процесу або генеруватися відповідно до деяких алгоритмів у процесі взаємодії людини й ЕОМ. У кожному разі інженерне проектування починається при наявності вираженої потреби суспільства в деяких технічних об'єктах, якими можуть бути об'єкти будівництва, промислові виробы або процеси. Проектування містить у собі розробку технічної пропозиції та (або) *технічного завдання* (ТЗ), що відбивають ці потреби, і реалізацію ТЗ у вигляді проектної документації.

Найчастіше ТЗ подають у вигляді деяких документів, і воно є вихідним (первинним) описом об'єкта. Результатом проектування, як правило, служить повний комплект документації, що містить достатні відомості для виготовлення об'єкта в заданих умовах. Ця документація і є *проект*, точніше остаточний опис об'єкта. Більш коротко, проектування – процес, що полягає в одержанні й перетворенні вихідного опису об'єкта в остаточний опис на основі виконання комплексу робіт дослідницького, розрахункового й конструкторського характеру.

Перетворення вихідного опису в остаточний породжує ряд проміжних описів, що підводять підсумки розв'язання деяких задач і використовуваних для обговорення й прийняття рішень для закінчення або продовження проектування. Такі проміжні описи називають *проектними рішеннями*.

Проектування, при якому всі проектні рішення або їхня частина одержуються шляхом взаємодії людини й ЕОМ, називають *автоматизованим проектуванням*, на відміну від ручного (без використання ЕОМ) або автоматичного (без участі людини на проміжних етапах). Система, що реалізує автоматизоване проектування, являє собою *систему*

автоматизованого проектування (САПР, в англomовному написанні CAD System – Computer Aided Design System).

Автоматичне проектування можливо лише в окремих випадках для порівняно нескладних об'єктів. Переважаючим в цей час є автоматизоване проектування.

Проектування складних об'єктів засновано на застосуванні ідей і принципів, викладених у ряді теорій і підходів. Найбільш загальним підходом є системний підхід, ідеями якого пронизані різні методики проектування складних систем.

1.2 Принципи системного підходу

Проектування складних об'єктів засновано на застосуванні ідей і принципів, викладених у ряді теорій і підходів. Найбільш загальним підходом є *системний підхід*, ідеями якого пронизані різні методики проектування складних систем.

Основний загальний принцип системного підходу полягає в розгляді частин досліджуваного явища або складної системи з урахуванням їх взаємодії. Системний підхід містить у собі виявлення структури системи, типізацію зв'язків, визначення атрибутів, аналіз впливу зовнішнього середовища, формування моделі системи, дослідження моделі й *оптимізацію* її структури й функціонування.

Системи автоматизованого проектування й керування відносяться до числа найбільш складних сучасних систем. Їхнє проектування й використання неможливі без системного підходу. Інтерпретація й конкретизація системного підходу мають місце в ряді відомих підходів з іншими назвами, які також можна розглядати як компоненти системотехніки. Такими є структурні, блочно-ієрархічні, об'єктно-орієнтовані підходи.

При *структурному* підході, як різновиді системного, потрібно синтезувати варіанти системи з компонентів (блоків) і оцінювати варіанти при їхньому частковому переборі з попереднім прогнозуванням характеристик компонентів.

Блочно-ієрархічний підхід до проектування використовує ідеї декомпозиції складних описів об'єктів і відповідно засобів їхнього створення на ієрархічні рівні й аспекти, вводить поняття

стилю проектування (висхідне й спадне), встановлює зв'язок між параметрами сусідніх ієрархічних рівнів.

Ряд важливих структурних принципів, які використовуються при розробці інформаційних систем і їх програмного забезпечення (ПЗ), виражений у підході, що має назву *об'єктно-орієнтоване проектування* (ООП). Такий підхід має наступні переваги в розв'язанні проблем керування складністю й інтеграції ПЗ:

- вносить у моделі програм більшу структурну визначеність, розподіляючи подані в програмі дані й процедури між класами об'єктів;
- скорочує обсяг специфікацій завдяки введенню в описи ієрархії об'єктів і відносин наслідування між властивостями об'єктів різних рівнів ієрархії;
- зменшує ймовірність перекручування даних внаслідок помилкових дій за рахунок обмеження доступу до певних категорій даних в об'єктах.

Опис у кожному класі об'єктів припустимих звертань до них і прийнятих форматів повідомлень полегшує узгодження й інтеграцію ПЗ.

Для всіх підходів до проектування складних систем характерні також такі особливості:

а) структуризація процесу проектування, що виражає декомпозицією проектних задач і документації, виділенням стадій, етапів, проектних процедур. Ця структуризація є сутністю блочно-ієрархічного підходу до проектування;

б) ітераційний характер проектування;

в) типізація й уніфікація проектних рішень і засобів проектування.

До характеристик складних систем часто відносять такі поняття:

- *цілеспрямованість* – властивість штучної системи, що виражає призначення системи. Ця властивість необхідна для оцінки ефективності варіантів системи;
- *цілісність* – властивість системи, що характеризує взаємозв'язок елементів і наявність залежності вихідних параметрів від параметрів елементів, при цьому більшість вихідних

параметрів не є простим повторенням або сумою параметрів елементів;

- *ієрархічність* – властивість складної системи, що виражає можливість і доцільність її ієрархічного опису, тобто подання у вигляді декількох рівнів, між компонентами яких є відносини ціле—частина.

Складовими частинами системотехніки є такі основні розділи:

- ієрархічна структура систем, організація їхнього проектування;

- аналіз і моделювання систем;

- синтез і оптимізація систем.

Моделювання має дві чітко помітні задачі:

- а) створення моделей складних систем (в англійській мові – modeling);

- б) аналіз властивостей систем на основі дослідження їхніх моделей (simulation).

Синтез також підрозділяють на дві задачі:

- а) синтез структури проєктованих систем (структурний синтез);

- б) вибір чисельних значень параметрів елементів систем (параметричний синтез).

Ці задачі відносяться до галузі прийняття проєктних рішень.

Моделювання й оптимізацію бажано виконувати з урахуванням статистичної природи систем. Детермінованість – лише окремий випадок. При проектуванні характерна недостача достовірних вихідних даних, невизначеність умов прийняття рішень. Врахування статистичного характеру даних при моделюванні значною мірою засновано на методі статистичних випробувань (методі Монте-Карло), а прийняття рішень – на використанні нечітких множин, експертних систем, еволюційних обчислень.

1.3 Рівні проектування

При використанні блочно-ієрархічного підходу до проектування уявлення про проєктовану систему розчленовують на ієрархічні рівні. На верхньому рівні використовують найменш

деталізоване уявлення, що відбиває тільки найбільш загальні риси й особливості проектованої системи. На наступних рівнях ступінь подробиць опису зростає, при цьому розглядають вже окремі блоки системи, але з урахуванням впливів на кожний з його сусідів. Такий підхід дозволяє на кожному ієрархічному рівні формулювати задачі прийнятної складності, що піддаються розв'язанню за допомогою наявних засобів проектування. Розбивка на рівні повинна бути такою, щоб документація на блок будь-якого рівня була доступна для огляду й сприймана однією людиною.

Блочно-ієрархічний підхід є декомпозиційним підходом, що заснований на розбивці складної задачі великої розмірності на послідовно та (або) паралельно розв'язувані групи задач малої розмірності, що істотно скорочує вимоги до використовуваних обчислювальних ресурсів або до часу розв'язання задач.

Можна говорити не тільки про ієрархічні рівні специфікацій, але й про ієрархічні рівні проектування, розуміючи під кожним з них сукупність специфікацій деякого ієрархічного рівня разом з постановками задач, методами одержання описів і розв'язання проектних задач, що виникають.

Список ієрархічних рівнів для кожного застосування може бути специфічним, але для більшості застосувань характерно таке найбільш велике виділення рівнів:

- *системний рівень*, на якому розв'язують найбільш загальні задачі проектування систем, машин і процесів; результати проектування подають у вигляді структурних схем, генеральних планів, схем розміщення устаткування, діаграм потоків даних тощо;

- *макрорівень*, на якому проектують окремі пристрої, вузли машин і приладів; результати подають у вигляді функціональних, принципівих і кінематичних схем, складальних креслень тощо;

- *мікрорівень*, на якому проектують окремі деталі й елементи машин і приладів.

Для кожного застосування число виділюваних рівнів і їх найменувань можуть бути різними. Так, у радіоелектроніці мікрорівень часто називають компонентним, макрорівень – *схемотехнічним рівнем*. Між *схемотехнічним* і *системним*

рівнями вводять рівень, що має назву *функціонально-логічного рівня*. В обчислювальній техніці системний рівень підрозділяють на рівні проектування ЕОМ (обчислювальних систем) і обчислювальних мереж. У машинобудуванні є рівні деталей, вузлів, машин, комплексів.

Залежно від послідовності розв'язання задач ієрархічних рівнів розрізняють спадне, висхідне й змішане проектування (*стили проектування*). Послідовність розв'язання задач від нижніх рівнів до верхнього характеризує *висхідне проектування*, зворотна послідовність приводить до *спадного проектування*, у *змішаному стилі* є елементи як висхідного, так і спадного проектування. У більшості випадків для складних систем віддають перевагу спадному проектуванню. Відзначимо однак, що при наявності заздалегідь спроектованих складених блоків (пристроїв) можна говорити про змішане проектування.

Невизначеність і нечіткість вихідних даних при спадному проектуванні (ще не спроектовані компоненти) або вихідних вимог при висхідному проектуванні (оскільки ТЗ є на всю систему, а не на її частини) обумовлюють необхідність прогнозування відсутніх даних з наступним їхнім уточненням, тобто послідовного наближення до остаточного рішення (ітераційність проектування).

Поряд з декомпозицією описів на ієрархічні рівні застосовують поділ уявлень про спроектовані об'єкти на *аспекти*.

Аспект опису – опис системи або її частин з деякої обговореної точки зору, обумовленої функціональним, фізичними або іншим типом відносин між властивостями й елементами.

Розрізняють функціональний, інформаційний, структурний і поведінковий (процесний) аспекти. Функціональний опис відносять до функцій системи й найчастіше подають його функціональними схемами. Одержання функціональних описів часто називають *функціональним проектуванням*.

Інформаційний опис містить у собі основні поняття предметної галузі (сутності), словесне пояснення або числові значення характеристик (атрибутів) використовуваних об'єктів, а також опис зв'язків між цими поняттями й характеристиками. Інформаційні моделі можна подавати графічно (графи, діаграми сутність-відношення), у вигляді таблиць або списків. Одержання

інформаційних описів часто називають інформаційним проектуванням або стосовно до створення баз даних – інфологічним проектуванням.

Структурний опис відноситься до морфології системи, характеризує складові частини системи і їх з'єднання й може бути поданий структурними схемами, а також різного роду конструкторською документацією. Одержання конструкторської документації, тобто опис геометричних форм виробів, складу компонентів і їхнього просторового розміщення, називають *конструкторським проектуванням*.

Поведінковий опис характеризує процеси функціонування (алгоритми) системи й (або) технологічні процеси створення системи. Розробка алгоритмів і програмного забезпечення систем є предметом *алгоритмічного проектування*, а розробка технологічних процесів виготовлення виробів – предметом *технологічного проектування*.

Іноді аспекти описів зв'язують із підсистемами, функціонування яких засновано на різних фізичних процесах.

В загальному випадку виділення аспектів опису може бути неоднозначним. Треба виділити такі аспекти, як функціональне (розробка принципів дії, структурних, функціональних, принципових схем), конструкторське (визначення форм і просторового розташування компонентів виробів), алгоритмічне (розробка алгоритмів і програмного забезпечення) і технологічне (розробка технологічних процесів) проектування систем.

1.4 Стадії проектування

Стадії проектування – найбільш великі частини проектування, як процесу, що розвивається в часі. У загальному випадку виділяють стадії *науково-дослідних робіт* (НДР), *ескізного проекту* або *дослідно-конструкторських робіт* (ДКР), *технічного, робочого* проектів, *випробувань дослідних зразків* або *дослідних партій*. Стадію НДР іноді називають *передпроектними дослідженнями* або стадією *технічної пропозиції*. Очевидно, що в міру переходу від стадії до стадії ступінь подробиць й глибина опрацювання проекту зростають, і робочий проект вже повинен

бути цілком достатнім для виготовлення дослідних або серійних зразків. Близьким до визначення стадії, але менш чітко обговореним поняттям, є поняття *етапу проектування*. Проектування на початкових етапах, у процесі якого приймаються принципові проектні рішення щодо вигляду й принципів дії проєктованих пристроїв і систем, називають *концептуальним проектуванням*.

Стадії (етапи) проектування підрозділяють на складові частини, що мають назву *проектних процедур*. Прикладами проектних процедур можуть служити підготовка деталювальних креслень, аналіз кінематики, моделювання перехідного процесу, оптимізація параметрів і інші проектні задачі. У свою чергу, проектні процедури можна розчленувати на більш дрібні компоненти, що мають назву *проектних операцій*, наприклад, при аналізі міцності деталі сітковими методами операціями можуть бути побудова сітки, вибір або розрахунок зовнішніх впливів, моделювання полів напруг і деформацій, подання результатів моделювання в графічній і текстовій формах. Проектування зводиться до виконання деяких послідовностей проектних процедур – *маршрутів проектування*.

Прагнення скоротити тимчасові витрати на проектування привело до розробки методик *паралельного проектування* (сполученого проектування), при якому паралельно в часі розв'язуються задачі, зв'язані одна з одною за вхідними і вихідними даними таким чином, що для розв'язання однієї з них знадобиться знання результатів розв'язання іншої задачі. Оскільки ці результати до початку процедури паралельного проектування ще не отримані, у методиці паралельного проектування повинні бути зазначені способи завдання ще не визначених значень параметрів. Іноді розробку технічного завдання на проектування називають *зовнішнім проектуванням*, а реалізацію ТЗ – *внутрішнім проектуванням*.

У ТЗ на проектування об'єкта вказують такі дані:

- а) призначення об'єкта;
- б) умови експлуатації. Поряд з якісними характеристиками (поданими у вербальній формі) є числові параметри, що мають назву *зовнішніх параметрів*, для яких зазначені області

припустимих значень. Приклади зовнішніх параметрів: температура навколишнього середовища, зовнішні сили, електричні напруги, навантаження тощо;

в) вимоги до *вихідних параметрів*, тобто до величин, що характеризують властивості об'єкта, що цікавлять споживача. Ці вимоги виражені у вигляді умов *працездатності*: $y_i R T_i$, де y_i – i -й вихідний параметр, $R \in \{=, <, >, \leq, \geq\}$ – вид відносини; T_i – норма i -го вихідного параметра. У випадку, якщо R – відношення рівності, потрібно задати необхідну точність виконання рівності.

1.5 Моделі і їх параметри в САПР

В автоматизованих проектних процедурах замість ще неіснуючого проектованого об'єкта оперують деяким квазіоб'єктом – *моделлю*, що відбиває деякі властивості об'єкта, які цікавлять дослідника. Модель може бути фізичним об'єктом (макет, стенд) або специфікацією. Серед моделей-специфікацій розрізняють функціональні, поведінкові, інформаційні, структурні моделі. Ці моделі називають *математичними моделями*, якщо вони формалізовані засобами апарата й мови математики.

У свою чергу, математичні моделі можуть бути геометричними, топологічними, динамічними, логічними й т.п., якщо вони відбивають відповідні властивості об'єктів. Поряд з математичними моделями при проектуванні використовують функціональні моделі, інформаційні моделі у вигляді діаграм сутність-відношення, геометричні моделі (креслення). Надалі, якщо немає спеціального застереження, під словом "модель" будемо мати на увазі математичну модель.

Математична функціональна модель у загальному випадку являє собою алгоритм обчислення вектора вихідних параметрів Y при заданих векторах параметрів елементів (*внутрішніх параметрів*) X і *зовнішніх параметрів* Q .

Математичні моделі можуть бути символічними й чисельними. При використанні *символьних* моделей оперують не значеннями величин, а їхніми символічними позначеннями (ідентифікаторами). *Чисельні* моделі можуть бути

аналітичними моделями, тобто їх можна подати у вигляді явно виражених залежностей вихідних параметрів Y від параметрів внутрішніх X і зовнішніх Q , або *алгоритмічними моделями*, у яких зв'язок Y , X і Q заданий неявно у вигляді алгоритму *моделювання*. Найважливіший окремий випадок алгоритмічних моделей – *імітаційні моделі*, вони відображають процеси в системі при наявності зовнішніх впливів на систему. Інакше кажучи, імітаційна модель – це алгоритмічна поведінкова модель.

Класифікацію математичних моделей виконують також за рядом інших ознак.

Так, залежно від приналежності до того або іншого ієрархічного рівня виділяють моделі рівнів системного, функціонально-логічного, макрорівня (зосередженого) і мікрорівня (розподіленого).

За характером використовуваного для опису математичного апарата розрізняють моделі лінгвістичні, теоретико-множинні, абстрактно-алгебраїчні, нечіткі, автоматні й т.п.

Наприклад, на системному рівні переважно застосовують моделі систем масового обслуговування й мережі Петрі, на функціонально-логічному рівні – автоматні моделі на основі апарата передатних функцій або кінцевих автоматів, на макрорівні – системи алгебро-диференціальних рівнянь, на мікрорівні – диференціальні рівняння в частинних похідних. Особливе місце займають геометричні моделі, що використовуються в системах конструювання.

Крім того, введені поняття *повних моделей* і *макромоделей*, моделей статичних і динамічних, детермінованих і стохастичних, аналогових і дискретних, символічних і чисельних.

Повна модель об'єкта на відміну від макромоделі описує не тільки процеси на зовнішніх виводах об'єкта, якій моделюється, але й внутрішні для об'єкта процеси.

Статичні моделі описують статичні стани, у них не є присутнім час у функції незалежної змінної. Динамічні моделі відбивають поведінку системи, тобто в них обов'язково використовується час.

Стохастичні й детерміновані моделі розрізняються залежно від врахування або неврахування випадкових факторів.

Інформаційні моделі використовують насамперед при інфологічному проектуванні баз даних (БД) для опису зв'язків між одиницями інформації.

Найбільші труднощі виникають при створенні моделей слабкоструктурованих систем, що характерно насамперед для системного рівня проектування. Тут значна увага приділяється експертним методам. У теорії систем сформульовані загальні рекомендації з підбору експертів при розробці моделі, організації експертизи, по обробці отриманих результатів. Досить загальний підхід до побудови моделей складних слабкоструктурованих систем виражений у методиках IDEF.

Звичайно в імітаційних моделях фігурують величини, що характеризують стан модельованої системи й мають назву *фазових змінних*.

В *аналогових моделях* фазові змінні – безперервні величини, у *дискретних моделях* – дискретні, в окремому випадку дискретні моделі є логічними (булівськими), у них стан системи і її елементів описується булівськими величинами. У ряді випадків корисне застосування *змішаних моделей*, у яких одна частина підсистем характеризується аналоговими моделями, інша – логічними.

Вихідні параметри систем можуть бути двох типів. По-перше, це параметри-функціонали, тобто функціонали залежностей. Приклади таких параметрів: амплітуди сигналів, тимчасові затримки, потужності розсіювання й т.п. По-друге, це параметри, що характеризують здатність проектованого об'єкта працювати при певних зовнішніх умовах. Ці вихідні параметри є граничними значеннями діапазонів зовнішніх змінних, у яких зберігається працездатність об'єкта.

1.6 Проектні процедури

Створити проект об'єкта (виробу або процесу) означає вибрати структуру об'єкта, визначити значення всіх його параметрів і подати результати у встановленій формі. Результати (проектна документація) можуть бути виражені у вигляді креслень, схем, пояснювальних записок, програм для програмно-керованого технологічного устаткування й інших документів на папері або на машинних носіях інформації.

Розробка (або вибір) структури об'єкта є проектна процедура, що має назву *структурного синтезу*, а розрахунок (або вибір) значень параметрів елементів X – процедура *параметричного синтезу*.

Задача структурного синтезу формулюється в системотехніці як задача прийняття рішень (ЗПР). Її суть полягає у визначенні мети, множини можливих рішень і обмежуючих умов.

Класифікацію ЗПР здійснюють за рядом ознак. За числом критеріїв розрізняють задачі одно- і багатокритеріальні. За ступенем невизначеності розрізняють ЗПР детерміновані, ЗПР в умовах ризику – при наявності у формулюванні задачі випадкових параметрів, ЗПР в умовах невизначеності, тобто при неповноті або невірогідності вихідної інформації.

Реальні задачі проектування, як правило, є багатокритеріальними. Одна з основних проблем постановки багатокритеріальних задач – встановлення правил переваги варіантів. Способи зведення багатокритеріальних задач до однокритеріальних і наступні шляхи розв'язання вивчаються в дисциплінах, присвячених методам оптимізації й математичному програмуванню.

Наявність випадкових факторів ускладнює розв'язання ЗПР. Основні підходи до розв'язання ЗПР в умовах ризику полягають або в розв'язанні "для найгіршого випадку", або в урахуванні в цільовій функції математичного очікування й дисперсії вихідних параметрів. У першому випадку задачу вирішують як детерміновану при завищених вимогах до якості розв'язання, що є головним недоліком підходу. У другому випадку вірогідність результатів розв'язання набагато вище, але виникають труднощі з оцінкою цільової функції. Застосування методу Монте-Карло у випадку алгоритмічних моделей стає єдиною альтернативою й, отже, для розв'язання потрібні значні обчислювальні ресурси.

Існують дві групи ЗПР в умовах невизначеності. Одна з них вирішується при наявності протидії розумного супротивника. Такі задачі вивчаються в теорії ігор, для задач проектування в техніці вони не характерні. У другій групі досягненню мети протидіють сили природи. Для їхнього розв'язання корисно використати теорію й методи нечітких множин.

При синтезі структури автоматизованої системи постановка задачі повинна включати як вихідні дані такі відомості:

- множина функцій, що виконуються системою (інакше кажучи, множина робіт, кожна з яких може складатися з однієї або більше операцій); можливо, що в цій множині є часткова впорядкованість робіт, що може бути подано у вигляді орієнтованого графа, у якому вершини відповідають роботам, а дуги - відносинам порядку;

- типи припустимих для використання серверів (машин), що виконують функції системи;

- множина зовнішніх джерел і споживачів інформації;

- у багатьох випадках задається також деяка вихідна структура системи у вигляді взаємозалежної сукупності серверів певних типів; ця структура може розглядатися як узагальнена надлишкова або як варіант першого наближення;

- різного роду обмеження, зокрема, обмеження на витрати матеріальних ресурсів і (або) на часи виконання функцій системи.

Задача полягає в синтезі (або корекції) структури, визначенні типів серверів (програмно-апаратних засобів), розподілі функцій по серверах таким чином, щоб досягався екстремум цільової функції при виконанні заданих обмежень.

Конструювання, розробка технологічних процесів, оформлення проектної документації – окремі випадки структурного синтезу.

Задачу параметричного синтезу називають параметричною оптимізацією (або оптимізацією), якщо її вирішують як задачу математичного програмування, тобто

$$\text{extr } F(\mathbf{X}), \mathbf{X} \in \mathbf{D}_x,$$

де $F(\mathbf{X})$ – цільова функція; \mathbf{X} — вектор керованих параметрів (що мають назву також проектних або таких, що варіюються); $\mathbf{D}_x = \{ \mathbf{X} \mid \Phi(\mathbf{X}) < 0, \Psi(\mathbf{X}) = 0 \}$ – припустима область; $\Phi(\mathbf{X})$ і $\Psi(\mathbf{X})$ – функції-обмеження.

Наступна після синтезу група проектних процедур – процедури *аналізу*. Мета аналізу – одержання інформації про характер функціонування й значення вихідних параметрів \mathbf{Y} при

заданих структурі об'єкта, відомостях про зовнішні параметри Q й параметри елементів X . Якщо задані фіксовані значення параметрів X і Q , то має місце процедура *одноваріантного аналізу*. Одноваріантний аналіз часто виконується за допомогою моделювання.

Моделювання складається з етапів *формування моделі* (modeling) і *дослідження моделі* (рішення, simulation). А формування моделі включає дві процедури: по-перше, розробку моделей окремих компонентів, по-друге, формування моделі системи з моделей компонентів.

Перша із цих процедур виконується попередньо стосовно типових компонентів поза маршрутом проектування конкретних об'єктів. Як правило, моделі компонентів розробляються фахівцями в прикладних галузях, причому таких що знають вимоги до моделей і форм їхнього подання в САПР. Звичайно на допомогу розроблювачеві моделей у САПР пропонуються методики й допоміжні засоби, наприклад, у вигляді програм аналізу для експериментального відпрацьовування моделей. Створені моделі включаються в бібліотеки моделей прикладних програм аналізу.

На маршруті проектування кожного нового об'єкта виконується друга процедура (рисунок 1.1) — формування моделі системи з використанням бібліотечних моделей компонентів.

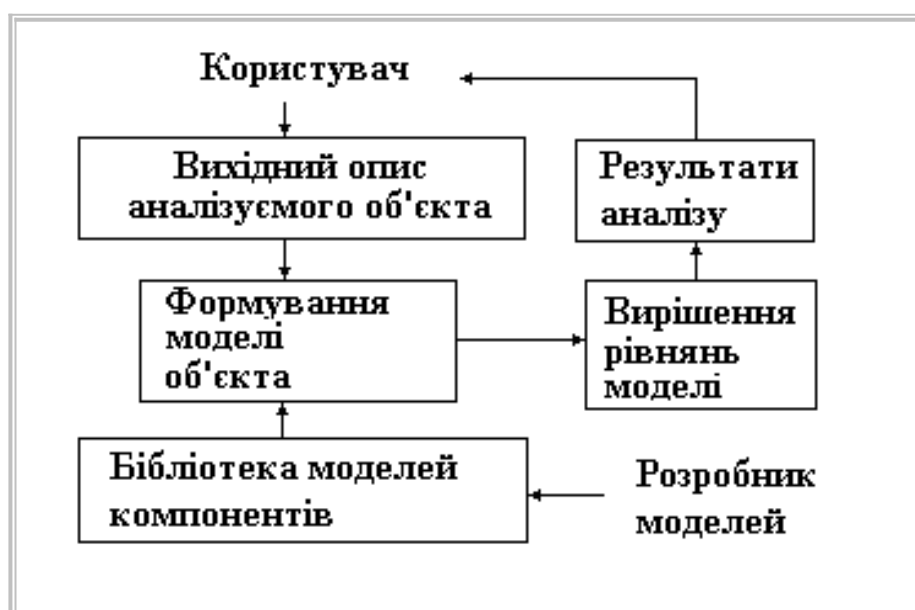


Рисунок 1.1 – Формування моделі системи

Ця процедура виконується автоматично за алгоритмами, включеними у заздалегідь розроблені програми аналізу. Приклади таких програм є в різних додатках і насамперед у галузях загального машинобудування й радіоелектроніки.

При застосуванні цих програм користувач описує досліджуваний об'єкт вхідною мовою програми аналізу не у вигляді системи рівнянь, що буде отримана автоматично, а у вигляді списку елементів структури, еквівалентної схеми, ескізу або креслення конструкції.

Друга процедура моделювання — *simulation* — зводиться до розв'язання рівнянь математичної моделі, наприклад, системи диференціальних рівнянь, і обчислення вектора вихідних параметрів Y . Якщо задані статистичні відомості про параметри X й потрібно одержати оцінки числових характеристик розподілів вихідних параметрів (наприклад, оцінки математичних очікувань і дисперсій), то це процедура *статистичного аналізу*. Якщо потрібно розрахувати матриці абсолютної A й (або) відносної B чутливості, то має місце задача *аналізу чутливості*.

Елемент $A_{j\bar{i}}$ матриці A називають абсолютним коефіцієнтом чутливості, він являє собою частинну похідну J -го вихідного параметра y_j за i -м параметром x_i . Інакше кажучи, $A_{j\bar{i}}$ є елементом вектора градієнта J -го вихідного параметра. На практиці зручніше використовувати безрозмірні відносні коефіцієнти чутливості $B_{j\bar{i}}$, що характеризують ступінь впливу змін параметрів елементів на зміни вихідних параметрів

$$B_{j\bar{i}} = \frac{A_{j\bar{i}} x_{i\text{ном}}}{y_{j\text{ном}}},$$

де $x_{i\text{ном}}$ й $y_{j\text{ном}}$ — номінальні значення параметрів x_i і y_j .

У процедурах *багатоваріантного аналізу* визначається вплив зовнішніх параметрів, розкиду й нестабільності параметрів елементів на вихідні параметри. Процедури статистичного аналізу й аналізу чутливості — характерні приклади процедур багатоваріантного аналізу.

Виконання аналізу й зіставлення отриманих результатів з бажаними значеннями називають процедурою *верифікації*.

1.7 Життєвий цикл виробів

Життєвий цикл промислових виробів (ЖЦВ) включає ряд етапів, починаючи від зародження ідеї нового продукту до його утилізації по закінченні строку використання. До них відносяться етапи проектування, технологічної підготовки виробництва (ТПВ), виробництва, реалізації продукції, експлуатації й, нарешті, утилізації (у число етапів життєвого циклу можуть також входити маркетинг, закупівля матеріалів і комплектуючих, надання послуг, упакування й зберігання, монтаж і введення в експлуатацію).

Розглянемо зміст основних етапів ЖЦВ для виробів машинобудування.

На етапі проектування виконуються проектні процедури – формування принципового рішення, розробка геометричних моделей і креслень, розрахунки, моделювання процесів, оптимізація й т.п.

На етапі підготовки виробництва розробляються маршрутна й операційна технології виготовлення деталей, реалізовані в програмах для верстатів ЧПК; технологія складання й монтажу виробів; технологія контролю й випробувань.

На етапі виробництва здійснюються: календарне й оперативне планування; придбання матеріалів і комплектуючих з їхнім вхідним контролем; механообробка й інших необхідні види обробки; контроль результатів обробки; складання; випробування й підсумковий контроль.

На поствиробничих етапах виконуються консервація, упакування, транспортування; монтаж у споживача; експлуатація, обслуговування, ремонт; утилізація.

На всіх етапах життєвого циклу є свої цільові настанови. При цьому учасники життєвого циклу прагнуть досягти поставлених цілей з максимальною ефективністю. На етапах проектування, ТПВ і виробництва потрібно забезпечити виконання вимог, поставлених до продукту, при заданому ступені надійності виробу й мінімізації матеріальних і тимчасових витрат, що необхідно для досягнення успіху в конкурентній боротьбі в умовах ринкової економіки. Поняття ефективності охоплює не тільки зниження собівартості продукції й скорочення

строків проектування й виробництва, але й забезпечення зручності освоєння й зниження витрат на майбутню експлуатацію виробів.

Досягнення поставлених цілей на сучасних підприємствах, що випускають складні технічні вироби, виявляється неможливим без широкого використання *автоматизованих систем* (АС), заснованих на застосуванні комп'ютерів і призначених для створення, переробки й використання всієї необхідної інформації про властивості виробів і супровідних процесів. Специфіка задач, розв'язуваних на різних етапах життєвого циклу виробів, обумовлює різноманітність застосовуваних АС.

1.8 Структура САПР

Як і будь-яка складна система, САПР складається з підсистем. Розрізняють підсистеми, що проектують і обслуговують.

Підсистеми, що проектують, безпосередньо виконують проектні процедури. Прикладами підсистем, що проектують, можуть служити підсистеми геометричного тривимірного моделювання механічних об'єктів, виготовлення конструкторської документації, схемотехнічного аналізу, трасування з'єднань у друкованих платах.

Обслуговуючі підсистеми забезпечують функціонування підсистем, що проектують, їхню сукупність часто називають системним середовищем (або оболонкою) САПР. Типовими обслуговуючими підсистемами є підсистеми керування проектними даними, підсистеми розробки й супроводу програмного забезпечення CASE (Computer Aided Software Engineering), підсистеми, що навчають, для освоєння користувачами технологій, реалізованих у САПР.

Структурування САПР за різними аспектами обумовлює поява *видів забезпечення САПР*. Прийнято виділяти сім видів забезпечення:

- *технічне забезпечення* (ТЗ), що включає різні апаратні засоби (ЕОМ, периферійні пристрої, мережне комутаційне устаткування, лінії зв'язку, вимірювальні засоби);

- *математичне забезпечення* (МЗ), що поєднує математичні методи, моделі й алгоритми для виконання проектування;

- *програмне забезпечення* (ПЗ), що представляє комп'ютерними програмами САПР;

- *інформаційне забезпечення* (ІЗ), що складається з баз даних (БД), систем керування базами даних (СКБД), а також включає інші дані, що використовуються при проектуванні;

- *лінгвістичне забезпечення* (ЛЗ), що виражається мовами спілкування між проектувальниками й ЕОМ, мовами програмування й мовами обміну даними між технічними засобами САПР;

- *методичне забезпечення* (МетЗ), що включає різні методики проектування, іноді до МетЗ відносять також математичне забезпечення;

- *організаційне забезпечення* (ОЗ), що представляється штатними розкладами, посадовими інструкціями й іншими документами, які регламентують роботу проектного підприємства.

Відзначимо, що вся сукупність використовуваних при проектуванні даних називається інформаційним фондом САПР. *Базою даних* називають упорядковану сукупність даних, що відображають властивості об'єктів і їхнього взаємозв'язку в деякій предметній галузі. Доступ до БД для читання, запису й модифікації даних здійснюється за допомогою СКБД, а сукупність БД і СКБД називають *банком даних* (Бнд).

Класифікацію САПР здійснюють за рядом ознак, наприклад, за застосуванням, цільовим призначенням, масштабами (комплексністю розв'язуваних задач), характером базової підсистеми – ядра САПР.

За застосуванню найбільш представницькими й широко використовуваними є такі групи САПР:

- а) САПР для застосування в галузях загального машинобудування. Їх часто називають машинобудівними САПР або MCAD (Mechanical CAD) системами;

- б) САПР в галузі радіоелектроніки: системи ECAD (Electronic CAD) або EDA (Electronic Design Automation);

- в) САПР в галузі архітектури й будівництва.

Крім того, відомо велике число спеціалізованих САПР, або виділюваних у зазначених групах, або що являють самостійну галузь у класифікації. Прикладами таких систем є САПР великих інтегральних схем (ВІС); САПР літальних апаратів; САПР електричних машин і т.п.

За цільовим призначенням розрізняють САПР або підсистеми САПР, що забезпечують різні аспекти проектування. Так, у складі МСАД з'являються САЕ/САД/САМ системи.

За масштабами розрізняють окремі *програмно-методичні комплекси* (ПМК) САПР, наприклад, комплекс аналізу міцності механічних виробів відповідно до методу скінченних елементів (МСЕ) або комплекс аналізу електронних схем; системи ПМК; системи з унікальними архітектурами не тільки програмного (software), але й технічного (hardware) забезпечень.

За характером базової підсистеми розрізняють такі різновиди САПР.

а) САПР на базі підсистеми машинної графіки й геометричного моделювання. Ці САПР орієнтовані на застосування, де основною процедурою проектування є конструювання, тобто визначення просторових форм і взаємного розташування об'єктів. Тому до цієї групи систем відноситься більшість САПР в галузі машинобудування, побудованих на базі графічних ядер. У цей час широко використовуються уніфіковані графічні ядра, застосовувані більш ніж в одній САПР, це ядра Parasolid фірми EDS Unigraphics і ACIS фірми Intergraph;

б) САПР на базі СКБД. Вони орієнтовані на застосування, у яких при порівняно нескладних математичних розрахунках переробляється великий об'єм даних. Такі САПР переважно зустрічаються в техніко-економічних галузях, наприклад, при проектуванні бізнес-планів, але мають місце також при проектуванні об'єктів, подібних до щитів керування в системах автоматизації;

в) САПР на базі конкретного прикладного пакета. Фактично це автономно використовувані програмно-методичні комплекси, наприклад, імітаційного моделювання виробничих процесів, розрахунку міцності за методом скінченних елементів, синтезу й аналізу систем автоматичного керування й т.п. Часто такі САПР відносять до систем САЕ. Прикладами можуть служити програми

логічного проектування на базі мови VHDL, математичні пакети типу MathCAD;

г) комплексні (інтегровані) САПР, що складаються із сукупності підсистем попередніх видів. Характерними прикладами комплексних САПР є CAE/CAD/CAM-системи в машинобудуванні або САПР БІС. Так, САПР БІС містить у собі СКБД і підсистеми проектування компонентів, принципів, логічних і функціональних схем, топології кристалів, тестів для перевірки придатності виробів. Для керування настільки складними системами застосовують спеціалізовані системні середовища.

1.9 Етапи проектування автоматизованих систем

Методики й засоби проектування автоматизованих систем (АС) і, зокрема, САПР підрозділяються на три великих рівні, що відповідають створенню інтегрованої корпоративної системи, окремих автоматизованих систем проектування й керування, програмно-методичних комплексів і компонентів автоматизованих систем.

На ринку програмного забезпечення пропонується ряд систем типу PLM, CPC і PDM. Існує ряд фірм, що спеціалізуються на розробці проектів системної інтеграції (наприклад, у Росії Jet Info, Consistent Software, Interface і ін.)

Для кожного класу АС (САПР, ERP, геоінформаційні системи й т.д.) можна вказати фірми, що спеціалізуються на розробці програмних (а іноді й програмно-апаратних) систем. Багато хто з них на основі однієї з базових технологій реалізують свій підхід до створення АС і дотримуються стратегії або тотального постачальника, або відкритості й розширення системи додатками й доповненнями третіх фірм.

Діє державний стандарт на стадії створення автоматизованих систем ДЕРЖСТАНДАРТ 34.601-90. Існує й міжнародний стандарт на стадії життєвого циклу програмної продукції (ISO 12207:1995). Як АС, так і компоненти АС є складними системами й при їхньому проектуванні потрібно використовувати один із стилів проектування:

- спадне проектування (Top-of-Design); чітка реалізація спадного проектування приводить до спіральної моделі розробки ПЗ, на кожному витку спіралі блоки попереднього рівня деталізуються, використовуються зворотні зв'язки (альтернативою є каскадна модель, що відноситься до почергової реалізації частин системи);

- висхідне проектування (Bottom-of-Design);
- еволюційне проектування (Middle-of-Design).

Найчастіше застосовують спадний стиль блочно-ієрархічного проектування.

Розглянемо етапи спадного проектування АС.

Верхній рівень проектування АС часто називають концептуальним проектуванням. Концептуальне проектування виконують у процесі передпроектних досліджень, формулювання ТЗ, розробки ескізного проекту й прототипування (відповідно ДО ДЕРЖСТАНДАРТУ, ці стадії називають формуванням вимог до АС, розробкою концепції АС і ескізним проектом).

Передпроектні дослідження проводять шляхом аналізу (обстеження) діяльності підприємства (компанії, установи, офісу), на якому створюється або модернізується АС. При цьому потрібно одержати відповіді на питання: що не влаштовує в існуючій технології? Що можна поліпшити? Кому це потрібно й, отже, який буде ефект? Перед обстеженням формуються та у процесі його проведення уточнюються цілі обстеження – визначення можливостей і ресурсів для підвищення ефективності функціонування підприємства на основі автоматизації процесів керування, проектування, документообігу й т.п. Зміст обстеження – виявлення структури підприємства, виконуваних функцій, інформаційних потоків, наявного досвіду й засобів автоматизації. Обстеження проводять системні аналітики (системні інтегратори) разом із представниками організації-замовника.

На основі аналізу результатів обстеження будують модель, що відбиває діяльність підприємства на даний момент (до реорганізації). Її називають *моделлю "As Is"*. Далі розробляють вихідну концепцію АС. Ця концепція містить у собі пропозиції щодо зміни структури підприємства, взаємодії підрозділів, інформаційних потоків, що виражається в *моделі "To Be"* (як повинно бути).

Результати аналізу конкретизуються в ТЗ на створення АС. У ТЗ вказують потоки вхідної інформації, типи вихідних документів і надаваних послуг, рівень захисту інформації, вимоги до продуктивності (пропускної спроможності) і т.п. ТЗ направляють замовникові для обговорення й остаточного узгодження.

Ескізний проект (технічну пропозицію) подають у вигляді проектної документації, що описує архітектуру системи, структуру її підсистем, склад модулів. Тут же вказуються пропозиції на вибір базових програмно-апаратних засобів, які повинні враховувати прогноз розвитку підприємства.

Відносно апаратних засобів і особливо ПЗ такий вибір найчастіше є вибір фірми-постачальника необхідних засобів (або, принаймні, базового ПЗ), тому що правильна спільна робота програм різних фірм досягається з великими складнощами. У проекті може бути запропоновано кілька варіантів вибору. При аналізі з'ясовуються можливості покриття функцій, які автоматизуються, наявними програмними продуктами й, отже, обсяги робіт з розробки оригінального ПЗ. Такий аналіз необхідний для попередньої оцінки тимчасових і матеріальних витрат на автоматизацію. Облік ресурсних обмежень дозволяє уточнити досяжні масштаби автоматизації, підрозділити проектування АС на роботи першої, другої і т.д. черги.

Після прийняття ескізного проекту розробляють прототип АС, що являє собою набір програм, які емулюють роботу готової системи. Завдяки прототипуванню можна не тільки розроблювачам, але й майбутнім користувачам АС побачити контури й особливості системи й, отже, завчасно внести корективи в проект.

Як на етапі обстеження, так і на наступних етапах доцільно дотримуватися певної дисципліни фіксації й подання одержуваних результатів, заснованої на тій або іншій методиці формалізації специфікацій. Формалізація потрібна для однозначного розуміння виконавцями й замовником вимог, обмежень і прийнятих рішень.

При концептуальному проектуванні застосовують ряд специфікацій, серед яких центральне місце займають моделі перетворення, зберігання й передачі інформації. Моделі,

отримані в процесі обстеження підприємства, є моделями його функціонування. У процесі розробки АС моделі, як правило, зазнають істотних змін (перехід від "As Is" до "To Be") і в остаточному вигляді модель "To Be" розглядають як модель проектованої АС.

Розрізняють функціональні, інформаційні, поведінкові й структурні моделі. *Функціональна модель* системи описує сукупність виконуваних системою функцій. *Інформаційна модель* відбиває структури даних – їхній склад і взаємозв'язки. *Поведінкова модель* описує інформаційні процеси (динаміку функціонування), у ній фігурують такі категорії, як стан системи, подія, перехід з одного стану в інший, умови переходу, послідовність подій, здійснюється прив'язка вчасно. *Структурна модель* характеризує морфологію системи (її побудову) — склад підсистем, їхнього взаємозв'язку.

Змістом наступних етапів спадного проектування (відповідно до ДЕРЖСТАНДАРТУ, це стадії розробки технічного проекту, робочої документації, запровадження в дію) є уточнення переліків устаткування, що здобуває, і готових програмних продуктів, побудова системного середовища, детальне інфологічне проектування БД і їхнього первісного наповнення, розробка власного оригінального ПЗ, що, у свою чергу, ділиться на ряд етапів спадного проектування. Ці роботи становлять зміст робочого проектування. Після цього виконуються закупівля й інсталяція програмно-апаратних засобів, впровадження й експериментальна експлуатація системи.

Особливе місце в ряді проектних задач займає розробка проекту корпоративної обчислювальної мережі, оскільки технічне забезпечення АС має мережну структуру.

Якщо територіально АС розташовується в одному будинку або в декількох близько розташованих будинках, то корпоративна мережа може бути виконана у вигляді сукупності декількох локальних підмереж, зв'язаних опорною локальною мережею. Крім вибору типів підмереж, зв'язних протоколів і комутаційного устаткування доводиться вирішувати задачі розподілу вузлів за підмережами, виділення серверів, вибору мережного ПЗ, визначення способу керування даними в обраній схемі розподілених обчислень і т.п.

Якщо АС розташовується у відокремлених один від одного пунктах, зокрема, розташованих у різних містах, то вирішується питання про оренду каналів зв'язку для корпоративної мережі, оскільки альтернативний варіант використання виділеного каналу в більшості випадків виявляється неприйнятним через високу ціну. Природно, що при цьому розглядається можливість використання послуг Internet. Проблеми, що виникають при цьому, пов'язані із забезпеченням інформаційної безпеки й надійності доставки повідомлень.

Питання, тести для самоконтролю

- 1 Дайте визначення поняття «проектування».
- 2 Що є предметом вивчення в теорії систем?
- 3 Назвіть ознаки, властиві складній системі.
- 4 Наведіть приклади ієрархічної структури технічних об'єктів, їх внутрішніх, зовнішніх і вихідних параметрів.
- 5 Опишіть структуру процесу проектування.
- 6 Опишіть типові маршрути і процеси проектування.
- 7 Поняття САПР. Склад і різновиди САПР.
- 8 Основні етапи проектування.
- 9 Основні відомості про подання та передачу інформації.
- 10 Схема обробки інформації при проектуванні ТП.
- 11 Забезпечення САПР. Його склад та вимоги.
- 12 Узагальнена модель САПР. Підсистеми САПР.
- 13 Чому проектування зазвичай має ітераційний характер?
- 14 Які причини призвели до появи та розвитку CALS-технології?
- 15 Наведіть приклади проектних процедур, що виконуються в системах CAE, CAD. CAM.

2 ТЕХНІЧНЕ ТА МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ САПР

2.1 Вимоги до технічного забезпечення САПР

Технічне забезпечення САПР містить у собі різні технічні засоби (hardware), використовувані для виконання автоматизованого проектування, а саме обчислювальні системи, ЕОМ (комп'ютери), периферійні пристрої, мережне устаткування, а також устаткування деяких допоміжних систем (наприклад, вимірювальних), що підтримують проектування. Відзначимо, що *обчислювальною системою* (на відміну від ЕОМ і обчислювальної мережі) називають сукупність апаратних і програмних засобів, спільно використовуваних при розв'язанні задач і розташовуваних компактно на території, розміри якої порівнянні з розмірами апаратних засобів.

Технічні засоби, що використовуються в САПР, повинні забезпечувати:

- виконання всіх необхідних проектних процедур, для яких є відповідне ПЗ;
- взаємодію між проектувальниками й ЕОМ, підтримку інтерактивного режиму роботи;
- взаємодію між членами колективу, що виконують роботу над загальним проектом.

Перша із цих вимог виконується при наявності в САПР обчислювальних машин і систем з достатніми продуктивністю і ємністю пам'яті.

Друга вимога ставиться до користувальницького інтерфейсу й виконується за рахунок включення в САПР зручних засобів вводу-виводу даних і, насамперед, пристроїв обміну графічною інформацією.

Третя вимога обумовлює об'єднання апаратних засобів САПР в обчислювальну мережу.

У результаті загальна структура ТЗ САПР являє собою мережу вузлів, зв'язаних між собою середовищем передачі даних. Вузлами (станціями даних) є робочі місця проектувальників, що часто мають назву *автоматизованих робочих місць* (АРМ) або робочих станцій (WS – Workstation), ними можуть бути також

великі ЕОМ (мейнфрейми), окремі периферійні й вимірювальні пристрої. Саме в АРМ повинні бути засоби для інтерфейсу проектувальника з ЕОМ. Що стосується обчислювальної потужності, то вона може бути розподілена між різними вузлами обчислювальної мережі.

Середовища передачі даних подані каналами передачі даних, що складаються з ліній зв'язку й комутаційного устаткування.

Призначення *процесора* – керування обчислювальним процесом і обробка даних відповідно до заданої програми. Процесори можуть бути універсальними, здатними вирішувати широке коло задач і спеціалізованими, орієнтованими на певне вузьке коло задач. Прикладом універсальних процесорів можуть служити процесори Pentium компанії Intel, а до спеціалізованих процесорів відносяться процесори графічні, сигнальні, вводу/виводу й ін.

Основна характеристика процесора – продуктивність або *швидкодія*. Використовують кілька різних показників продуктивності. Найбільш очевидним і історично першим показником продуктивності є швидкодія, вимірювана числом команд комп'ютера, виконуваних в одиницю часу, тобто вимірюване в Mips (1 Mips відповідає 10^6 команд у секунду). При розв'язанні науково-технічних задач характерне використання чисел із плаваючою точкою, у цьому випадку більш справедлива оцінка продуктивності в Mflops, тобто числом операцій над числами із плаваючою точкою в одиницю часу.

Пам'ять комп'ютера призначена для зберігання інформації й характеризується такими основними параметрами:

- об'ємом (ємністю), тобто максимально можливим числом блоків даних, розташовуваних у пам'яті;
- швидкодією, що характеризується середнім часом звертання до пам'яті (середнім часом пошуку, зчитування й/або запису одного блоку даних);
- ціною, віднесеною до одного блоку збережених даних.

Пам'ять комп'ютера має ієрархічну структуру у зв'язку з тим, що в єдиному запам'ятовувальному пристрої не вдається одночасно належною мірою задовольнити вимоги великого об'єму пам'яті й високої швидкодії. Тому звичайно до складу

процесора включають швидкодіючу *кеш-пам'ять* порівняно малого об'єму (часто також поділювану на два або три рівні), далі по мірі росту об'єму й часу звертання до пам'яті виділяють *оперативну пам'ять* і *зовнішню пам'ять*.

Фізично *кеш* і оперативна пам'ять у сучасних комп'ютерах, як правило, є напівпровідниковими. Для реалізації зовнішньої пам'яті використовують магнітні й оптичні принципи запису й читання інформації.

Залежно від складу виконуваних операцій розрізняють кілька типів пам'яті. *Пам'ять із довільним доступом* (операціями як читання, так і запису) звичайно позначають RAM (Random Access Memory), а *постійну пам'ять*, призначену тільки для читання, — ROM (Read Only Memory).

Як елементи оперативної пам'яті використовують осередки, що являють собою конденсатори. Заряджений конденсатор зберігає "1", розряджений — "0". Під час зчитування інформації конденсатори розряджаються. Крім того, заряд у конденсаторі через витік зберігається обмежений час (декілька мілісекунд). Тому необхідна підзарядка, що виконується в процесі регенерації інформації. Ця обставина обумовила назву пам'яті подібного типу — *динамічна пам'ять* або DRAM (Dynamic RAM). Малі розміри й простота елементів DRAM дозволяють одержати досить велику ємність пам'яті, але через витрати часу на регенерацію знижується швидкодія.

З метою підвищення швидкодії DRAM розроблено кілька модифікацій цього типу пам'яті. *Синхронна пам'ять* типу SDRAM (Synchronous DRAM) відрізняється від асинхронної пам'яті типу DRAM тим, що такти роботи пам'яті засинхронізовані з тактами роботи процесора. Це дозволяє виключити цикли очікування, що мають місце в DRAM.

У пам'яті типу DDR SDRAM (Double Data Rate SDRAM) швидкодію вдалося збільшити вдвічі в порівнянні з попередніми типами DRAM за рахунок того, що звертання до пам'яті відбуваються двічі за такт — як за переднім, так і за заднім фронтом тактових сигналів.

У більш швидкій *статичній пам'яті* SRAM (Static RAM) елементами пам'яті є бістабільні осередки, при зчитуванні інформація не губиться й регенерації не потрібно. Тому SRAM

більш швидкодіюча пам'ять і використовується як кеш-пам'ять. Однак SRAM значно дорожче DRAM.

Пам'ять ROM може бути однократно програмувальною, тобто дозволяти лише однократний первісний запис інформації, або бути *перепрограмувальною пам'яттю*. В останньому випадку розрізняють види пам'яті *EPROM* (Electrically Programmable ROM) і *EEPROM* (Electrically Erasable Programmable ROM), що називається також флеш-пам'яттю. Властивість репрограмованості заснована на використанні Моп-транзисторів із двома затворами – основним і схованим у шарі діелектрика. Подача імпульсу підвищеної напруги на Моп-транзистори в осередках, які повинні зберігати нулі, приводить до появи на схованому затворі електричного заряду, що перешкоджає утворенню провідного каналу, що й відповідає зберіганню нуля. Стан нуля зберігається практично необмежений час, але при перепрограмуванні заряд ліквідується за допомогою зовнішнього ультрафіолетового випромінювання (в EPROM) або електричного імпульсу, що подається на затвори (в EEPROM). Ультрафіолетове випромінювання приводить до поступової зміни властивостей напівпровідника, тому пам'ять EPROM допускає лише обмежене число перепрограмувань, крім того, перезапис відбувається з порівняно більшими витратами часу. Тому в цей час основна увага приділяється розвитку пам'яті типу EEPROM.

Прикладом використання ROM може служити пам'ять ROM BIOS (ROM Basic Input Output System) у ПК, що зберігає дані для керування стандартними зовнішніми пристроями.

Іноді говорять про пам'ять типу ECC. Це тип пам'яті, у якій реалізований режим ECC (Error Control Correction) контролю функціонування пам'яті з відновленням помилок. У цьому режимі виправляються одиночні помилки в групах з восьми байтів.

Зовнішня пам'ять, реалізована на магнітних носіях, подана *накопичувачами на магнітних дисках*.

Для підвищення швидкодії зовнішньої пам'яті й відмовостійкості застосовують підсистеми зовнішньої пам'яті з високою готовністю, що реалізують технологію надлишкових масивів дисків *RAID* (Redundant Arrays of Inexpensive Disks), об'єднаних в один блок із загальним інтелектуальним контролером. RAID є спосіб організації роботи декількох

фізичних дисків аналогічно одному логічному диску, при якому дисковий масив подається прикладній задачі як один диск. Як правило, масиви RAID використовуються в серверах для забезпечення надійності за рахунок дублювання даних.

Поширення набувають *рідкокристалічні дисплеї* (LCD – liquid cristal display). Екран LCD-дисплея являє собою простір між двома скляними пластинами, заповненими рідкими кристалами. Рідкі кристали здатні під дією електричного поля змінювати свою орієнтацію й тим самим змінювати свою здатність відбивати світло, створюючи необхідне зображення.

З'явилися перші зразки стереоскопічних моніторів, що дозволяють без додаткових стереоокулярів спостерігати тривимірні зображення.

Периферійні пристрої комп'ютерів ділять на пристрої вводу, пристрою виводу й зовнішні запам'ятовувальні пристрої. До *пристроїв вводу* відносяться клавіатура, миша, сканер, дигітайзер. *Пристрої виводу* подані принтерами, плотерами (графобудівниками), дисплеями.

Звичайно в автоматизованих системах користувачі працюють на *автоматизованих робочих місцях* (АРМ), на яких є комп'ютер з виходом у корпоративну мережу й необхідні периферійні пристрої. Типовий склад пристроїв АРМ: ПК або робоча станція з вінчестером; пристрої вводу-виводу, що включають, як мінімум, клавіатуру, мишу, дисплей; додатково до складу АРМ можуть входити принтер, сканер, плотер і деякі інші периферійні пристрої.

АРМ розрізняються своєю орієнтацією на певні групи проектних або бізнес-процедур і відповідно типами й характеристиками обчислювального та периферійного устаткування. Для уведення інформації в ЕОМ використовують дигітайзери, інтерактивні дошки, сканери, візуалізатори, цифрові відеокамери (Web-камери).

Дигітайзери призначені для ручного уведення графічної інформації, у цей час їх застосовують досить рідко. Дигітайзер має вигляд кульмана, по його електронній дошці переміщається курсор, на якому розташований візир і кнопкова панель. Курсор має електромагнітний зв'язок із сіткою провідників в електронній дошці. При натисканні кнопки в деякій позиції курсора

відбувається занесення інформації про координати цієї позиції. У такий спосіб може здійснюватися ручне введення креслень.

Інтерактивна дошка (smart board), називана також інтелектуальною дошкою, дозволяє користувачам виводити на поверхню дошки вміст файлів з пам'яті комп'ютера, Internet-джерел, CD-ROM, редагувати виведене зображення, наносити поверх нього свої графічні й текстові фрагменти. Усе, що користувач намалював або написав на дошці, буде збережено у вигляді комп'ютерних файлів, може бути роздруковано, послано електронною поштою, поміщено в Web-сторінку.

Інтерактивні дошки є зручним засобом подання, редагування й збереження навчальних матеріалів, формованих викладачем безпосередньо під час проведення навчальних занять. Крім того, за допомогою інтерактивної дошки викладач може управляти комп'ютером, вибираючи потрібні програми або файли простою вказівкою на екранну кнопку на поверхню дошки.

Для автоматичного уведення інформації з наявних текстових або графічних документів використовують *сканери* планшетного або протяжливого типу й візуалізатори. Спосіб зчитування – оптичний. У скануючий голівці сканера розміщуються оптоволоконні самофокусуючі лінзи й фотоелементи. Розгортка у різних моделях становить від 300 до 2400 точок на дюйм (цей параметр часто позначають dpi). Зчитана інформація має растрову форму, програмне забезпечення сканера подає її в одному зі стандартних форматів, наприклад tiff, gif, psx, jpeg, і для подальшої обробки може бути виконана *векторизація* – переклад графічної інформації у векторну форму, наприклад, у формат dxf.

Візуалізатор служить для зчитування інформації з документа і її передачі на вхід комп'ютера для збереження в пам'яті (режим сканера) або на вхід мультимедійного проектора для збільшеного зображення на екрані під час презентацій або навчальних занять. Як документ можуть використовуватися не тільки плоскі сторінки з текстом, малюнками, кресленнями, але й інші предмети, наприклад, деталі механічних або електронних пристроїв порівняно невеликих розмірів. Можна накладати зображення з візуалізатора на інші зображення, що надходять із комп'ютера.

Для зйомки відеофрагментів з метою їхнього наступного включення в мультимедійні навчальні матеріали використовують цифрові *відеокамери*.

Для виведення інформації у вигляді твердих копій застосовують *принтери* й *плотери* (графобудівники). Перші з них орієнтовані на одержання документів малого формату (А3, А4), другі – для виведення графічної інформації на широкоформатні носії.

Дигітайзери, сканери, принтери, плотери можуть входити до складу АРМ або розділятися користувачами декількох робочих станцій у складі локальної обчислювальної мережі.

Виведення інформації на екран збільшених розмірів, що потрібно під час презентацій і навчальних занять, виконується за допомогою мультимедійних проекторів. Вони відрізняються від відеопроєкторів тим, що сприймають не відеосигнали, а цифрові сигнали від комп'ютера.

Зв'язок різних пристроїв комп'ютера між собою здійснюється за допомогою *шин*. До шин ставляться вимоги високої швидкодії (пропускної спроможності), високій надійності й малої вартості. Пропускна спроможність шини визначається обсягом інформації, переданої по шині в одиницю часу, і дорівнює добутку тактової частоти шини на її розрядність.

У найпростішому випадку комп'ютер може мати єдину шину, що має назву *системної*. Однак застосування єдиної шини не дозволяє запаралелити операції обміну, крім того, для ряду пристроїв виявляються не погодженими параметри швидкодії пристрою й шини. Тому звичайно використовують ієрархію шин. У сучасних ПК є *системна шина*, що зв'язує процесор з оперативною пам'яттю (і можливо з кеш-пам'яттю другого рівня), *шина розширення* для підключення мережного контролера й швидких зовнішніх пристроїв і шина повільних пристроїв, таких як клавіатура, миша, принтер і т.п. Часто окрема шина, більш швидкодіюча, ніж шина розширення, зв'язує відеоадаптер з оперативною пам'яттю. Шини можуть бути декількох профілів залежно від кількості провідних пристроїв, тобто пристроїв, які можуть ініціювати транзакції зчитування або запису. При декількох провідних пристроях надання повноважень на захоплення шини здійснює схема арбітражу.

Розрізняють синхронні й асинхронні шини. У синхронних шинах передбачається, що тактові сигнали зв'язуючих пристроїв засинхронізовані, тому можлива більша швидкодія, але при істотному обмеженні довжини шини. В асинхронних шинах використовується старт-стопний режим передачі з попереднім запитом зв'язку й згодою на нього.

Комп'ютери (ЕОМ) і обчислювальні системи (ОС) прийнято класифікувати за рядом ознак.

Залежно від продуктивності й вартості обчислювального встаткування виділяють кілька типів ЕОМ і ОС, причому різні покоління обчислювальної техніки мали свою шкалу типів. ЕОМ і ОС підрозділяють на персональні комп'ютери, робочі станції, сервери, мейнфрейми, кластери й суперкомп'ютери.

Найбільше поширення в САПР одержали персональні комп'ютери (ПК), робочі станції й сервери.

Персональні комп'ютери з'явилися на початку 80-х років минулого століття в результаті трансформації міні-ЕОМ у порівняно дешеві настільні системи для індивідуального використання, завдяки розвитку елементної бази обчислювальної техніки, що привело до створення великих (ВІС) і надвеликих (НВІС) інтегральних схем.

Робочі станції являють собою обчислювальні системи, орієнтовані на розв'язання деяких задач, насамперед в автоматизованих системах проектування (САПР). Комп'ютери в САПР орієнтовані на розв'язання складних проектних задач, що обумовлює підвищені швидкодію й об'єм пам'яті, а також розширені можливості обробки й візуалізації графічної інформації робочих станцій у порівнянні з персональними комп'ютерами. Тому звичайно робочі станції мають більш складну структуру й більш дорогі пристрої, ніж ПК.

В обчислювальних мережах виконання функцій, пов'язаних з обслуговуванням всіх вузлів мережі, покладається на *сервери*. За функціональним призначенням розрізняють сервери файлові, баз даних, комутаційні, прикладні, поштові й т.п. Сервери, як правило, повинні мати більшу швидкодію, надійність й у багатьох випадках збільшену ємність пам'яті в порівнянні з комп'ютерами в клієнтських вузлах.

Мейнфреймами називають великі ЕОМ. Висока продуктивність і велика ємність пам'яті забезпечують розв'язання складних проблем, дозволяють використовувати такі комп'ютери як центральний вузол ОС, що управляє роботою багатьох простих терміналів. Однак за співвідношенням продуктивність/ціна мейнфрейми звичайно помітно поступаються попереднім типам ОС, їхнє використання в цей час досить обмежене.

Кластер – це розподілена система комп'ютерів, що функціонує як єдина система із загальними ресурсами. Основна мета, що обумовила появу кластерів, – збереження працездатності ОС шляхом перерозподілу навантаження при виході з ладу частини ресурсів. Крім того, кластеризація – один із шляхів підвищення продуктивності за рахунок спільного використання багатьох комп'ютерів. Кластери дозволяють нарощувати обчислювальну потужність, оскільки легко масштабуються.

Комп'ютери, що характеризуються найбільшими значеннями продуктивності й ціни серед інших типів ЕОМ і ОС, відносять до категорії *суперкомп'ютерів*. Устаткування цього суперкомп'ютера займає площу, рівну площі двох баскетбольних площадок. Це унікальні обчислювальні системи, що найчастіше використовуються в наукових і освітніх установах для розв'язання складних задач при наукових дослідженнях.

Популярна класифікація ОС, заснована на архітектурних особливостях систем і запропонована Флином. Відповідно до неї виділяють чотири класи систем: *SISD* (Single Instruction Single Data — один потік команд і один потік даних), *SIMD* (Single Instruction Multi Data — один потік команд і багато потоків даних), *MISD* (Multi Instruction Single Data — багато потоків команд і один потік даних), *MIMD* (Multi Instruction Multi Data – багато потоків команд і багато потоків даних).

Класу *SISD* відповідають однопроцесорні ЕОМ без застосування розпаралелювання обчислювального процесу. Клас *MISD* не знайшов практичного застосування. У сучасних високопродуктивних ОС застосовують класи *SIMD* і *MIMD*.

В *SIMD* системах та сама машинна команда виконується одночасно над різними числами, що характерно, наприклад, для

матрично-векторних обчислень. Зокрема, до цього класу належать однопроцесорні векторні й векторно-конвейерні ОС, а також ОС на основі матриць процесорів.

Більшість сучасних суперкомп'ютерів має структуру MIMD. У цьому класі розрізняють кілька підкласів.

По-перше, це системи *SM* (Shared Memory), що розділяють загальну пам'ять, тобто що мають загальний адресний простір. До цього підкласу відносяться багатопроцесорні векторні ОС і системи *SMP* (Symmetric Multi-Processor). Очевидно, що в таких системах жорсткі вимоги ставляться до швидкодії каналів зв'язку процесорів з загальною пам'яттю. У векторних ОС використовується високошвидкісна шина між оперативною пам'яттю й векторними регістрами, у системах *SMP* висока продуктивність підтримується за рахунок введення кеш-пам'яті великого об'єму, тобто вимоги до пропускної спроможності тракту "оперативна пам'ять - кеш-пам'ять" істотно знижуються.

По-друге, системи с *розподіленою пам'яттю DM* (Distributed Memory) – це системи, у яких кожний процесор має свою оперативну пам'ять. Прикладом систем *DM* можуть служити кластерні системи. У свою чергу, кожний вузол може бути системою типу *SIMD* або *MIMD*. Тоді говорять про системи *DM-SIMD* або *DM-MIMD* відповідно. Якщо операційна система працює з єдиним простором адрес, а кожний процесор виконаний як система *MIMD*, то маємо систему *SM-MIMD*.

По-третє, системи з *масовим паралелізмом MPP* (Massively Parallel Processing), у яких кілька копій однієї програми паралельно виконуються в різних вузлах з різними даними. Відмітна ознака *MPP* – помітно більше число процесорів, ніж у системах *SMP*, що ще більше ускладнює проблему ефективного розпаралелювання. Для користувача системи *MPP* можуть виглядати як *SM-MIMD*.

2.2 Вимоги до математичних моделей і методів у САПР

Основними вимогами до математичних моделей є вимоги адекватності, точності, економічності.

Модель завжди лише приблизно відбиває деякі властивості об'єкта. Адекватність має місце, якщо модель відбиває задані

властивості об'єкта із прийнятною точністю. Під точністю розуміють ступінь відповідності оцінок однойменних властивостей об'єкта й моделі.

Економічність (обчислювальна ефективність) визначається витратами ресурсів, необхідних для реалізації моделі. Оскільки в САПР використовуються математичні моделі, далі мова йтиме про характеристики саме математичних моделей, і економічність буде характеризуватися витратами машинних часу й пам'яті.

Адекватність оцінюється переліком відбиваних властивостей і областями адекватності. Область адекватності – область у просторі параметрів, у межах якої погрішності моделі залишаються в припустимих межах.

Вихідний математичний опис процесів в об'єктах на макрорівні подано системами звичайних диференціальних і алгебраїчних рівнянь. Аналітичні рішення таких систем при типових значеннях їхніх порядків у практичних задачах одержати не вдається, тому в САПР переважно використовуються алгоритмічні моделі. У цьому параграфі викладений узагальнений підхід до формування алгоритмічних моделей на макрорівні, справедливий для більшості застосувань.

Вихідними для формування математичних моделей об'єктів на макрорівні є компонентні й топологічні рівняння.

Компонентними рівняннями називають рівняння, що описують властивості елементів (компонентів), інакше кажучи, математична модель елемента (ММЕ) подається компонентними рівняннями.

Топологічні рівняння описують взаємозв'язки в складі системи, що моделюється.

У сукупності компонентні й топологічні рівняння конкретної фізичної системи являють собою вихідну математичну модель системи (ММС).

Очевидно, що компонентні й топологічні рівняння в системах різної фізичної природи відбивають різні фізичні властивості, але можуть мати однаковий формальний вигляд. Однакова форма запису математичних співвідношень дозволяє говорити про формальні аналогії компонентних і топологічних рівнянь. Такі аналогії існують для механічних поступальних, механічних обертальних, електричних, гідравлічних

(пневматичних), теплових об'єктів. Наявність аналогій приводить до практично важливого висновку: значна частина алгоритмів формування й дослідження моделей у САПР виявляється інваріантною й може бути застосована до аналізу проєктованих об'єктів у різних предметних областях. Єдність математичного апарата формування ММС особливо зручна при аналізі систем, що складаються з фізично різнорідних підсистем.

Розрізняють фазові змінні двох типів, їхні узагальнені найменування – фазові змінні типу потенціалу (наприклад, електрична напруга) і фазові змінні типу потоку (наприклад, електричний струм). Кожне компонентне рівняння характеризує зв'язки між різнотипними фазовими змінними, стосовними до одного компонента (наприклад, закон Ома описує зв'язок між напругою й струмом у резисторі), а топологічне рівняння – зв'язок між однотипними фазовими змінними в різних компонентах.

Моделі можна подавати у вигляді систем рівнянь або в графічній формі, якщо між цими формами встановлена взаємно однозначна відповідність. Як графічна форма часто використовують еквівалентні схеми.

Нижче розглянемо приклади *компонентних* і *топологічних* рівнянь для різних типів систем.

Аналіз процесів у проєктованих об'єктах можна робити в часовій і частотній областях. *Аналіз у часовій області* (динамічний аналіз) дозволяє одержати картину перехідних процесів, оцінити динамічні властивості об'єкта, він є важливою процедурою при дослідженні як лінійних, так і нелінійних систем.

Методи аналізу в часовій області, використовувані в універсальних програмах аналізу в САПР, – це чисельні методи інтегрування систем звичайних диференціальних рівнянь (СЗДР)

$$\mathbf{F}\left(\frac{d\mathbf{V}}{dt}, \mathbf{V}, t\right) = 0.$$

Інакше кажучи, це методи алгебраїзації диференціальних рівнянь. Формули інтегрування СЗДР можуть входити в математичну модель (ММ) незалежно від компонентних рівнянь або бути інтегрованими в ММ компонентів, як це виконано у вузловому методі.

Від вибору методу розв'язання СЗДР істотно залежать такі характеристики аналізу, як точність і обчислювальна ефективність. Ці характеристики визначаються насамперед типом і порядком обраного методу інтегрування СЗДР.

Застосовують два типи методів інтегрування — *явні методи* (інакше екстраполяційні або методи, засновані на формулах інтегрування вперед), і *неявні методи* (інтерполяційні, засновані на формулах інтегрування назад). Розходження між ними зручно показати на прикладі найпростіших методів першого порядку — *методів Ейлера*.

Формула явного методу Ейлера являє собою таку формулу заміни похідних у точці t_n

$$\left. \frac{d\mathbf{V}}{dt} \right|_n = \frac{\mathbf{V}_{n+1} - \mathbf{V}_n}{h_n}.$$

Тут індекс дорівнює номеру кроку інтегрування; $h_n = t_{n+1} - t_n$ — розмір кроку інтегрування (звичайно h_n називають просто кроком інтегрування). У формулі неявного методу Ейлера використане диференціювання назад:

$$\left. \frac{d\mathbf{V}}{dt} \right|_n = \frac{\mathbf{V}_n - \mathbf{V}_{n-1}}{h_n},$$

де $h_n = t_n - t_{n-1}$.

Метод інтегрування СЗДР називають *A-стійким*, якщо погрішність інтегрування залишається обмеженою при будь-якому кроці $h > 0$.

Застосування *A-стійких методів* дозволяє істотно зменшити необхідні числа кроків \mathbb{I} . У цих методах крок вибирається автоматично не з умов стійкості, а тільки з міркувань точності рішення.

Вибір порядку методу розв'язання СЗДР досить простий: по-перше, більш високий порядок забезпечує більш високу точність, по-друге, серед неявних різницевих методів, крім методу Ейлера, *A-стійкі* також методи другого порядку й серед них — метод трапецій. Тому переважне поширення в програмах аналізу одержали методи другого порядку — модифікації методу трапецій.

Одна із вдалих реалізацій неявного методу другого порядку, яку можна вважати модифікацією *методу трапецій*, заснована на комбінованому використанні явної й неявної формул Ейлера. Розглянемо питання, чому таке комбінування знижує погрішність і приводить до підвищення порядку методу.

Попередньо відзначимо, що в методах P -го порядку локальна погрішність, тобто погрішність, допущена на одному n -му кроці інтегрування, оцінюється старшим із членів, що відкидають у розкладанні $\mathbf{V}(t)$ рішення в ряд Тейлора

$$\delta = c |\mathbf{V}^{(p+1)}(\tau)| h^{p+1},$$

де c – постійний коефіцієнт, що залежить від методу, $|\mathbf{V}^{(p+1)}(\tau)|$ норма $(p+1)$ -х похідних вектора $\mathbf{V}(t)$, що оцінюється за допомогою кінцево-різницевої апроксимації, τ – значення часу t усередині кроку.

Якщо n -й крок інтегрування в *комбінованому методі* був неявним, тобто виконаним за неявною формулою, то наступний крок з тим же значенням h повинен бути явним.

Неявні методи й, зокрема, розглянутий комбінований метод доцільно використовувати тільки при змінній величині кроку. Дійсно, при помітних швидкостях зміни фазових змінних погрішність залишається в припустимих межах тільки при малих кроках, у квазистатичних режимах крок може бути в багато разів більше.

Алгоритми *автоматичного вибору кроку* засновані на порівнянні допущених і припустимої локальних погрішностей. Наприклад, уводиться деякий діапазон (коридор) погрішностей δ , у межах якого крок зберігається незмінним. Якщо ж допущена погрішність перевищує верхню границю діапазону, то крок зменшується, якщо ж виходить за нижню границю, то крок збільшується.

У програмах аналізу в САПР для розв'язання СЛАР найчастіше застосовують метод Гаусса або його різновиду. *Метод Гаусса* – метод послідовного виключення невідомих із системи рівнянь.

Можна помітно підвищити обчислювальну ефективність аналізу, якщо використати характерну практично для всіх застосувань властивість високої *розрідженості матриці* \mathbf{A} в моделі.

При використанні *методів розріджених матриць* потрібно враховувати залежність обчислювальної ефективності від того, як подана матриця коефіцієнтів \mathbf{A} , точніше від того, у якому порядку записані її рядки й стовпці.

Методом розріджених матриць називають метод розв'язання СЛАР на основі методу Гаусса з урахуванням розрідженості (первинної й вторинної) матриці коефіцієнтів.

Метод розріджених матриць можна реалізувати шляхом інтерпретації й компіляції. В обох випадках створюються масиви ненульових коефіцієнтів матриці (з урахуванням вторинних нулів) і масиви координат цих ненульових елементів.

Обчислення при розв'язанні СНАР складаються з декількох вкладених один в інший циклічних процесів. Зовнішній цикл – цикл покрокового чисельного інтегрування, параметром циклу є номер кроку. Якщо модель аналізованого об'єкта нелінійна, то на кожному кроці виконується проміжний цикл – ітераційний цикл розв'язання системи нелінійних алгебраїчних рівнянь (СНАР).

Аналіз у частотній області більш специфічний у порівнянні з аналізом у часовій області. Його застосовують, як правило, до об'єктів з ММ, які лінеаризуються, при дослідженні коливальних стаціонарних процесів, аналізі стійкості, розрахунку перекручувань інформації, що подається спектральними складових сигналів, і т.п.

Аналіз у частотній області виконується відповідно до лінеаризованих моделей об'єктів. Для лінійних СЛАР справедливе застосування для алгебраїзації диференціальних рівнянь перетворення Фур'є, у якому оператор d/dt замінюється на оператор $j\omega$.

Характерною особливістю є комплексний характер матриці коефіцієнтів, що до деякої міри ускладнює процедуру розв'язання, але не створює принципових труднощів. При розв'язанні задають ряд частот ω_k . Для кожної частоти вирішують СЛАР й визначають дійсні й уявні частини шуканих фазових змінних. За ними визначають амплітуду й фазовий кут

кожної спектральної складової, що й дозволяє побудувати амплітудно-частотні характеристики, знайти власні частоти коливальної системи й т.п.

Одноваріантний аналіз дозволяє одержати інформацію про стан і поведіння проєктованого об'єкта в одній точці простору внутрішніх X і зовнішніх Q параметрів. Очевидно, що для оцінки властивостей проєктованого об'єкта цього недостатньо. Потрібно виконувати багатоваріантний аналіз, тобто досліджувати поведінку об'єкта у ряді точок згаданого простору, що для стислості будемо далі називати простором аргументів.

Найчастіше *багатоваріантний аналіз* у САПР виконується в інтерактивному режимі, коли розроблювач неодноразово міняє в математичній моделі ті або інші параметри із множин X і Q , виконує одноваріантний аналіз і фіксує отримані значення вихідних параметрів. Подібний багатоваріантний аналіз дозволяє оцінити *області працездатності*, ступінь виконання умов працездатності, а отже, ступінь виконання ТЗ на проєктування, розумність прийнятих проміжних рішень щодо зміни проєкту тощо.

Областю працездатності називають область у просторі аргументів, у межах якої виконуються всі задані умови працездатності, тобто значення всіх вихідних параметрів перебувають у припустимих за ТЗ межах.

Серед процедур багатоваріантного аналізу можна виділити типові, виконувані за заздальгідь складеними програмами. До таких процедур відносяться аналіз чутливості й статистичний аналіз.

Найбільш просто *аналіз чутливості* реалізується шляхом чисельного диференціювання. Такий метод чисельного диференціювання називають *методом приростів*. Для аналізу чутливості, відповідно до методу приростів, потрібно виконати $N+1$ раз одноваріантний аналіз. Результат його застосування — матриці абсолютної й відносної чутливості, елементами яких є коефіцієнти $A_{j\bar{j}}$ й $B_{j\bar{j}}$.

Аналіз чутливості — це розрахунок векторів градієнтів вихідних параметрів, що входить складовою частиною в програми параметричної оптимізації, що використовують градієнтні методи.

Мета *статистичного аналізу* – оцінка законів розподілу вихідних параметрів i (або) числових характеристик цих розподілів. Випадковий характер величин y_j обумовлений випадковим характером параметрів елементів x_i , тому вихідними даними для статистичного аналізу є відомості про закони розподілу x_i . Відповідно до результатів статистичного аналізу прогнозують такий важливий виробничий показник, як відсоток бракованих виробів у готовій продукції.

У САПР статистичний аналіз здійснюється чисельним методом – *методом Монте-Карло* (статистичних випробувань). Відповідно до цього методу виконуються N статистичних випробувань, кожне статистичне випробування являє собою одноваріантний аналіз, виконуваний при випадкових значеннях параметрів-аргументів. Ці випадкові значення вибирають відповідно до заданих законів розподілу аргументів x_i . Отримані в кожному випробуванні значення вихідних параметрів накопичують, після N випробувань обробляють, що дає такі результати: гістограми вихідних параметрів; оцінки математичних очікувань і дисперсій вихідних параметрів; оцінки коефіцієнтів кореляції й регресії між обраними вихідними й внутрішніми параметрами, які, зокрема, можна використати для оцінки коефіцієнтів чутливості.

Статистичний аналіз, виконуваний відповідно до методу Монте-Карло – трудомістка процедура, оскільки число випробувань N доводиться вибирати досить великим, щоб досягти прийнятної точності аналізу. Інша причина, що ускладнює застосування методу Монте-Карло, — труднощі в одержанні достовірної вихідної інформації про закони розподілу аргументів-параметрів x_i .

Більш типова ситуація, коли закони розподіли x_i невідомі, але з великою часткою впевненості можна вказати гранично припустимі відхилення Δx_i параметрів x_i від номінальних значень $x_{j\text{ном}}$ (такі відхилення часто вказуються в паспортних даних на комплектуючі деталі). У таких випадках більш реалістично застосовувати метод *аналізу на найгірший випадок*. Відповідно до цього методу, спочатку виконують аналіз чутливості з метою визначення знаків коефіцієнтів чутливості.

Далі здійснюють m раз одноваріантний аналіз, де m – число вихідних параметрів. У кожному варіанті задають значення аргументів, найбільш несприятливі для виконання умов працездатності чергового вихідного параметра $y_j, j \in [1:m]$.

Однак варто помітити, що, проводячи аналіз на найгірший випадок, можна одержати завищені значення розкиду вихідних параметрів, і якщо домагатися виконання умов працездатності в найгірших випадках, то це часто веде до невиправданого збільшення вартості, габаритних розмірів, маси й інших показників проєктованих конструкцій, хоча й гарантує із запасом виконання умов працездатності.

2.3 Математичні моделі та методи аналізу на мікрорівні

Математичними моделями на мікрорівні є диференціальні рівняння в частинних похідних або інтегральні рівняння, що описують поля фізичних величин. Інакше кажучи, на мікрорівні використовуються *моделі з розподіленими параметрами*. Як незалежні змінні у моделях можуть фігурувати просторові змінні x_1, x_2, x_3 й час t .

Характерними прикладами моделей можуть служити рівняння математичної фізики разом із заданими крайовими умовами.

У САПР розв'язання диференціальних або інтегро-диференціальних рівнянь із частковими похідними виконується чисельними методами. Ці методи засновані на дискретизації незалежних змінних – їхньому поданні кінцевою множиною значень в обраних вузлових точках досліджуваного простору. Ці точки розглядаються як вузли деякої сітки, тому використовувані в САПР методи – це сіткові методи.

Серед сіткових методів найбільше поширення одержали два методи: метод скінченних різностей (МСР) і метод скінченних елементів (МСЕ). Звичайно виконують дискретизацію просторових незалежних змінних, тобто використовують просторову сітку. У цьому випадку результатом дискретизації є система звичайних диференціальних рівнянь для нестационарної задачі або система алгебраїчних рівнянь для стаціонарної.

У методі скінченних різностей алгебраїзація похідних за просторовими координатами базується на апроксимації похідних скінченнорізницевиими виразами. При використанні методу потрібно вибрати кроки сітки по кожній координаті й вид шаблону. Під шаблоном розуміють множину вузлових точок, значення змінних у яких використовуються для апроксимації похідної в одній конкретній точці.

Метод скінченних елементів заснований на апроксимації не похідних, а самого рішення $V(\mathbf{z})$. Але оскільки воно невідомо, то апроксимація виконується виразами з невизначеними коефіцієнтами φ_i .

Найбільше поширення МСЕ одержав у САПР машинобудування для аналізу міцності об'єктів. Для цієї задачі можна використувати розглянутий підхід, тобто виконати алгебраїзацію вихідного рівняння пружності (рівняння Ламе). Однак більш зручним у реалізації МСЕ виявився підхід, заснований на варіаційних принципах механіки.

Питання, тести для самоконтролю

- 1 Технічні засоби САПР. Склад і структура засобів.
- 2 Способи подання інформації в пам'яті ЕОМ.
- 3 Основні види технологічної інформації. Форми подання технологічної інформації.
- 4 Перелічіть формальні мови САПР. Вимоги до формальних мов. Наведіть приклади формальних мов.
- 5 Що таке метамови і яке їх призначення. Наведіть приклади природних і штучних мов і їх призначення.
- 6 Загальні принципи організації інтерфейсу в САПР.
- 7 Математичне моделювання – основа побудови САПР.
- 8 Поняття моделі і математичної моделі.
- 9 Автоматизація логічного виведення – основа розв'язання завдань в САПР.
- 10 Поясніть склад і призначення облаштувань графічної робочої станції.
- 11 Що таке "промисловий комп'ютер"? Які його особливості?

12 Дайте порівняльну характеристику методів комутації каналів і комутації пакетів.

13 У чому полягає суть методів тимчасового (TDM) і частотного (FDM) розділення каналів?

14 Чому в МДКН/ОК повторні спроби захоплення лінії дозволяються через випадкові інтервали часу?

15 Чому в мережах Ethernet введено обмеження на розмір кадру знизу? Розрахуйте нижню межу довжини кадру для Gigabit Ethernet.

16 У чому полягають переваги переведення системи стільникового зв'язку у більш високочастотний діапазон?

17 Назвіть чинники, що обумовлюють високі швидкості передачі даних в мережах АТМ.

18 Що таке "маршрутизація від джерела"?

19 Що розуміють під віртуальною ЛОМ?

20 Математичні моделі структур технологічних процесів.

21 Основні принципи об'ємного моделювання в САПР.

22 Загальна характеристика технологічних процесів з точки зору автоматизації їх проектування.

23 Системи штучного інтелекту та можливі шляхи їх практичного використання в САПР.

24 Експертні системи та можливі шляхи їх практичного використання в САПР.

25 Оптимізація структури технологічного процесу з використанням методу Монте-Карло.

26 Математичні моделі деталей і заготовок.

27 Основні етапи автоматизованого проектування.

28 Основні методичні підходи до розв'язання задач проектування.

29 Загальні принципи побудови і розв'язання задач математичної оптимізації технологічних процесів на ЕОМ.

30 Дайте визначення області адекватності математичної моделі.

31 Що розуміють під постійною часу фізичної системи?

32 На чому ґрунтуються алгоритми автоматичного вибору кроку інтегрування при розв'язанні систем диференціальних рівнянь?

33 Що розуміють під областю працездатності?

34 Побудуйте таблиці логічних функції І і АБО для п'ятизначного алфавіту.

35 Поясніть сутність подієвого методу моделювання.

36 Що таке «параметрична модель»? «Асоціативне моделювання».

37 Дайте формулювання задачі математичного програмування.

38 У чому полягають труднощі розв'язання багатокритеріальних задач оптимізації?

39 Чи можна застосовувати метод проекції градієнта для розв'язання завдань оптимізації з обмеженнями типу нерівностей?

40 Які властивості характеризують клас NP-повних задач?

Індивідуальні практичні завдання

1 Розрахуйте розмір вікна зіткнень в мережі 10Base-5, якщо лінія передачі даних подана одним сегментом кабелю довжиною 500 м.

2 Якою може бути максимальна інформаційна швидкість у каналі передачі даних зі смугою пропускання 4 кГц і відношенням сигнал перешкода 130?

3 Розрахуйте затримку в передачі сигналу у супутникових системах з використанням геостаціонарних орбіт (висота супутника 36 тис. км).

4 Виконайте кілька кроків інтегрування для диференціального рівняння $dx/dt=10-2x$ явним і неявним методами Ейлера з початковою умовою $x_0 = 0$ і кроком $h= 2$. Якщо умова рівняння порушується, зробіть висновок про стійкість або нестійкості обчислень.

5 Розрахуйте максимальну інформаційну швидкість в каналі передачі даних із смугою пропускання 4 кГц і відношенням сигнал/перешкода 130?

3 АВТОМАТИЗАЦІЯ ПРОЕКТУВАННЯ В РАДІО-ЕЛЕКТРОНІЦІ

3.1 Етапи та стадії проектування

Етапи та стадії проектування розберемо на прикладі розробки ультразвукового сканера. Сканер повинен вирішувати певне завдання. Відрізок часу від початку проектування приладу до зняття його з експлуатації називається життєвим циклом цієї технічної системи. У життєвому циклі виробу розрізняються етапи та стадії (рисунок 3.1). Коротко обміркуємо деякі з них. На початку проектування обґрунтовуються характеристики приладу та формулюється технічне завдання (ТЗ), яке направляють розробнику.

В ТЗ відокремлюють три розділи.

а) чисельні вимоги до вихідних параметрів (наприклад кути сканування, глибина сканування та ін).

б) чисельні дані, які характеризують діапазон виміру параметрів (сканер повинен працювати при зміні навколишньої температури від -60 до $+60$ °С, при коливанні напруги джерела в межах 27 ± 2 В та ін.);

в) якісний опис обмежень, потреб та умов, кількісна оцінка яких ускладнена.

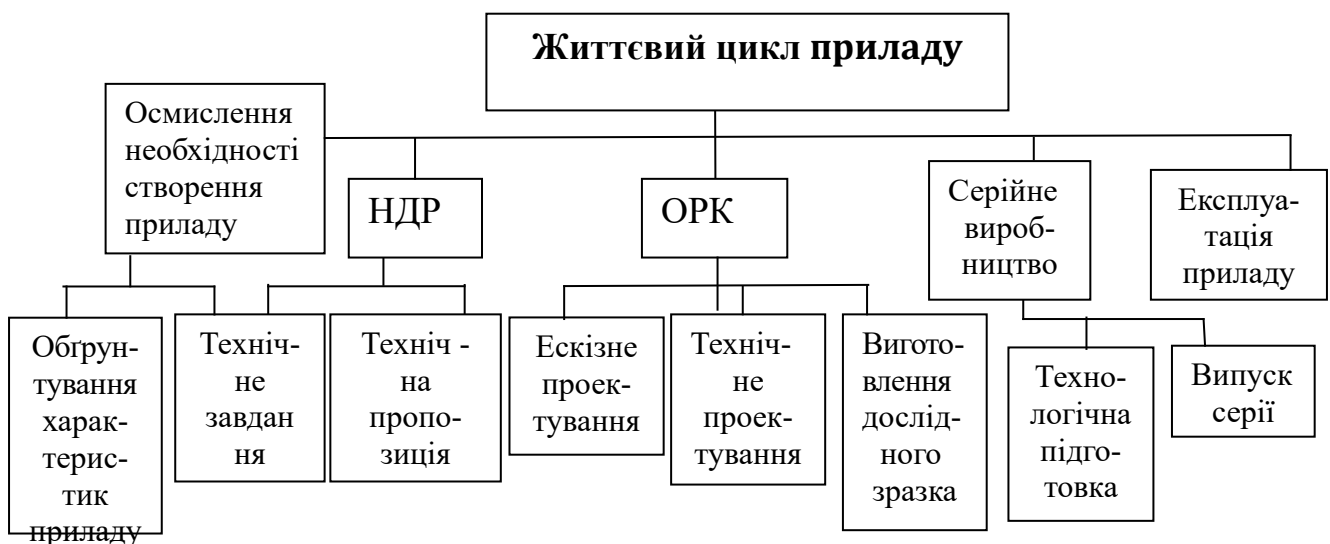


Рисунок 3.1 – Етапи та стадії життєвого циклу приладу

Розробник аналізує ТЗ, узгоджує його з їх замовником (при цьому уточнюється або навіть змінюється кожний із розділів завдання) та вирішує, з якого етапу почнеться проектування. Останнє визначається тим, наскільки новими виявляються потреби, чи можливо їх задовольнити модернізацією раніш використаних рішень або необхідний зовсім інший підхід.

На етапі науково-дослідної роботи (НДР) розглядається прилад у цілому та на стадії технічного пропонування визначаються принципи його дії – концепція приладу. Тому НДР називають стадією концептуального синтезу. У випадку із сканером на стадії НДР може, наприклад, обиратися вид випромінювання сигналу – імпульсний або безперервний.

Етап дослідно-конструкторської розробки (ДКР) вміщає стадії ескізного та технічного проектування, а також виготовлення дослідного зразка. При ескізному проектуванні визначається структура приладу і його гарт. Це стадія структурного синтезу. При технічному проектуванні у рамках прийнятої структури обираються її параметри, тобто здійснюється параметричний синтез. На останній стадії виготовляється дослідний зразок пристрою, здійснюються типові дослідження (кліматичні – температура, вологість; механічні – безперервні, змінні, ударні перевантаження тощо) та виготовляється технічна документація.

3.2 Блочно-ієрархічний підхід при технічному проектуванні

Структура блочно-ієрархічного підходу, який використовується при технічному проектуванні складних систем, складається з: системи, яка розбивається на вироби, пристроїв (блоків), вузлів і елементів (компонентів).

Сумісність блоків утворює один рівень, вузлів – інший, більш низький, компонентів – ще більш низький т. ін. Над рівнем пристроїв є рівень виробів. Якщо декілька виробів входять в деяку підсистему, то з'являється ще більш високий рівень.

Необхідність такого підходу зумовлюється тим, що складна задача великої розмірності розбивається на послідовні

розв'язання задачі прийнятої складності, яка піддається розумінню одним фахівцем відповідної кваліфікації.

Відмітимо особливість блочно-ієрархічного підходу: чим нижче рівень, до якого належить об'єкт, тим докладніше розглядається він у ході проектування. Відмітимо для прикладу на рівні компонентів біполярний транзистор. Припустимо, що при розробці сканера необхідно спроектувати новий транзистор. Причина цього полягає в тому, що існуючі транзистори не дозволяють побудувати передавач в потрібному діапазоні частот, а використання електронних приладів як підсилювачів неприпустимо через габарити. Очевидно, що проектування транзистора неможливо без докладного уявлення процесів, які діють у взаємодіючих р-п – переходах. Знання цих процесів дозволяє описати рух носіїв зарядів, знайти зв'язок між струмами й напругами на затискачах транзистора та встановити залежність цих величин від параметрів напівпровідникової структури. Математичні процеси описуються диференційними рівняннями із частинними похідними.

Перемістимось на рівень вузлів і відокремимо там підсилювач потужності. У його склад входить біполярний транзистор. При проектуванні цього підсилювача немає необхідності описувати транзистор дуже докладно, як це робиться при його розробці. Нас цілком задовольнить еквівалентна схема у вигляді триполюсника певної структури з такими параметрами, які дозволяють, наприклад, за заданим напругами на затискачах знайти струми, досить близькі до струмів справжнього транзистора. Використовуючи подібну еквівалентну схему, ми відвертаємо увагу від дифузії, дрейфу і рекомбінації носіїв заряду. Тепер залежність між струмами та напругами транзистора визначається звичайними диференційними рівняннями.

У зв'язку із сказаним зробимо деяке узагальнення. Зв'язок між струмами та напругами транзистора, коли він розглядається на компонентному рівні, визначався диференційними рівняннями у частинних похідних. Це й задавало математичну модель транзистора. На рівні вузлів зв'язок між цими величинами описувався вже звичайними диференційними рівняннями, що теж визначало математичну модель транзистора, але вже іншу.

Зрозуміло, що перша модель точніше в тому разі, коли в ній можна врахувати велику кількість реально діючих факторів. У другій моделі знання процесів, які протікають, важливо лише в тій мірі, в якій вони виявляють вплив на зовнішні характеристики об'єкта. Домовимося називати першу модель просто моделлю, а другу – макромоделлю.

Об'єкт, проектуванням займає в блочно-ієрархічній структурі найнижчий рівень, де він зустрічається вперше. Коли у цьому місці ми бажаємо описати його математично, то будемо мати справу з математичною моделлю цього об'єкта. Той самий об'єкт ми зустрічаємо і на сусідньому більш вищому рівні, але там він уже у спроектованому вигляді входить до складу більш складного утворення. Тому його математичний опис можна спростити, скористувавшись макромоделлю. Спрощення, тобто перехід до макромоделі, доцільне тим, що інакше різко зростає кількість рівнянь, які визначають об'єкт на більш вищому рівні. Характерна особливість макромоделі складається таким чином: об'єкт уявляють у вигляді “чорного ящика” з тією ж кількістю входів та виходів, як і у реального; вихідна реакція цього “ящика” на відповідний вхідний вплив повинна бути близькою до вихідної реакції вихідного об'єкта.

Є ще одна особливість блочно-ієрархічного підходу: для одного й того ж виробу складність рівней може бути різною. У одному варіанті до складу системи випромінювання увійде направлений відгалужувач, до якого через відповідні рознімачі приєднується приймач та передавач. Інші варіанти одержують, якщо направлений відгалужувач ввести до складу приймача або передавача. Як будуть сформовані рівні, залежить від багатьох обставин, зокрема від наявності потрібних фахівців, існуючої структури підрозділу на виробництві і т. ін.

Після того, як установлена блочно-ієрархічна структура, формується ТЗ на об'єкти кожного рівня. Відбувається це послідовно, з більш вищого рівня на більш нижчий. Наприклад, ТЗ на передавач надійде з рівня виробу, а ТЗ на вузли передавача – з рівня пристроїв. Розробник вузлів приймає рішення про компоненти: використовувати стандартні або розробляти нові. На нові компоненти формується ТЗ.

Розглянемо, які види проектування треба здійснити на рівні вузлів. Для конкретності візьмемо вузли передавача. У ході проектування цих вузлів повинна бути створена схема і розроблена конструкція. Для виготовлення вузла необхідно визначити технологію. Тому кажуть, що на рівні вузлів треба провести схемотехнічне й конструкторсько-технологічне проектування. Звернемо увагу на певну умовність у поділу проектування вузла на два вказаних типа. У ряді випадків, наприклад при розробці вузлів НВЧ, не вдається відділити схемотехнічне проектування від конструкторсько-технологічного. В ході викладу подальшого матеріалу ми збережемо поділ проектування вузла на два види й сконцентруємо увагу на схемотехнічному проектуванні.

3.3 Структура проектування схеми вузла

При автоматизації будь-якого процесу, зокрема проектування, виникає необхідність формально описати його. З цією метою утворюється структурна схема проектування, якою ми скористаємось, трохи видозмінивши її (рисунок 3.2), для з'ясування суттєвості схемотехнічного проектування на рівні вузлів.

Для визначеності будемо мати на увазі розробку схеми підсилювача потужності передавача. ТЗ на підсилювач, як би на всякий об'єкт проектування, складається з трьох частин. Нехай у першій частині сформульовані вимоги до вихідних характеристик ($P_{\text{вих min}}$ на середній частоті, смуга частот з урахуванням критерію, за яким вона визначається тощо). У другій частині зазначені параметри зовнішнього середовища (діапазон робочих температур, межі зміни живильної напруги, вихідного навантаження, потужності на вході тощо). Розглянемо, звертаючись до рисунка 3.2, як традиційно, тобто без допомоги ЕОМ, відбувається розробка схеми підсилювача.

Розробник, отримавши ТЗ, аналізує його і вигадує (синтезує) структуру майбутнього підсилювача. Цей крок не можна описати формально, він є творчим актом, тому що немає правил, керуючись якими, можна було б із технічних потреб визначити структуру підсилювача, що задовольняє ці вимоги.

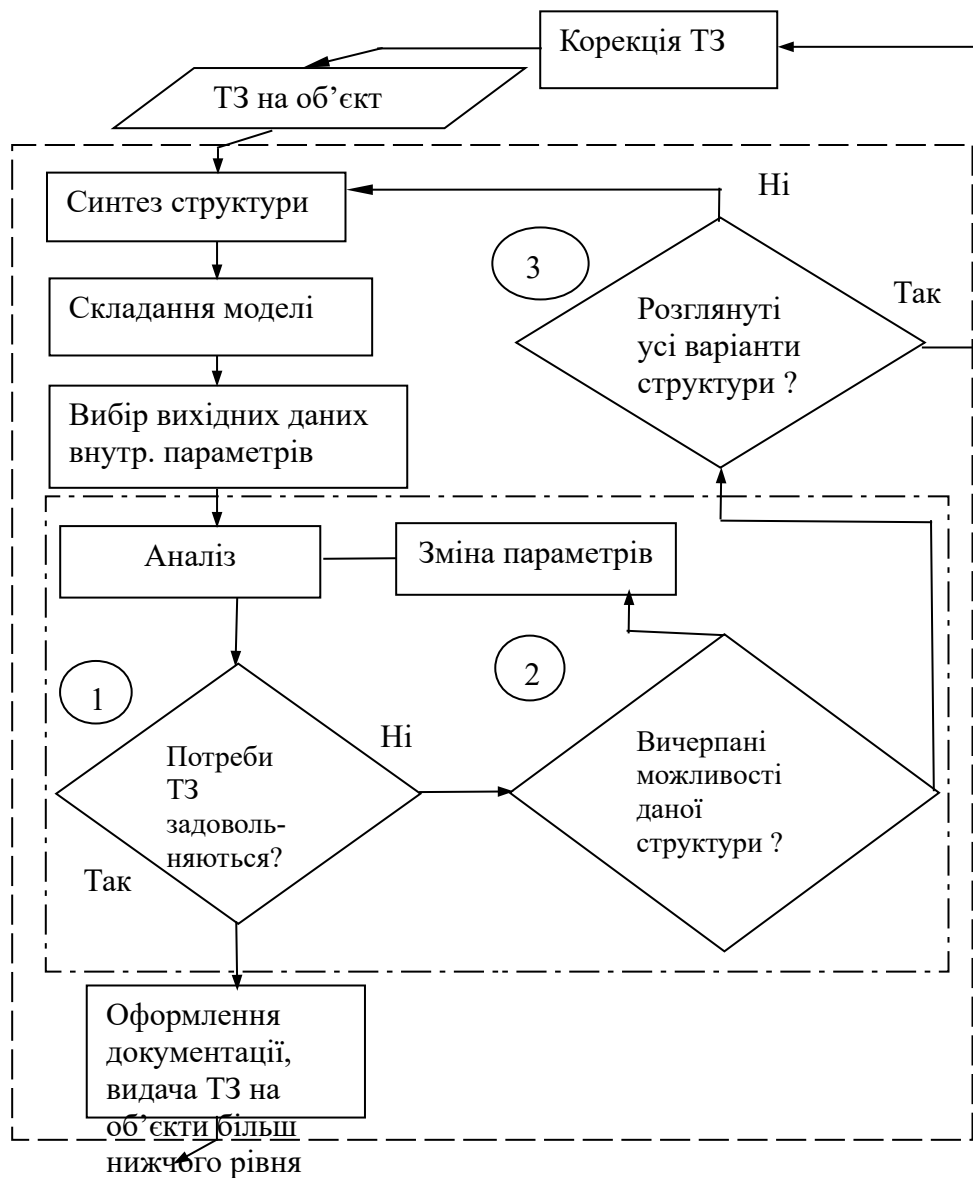


Рисунок 3.2 – Структурна схема проектування на рівні вузлів

Припустимо, основується на даних розробки інших підсилювачів, користуючись своєю досвідченістю та інтуїцією, розробник зупинився на варіанті (рисунок 3.3): в підсилювачі буде використаний транзистор, який може забезпечити потрібну потужність при ввімкненні його за схемою із загальним емітером; у вхідному колі узгодження буде використаний Т – фільтр, а на виході – П – фільтр.

Наступний крок в розробці (рисунок 3.2) – складання моделі. Він необхідний для того, щоб обрати початкові значення параметрів. Припустимо, прийнято рішення: описати транзистор зарядовою моделлю з апроксимацією нелінійних функцій

відрізками прямих; у фільтрах на вході та виході (рисунок 3.3) елементи $Z_{1,2,6}$ позначити індуктивностями із втратами, а $Y_{3,4,5}$ – ємностями.

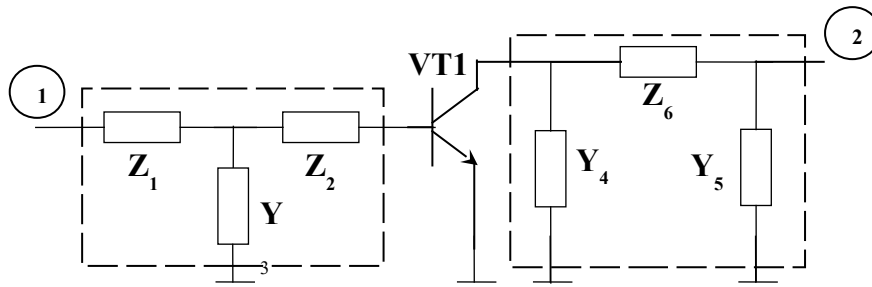


Рисунок 3.3 – Структура проектованого підсилювача потужності

Вибір початкових значень параметрів у започаткованій моделі підсилювача виконуємо за допомогою інженерної методики розрахунків, після визначення із паспортних даних параметрів моделі транзистора. В результаті розрахунків на середній частоті знайдемо режим транзистора (постійну напругу на його виходах і постійні струми, які протікають крізь них, а також амплітуди перших гармонік напруги і струмів) та визначимо значення ємностей і індуктивностей у фільтрах.

Отриманих знань досить для створення первісного варіанта підсилювача – його макета. У більшості випадків цей варіант не буде задовольняти потреби ТЗ. Причина такого результату у великій похибці методики розрахунків, обумовленій знехтуванням цілого ряду факторів як у використаних моделях, так і в розрахункових співвідношеннях. Отримати більш точні результати при розрахунку вручну практично неможливо. Отже традиційне проектування орієнтується на обробку схеми в натурному макеті. Іншими словами, розробник, добиваючись задовільнення технічних потреб на макеті, намагається врахувати вплив факторів, якими він знехтував при розрахунку.

На рисунку 3.2 процедура обробки схеми на макеті здійснюється в блоках, обведених штрих-пунктирною лінією: вимірюються вихідна характеристика при деяких значеннях параметрів (блок “аналіз”), результати порівнюються з вимогами ТЗ (логічний блок 1), при отриманні негативної відповіді у цьому та сусідньому блоках змінюються параметри, і все починається спочатку. Так робиться до тих пір, поки в логічних блоках 1 та 2

не будуть отримуватися позитивні відповіді. У першому випадку це означає, що потреби ТЗ виконуються і схема розроблена, у іншому – що при заданій структурі підсилювача не можна задовольнити ТЗ. Тоді необхідно змінювати структуру підсилювача і повторювати увесь процес спочатку. Якщо потреби не задовольняються при переборі можливих варіантів структури (відповідь “так” у логічному блоці 3), то необхідна корекція ТЗ. Для цього треба виходити на рівень, із якого надійшло ТЗ.

Описана процедура підбору параметрів макета має рутинний характер. Кажучи інакше, цю процедуру можна описати формально і, отже, автоматизувати. Сказане має слушність і у відношенні вибору початкових значень параметрів проєктованого підсилювача.

Розглянутий за допомогою рисунка 3.2 процес проєктування підсилювача визначає ітераційний характер розробки: створюється проєктний варіант, після чого цикл повторюють знову. Необхідність ітерацій викликана тим, що немає можливості зробити точний синтез, тобто знайти таку структуру і її параметри, які гарантували відповідність вимогам ТЗ.

Описаний перебір параметрів підсилювача з метою задовольнити ТЗ називається параметричним синтезом. Його можна зробити, підпорядкувавши деякому критерію. У цьому випадку кажуть про параметричну оптимізацію. Наприклад, підсилювач можна оптимізувати щодо мінімальної потреби живильної потужності від джерела живлення. Якщо структурний синтез проводиться на основі якогось критерію, то його також називають структурною оптимізацією.

За допомогою структурної схеми проєктування (рисунок 3.2) можна проєктувати об'єкти інших рівнів, а також розповсюдження його на інші види проєктування (наприклад, конструкторсько-технологічне).

3.4 Труднощі традиційного проєктування

Щоб зрозуміти труднощі, які з'являються в ході відпрацювання макета, нагадаємо мету макетування. Вона полягає в отриманні таких даних про пристрій (наприклад, про підсилювач потужності), за якими його можна було б передати у

виробництво, до того ж повинна бути впевненість, що після виготовлення цей пристрій буде задовольняти ТЗ.

За уявною простотою сказаного залишається великий об'єм вимірювань та випробувань. Дійсно, спочатку треба домогтися, щоб вимоги задовольнялись при номінальних значеннях параметрів зовнішнього середовища. У випадку з підсилювачем потужності це означає, що вимоги ТЗ повинні виконуватися при нормальній температурі навколишнього середовища, номінальних значеннях живильної напруги, опорі навантаження тощо. Відтак потрібна перевірка характеристик підсилювача в умовах варіації живильної напруги, опорі навантаження, температури та зміни інших чинників, обговорюваних в ТЗ. Нарешті, буде потрібно визначити працездатність підсилювача при розкиді параметрів елементів у межах допуску на них.

Кожний з трьох згаданих кроків, досить трудомісткий сам по собі, може неодноразово повторюватися в силу прийнятого вище ітераційного характеру проектування. Більш того, на кожному кроці не виключено перебільшення гранично досяжних напруг або струмів в якому-небудь елементі (наприклад, у транзисторі), що приведе до виходу цього елемента з ладу, що збільшить втрати часу на розробку. Щодо розкиду параметрів елементів, то облік його можливий, якщо є набір компонентів із крайніми значеннями параметрів, припустимими технічними умовами на використання деталей. Таким набором проектувальник, застосовуючи тільки що розроблені компоненти, як правило, не володіє. Все це приводить або до затримання терміну, або до неповної обробки всіх питань, тобто до неякісного проектування.

Як наслідок низька якість розробки макета може виявитися на будь-якому наступному етапі: у дослідному виробництві, при типових або державних випробуваннях, у ході виготовлення серії. Недоліки, що виникають на кожному з перелічених етапів, вкрай неприємні, але особливо вони болісні у серійному виробництві. Тут розробник повинен започаткувати вкрай термінові заходи, оскільки виробництво зупиняється при виході якого-небудь показника за межі потреб технічних умов.

Здолати перелічені труднощі традиційного проектування вдається за допомогою ЕОМ. Звернемось знову до прикладу з

транзисторним підсилювачем потужності для сканера. Припустимо, завершена підготовка до виготовлення макета. Отже, є схема підсилювача, відомий тип транзистора і його параметри, визначені номінали інших елементів схеми. Припустимо, процеси в підсилювачі можна описати математично, до того ж із доброю точністю. Вводячи математичний опис у ЕОМ, здійснимо машинне моделювання, уточнимо параметри, досягнувши задоволення технічних потреб: розглянемо вплив температури, живильної напруги, навантаження, вхідної потужності; зважимо на розкид параметрів у компонентах схеми, виявимо запас з гранично припустимих режимів і т. ін.

Машинне моделювання можна здійснити швидше та краще, ніж натуральне макетування. Само собою зрозуміло, простіш та швидше змінити параметр елемента при розрахунках, ніж фізично уявити цей елемент в макеті. Зміна параметра того або іншого компонента не викликає виходу із ладу схеми. Відпадає проблема створення набору елементів із крайніми значеннями параметрів.

Видно можливості, яких набуває розробник, використовуючи ЕОМ у ході проектування. Проте потрібно підкреслити, що у порівнянні з традиційним проектуванням відмічені переваги реалізуються при здійсненні ряду умов.

Потрібна математична модель розроблюваного об'єкта, в якій враховувались всі фактори, що практично впливають на його вихідну характеристику (ця умова у деякій мірі роз'яснює термін “ добра точність ” моделі). По-друге, повинні бути відомі параметри математичної моделі. По-третє, треба створити алгоритми розрахунків, задіяні у ході машинного моделювання. По-четверте, за розробленими алгоритмами належить скласти програму для ЕОМ. Нарешті, потрібна ЕОМ, яка могла здійснити необхідні розрахунки за прийнятний час.

3.5 Ручне та автоматизоване використання ЕОМ у процесі проектування

З моменту свого народження ЕОМ практично відразу стали використовуватися при розробці технічних систем. На перших порах проектувальник повинен був зробити ряд підготовчих

кроків перед виходом до ЕОМ: математично описати процеси у проєктованому об'єкті із заданою часткою точності (тобто обрати модель об'єкта та скласти її рівняння), знайти метод розв'язання рівняння та алгоритм розрахунків, записати цей алгоритм у вигляді програми та ввести її в машину разом з чисельною інформацією. Зрозуміло, що все перелічене вимагало значного часу та значних зусиль. Тому таке використання ЕОМ, яке пізніше стали звати ручним, мало обмежений характер.

Але отриманий досвід вказував на великі можливості цього шляху, хоч і розкривав його значні недоліки. Головний з них – необхідність фахівцю залучити до розв'язання своєї задачі професійного програміста або самому освоїти програмування. І перше і друге не так просто реалізувати. Перший спосіб потребує значної кількості програмістів. Другий зв'язаний з великими втратами часу при оволодінні алгоритмічними мовами та прийомами програмування, причому ці затрати не виправдовуються внаслідок швидкої втрати навичок при відсутності регулярної практики.

Через деякий час з'являється інший спосіб взаємодії розробника з ЕОМ, який отримав назву автоматизованого проєктування. Суть його полягає в наступному. Фахівцю не потрібно обирати модель об'єкта, складати рівняння, визначати чисельне рішення, розробляти алгоритм та писати програму. Все це зроблено раніш при утворенні проблемно-орієнтованої програми. В ній об'єднані кілька взаємодіючих програм, тому вона отримала ще одну назву – пакет прикладних програм (ППП). Пакет призначався для розв'язання однієї проблеми, наприклад для аналізу лінійних електронних схем. Ця проблема потребує послідовного розв'язання ряду завдань, які можуть відрізнятися як постановкою, так і методами. Треба зауважити, що проблемно-орієнтована програма дозволяє фахівцю вводити до ЕОМ задачу в предметній формі, наприклад, у вигляді електронної схеми. Іншими словами, вхідна мова програми зрозуміла фахівцю (спеціалісту) і не потребує великих зусиль для його засвоєння. Очевидно, ППП значно поширює коло людей, які використовують ЕОМ в проєктуванні, оскільки тепер знижуються потреби до рівня підготовки користувача в галузі програмування. З іншого боку, немає необхідності в збільшенні

кількості програмістів, так як створений пакет експлуатується продовж певного часу без великих переробок та обслуговує багато користувачів.

В теперішній час є кілька подібних пакетів, призначених для аналізу електронних схем. Як приклад назвемо такі програми: MICRO-CAP та ELECTRONICS WORKBENCH, DESIGN LAB, ORCAD, P-CAD, MICROWAVE OFFICE, SERENADE.

Проектування подібних програм – це плід зусиль колективу, утвореного з програмістів різної кваліфікації та фахівців у сфері радіоелектроніки. Поява проблемно-орієнтованих програм народжує передумову до переходу до системи автоматизованого проектування.

3.6 Комплекс засобів системи автоматизованого проектування

Для конкретності викладення будемо мати на увазі систему автоматизованого схемотехнічного проектування (АСхП) одного вузла передавача, яка може входити в крізну САПР цього вузла. Система АСхП складається з комплексу засобів, завдяки яким здійснюється її функціонування. Цей комплекс вміщає кілька видів забезпечення: математичне, технічне, програмне, інформаційне, лінгвістичне, методичне та організаційне.

Розглянемо перелічені види забезпечення.

Під математичним забезпеченням розуміють методи розв'язання проектних завдань та їх алгоритми, математичні моделі об'єктів проектування та засоби отримання їх параметрів.

Технічне забезпечення – це засоби для прийому, обробки, збереження та передачі інформації, тобто ЕОМ та периферійне обладнання, для вводу та виводу інформації, для виготовлення документації тощо.

Програмне забезпечення включає до себе загальні та спеціальні програми. За допомогою перших здійснюється функціонування системи проектування, інші належать до розв'язання проектних задач.

В інформаційне забезпечення входять відомості, необхідні для виконання проектування: довідкові дані, стандарти тощо, а також результати проектування, які проводилися раніш для подібних вузлів.

Лінгвістичне забезпечення визначає сукупність мов для запису алгоритмів, опису вхідних даних та результатів розрахунків, для обміну інформацією між машиною та користувачем.

Методичне забезпечення дає методику використання систем для проектування, в якій записано склад системи та закладені правила її експлуатації.

Організаційне забезпечення – це положення, яке встановлює склад та функції підрозділів, де буде вестись автоматизоване проектування.

Розкриємо докладніше зміст деяких із перелічених видів забезпечення, маючи на увазі схмотехнічне проектування транзисторного підсилювача потужності (одного з вузлів медичного сканера).

Почнемо з математичного забезпечення. З цією метою перелічимо проектні задачі, які виникають при розробці підсилювача. Вони витікають із технічних потреб. Нехай, для прикладу, необхідно, щоб потужність в навантаженні підсилювача була в діапазоні $P_{\text{нmin}} - P_{\text{нmax}}$, якщо частота сигналу змінюється від f_{min} до f_{max} , а потужність на виході – від $P_{\text{нmin}}$ до $P_{\text{нmax}}$. Навантаження підсилювача – 50 Ом, такий і його вхідний опір.

Необхідно вибрати транзистор, який у заданому діапазоні частот дозволяв би розвинути потрібну потужність. Наступний крок – визначення структури та параметрів кола узгодження на вході та виході, які при відповідних живильних напругах забезпечують необхідний рівень вихідної потужності в деякій точці області значень частоти і вихідної потужності. Окрім цього, не повинні бути перевищені граничні струми, напруги та розсіяна потужність на транзисторі та інших елементах схеми.

До математичного забезпечення увійде методика вибору транзистора за робочою частотою і вихідною потужністю, методика визначення структури кіл узгодження і параметрів цієї структури за заданою смугою частот, рівнем потужності у навантаженні та на виході, за режимом транзистора. Крім того, математичне забезпечення повинно включати модель транзистора, адекватну умовам роботи підсилювача, спосіб отримання параметрів моделі, а також метод перевірки

функціонування розробленого вузла. Склад розглянутого виду забезпечення ще більш ускладниться, якщо треба врахувати, що підсилювач повинен працювати в зазначеному діапазоні температур, при зміні живильної напруги та навантаження. Ускладнення обумовлене тим, що треба не тільки виявити вплив перелічених факторів, але й вказати шлях зменшення їх впливу. Практика показує, що в теорії та методах дослідження транзисторного підсилювача потужності для деяких з цих задач поки що відсутнє повне рішення.

Перейдемо до обговорювання програмного та технічного забезпечення. Між ними є взаємозв'язок, від того вони і розглядаються разом. При створенні автоматизованих систем проектування перевага віддається ЕОМ і операційним системам, які дозволяють організувати діалог із користувачем. Необхідні під час розробки розрахунки виконуються за допомогою пакетів прикладних програм, які з'єднуються в інформаційно-сумісний програмний комплекс.

Діалог користувача з цим комплексом буде успішним, якщо ЕОМ має відповідний об'єм оперативної пам'яті із відповідною швидкодією. Мала швидкодія приводить до довгого очікування результату біля терміналу. Нестача оперативної пам'яті змушує розбивати програмний комплекс на невеликі блоки, кожний з яких може розміститись у відведеній для розв'язання ділянці пам'яті. В ході розрахунків із тривалої до оперативної пам'яті надходить потрібний в певний час блок, потім його замінює інший і т.д. Чим менший блок, тим їх більше, і тим частіше потреба у їх обміні, що різко підвищує витрати машинного часу. Таким чином, програмні та технічні засоби повинні відповідати задачам, які вирішуються в ході розробки.

Нарешті, зупинимося на інформаційному забезпеченні схемотехнічного проектування підсилювача потужності. У склад цього виду забезпечення увійдуть: моделі використаних транзисторів і їх параметри у вказаній області, де зберігається потрібна точність; інформація про можливості кіл узгодження заданої структури (наприклад, про смугу і максимальний коефіцієнт трансформації опору), архів спроектованих раніш підсилювачів із їх показниками тощо. Побудова інформаційного забезпечення мусить допускати оновлення змісту. Для цього

потрібен постійний аналіз нових структур кіл та безперервна робота з визначення параметрів моделей знову спроектованих транзисторів. Останнє у свою чергу впливає не склад математичного забезпечення, так як потребує створення і оновлення методики, за допомогою якої визначаються параметри моделі за результатами вимірів параметрів реального транзистора.

Питання, тести для самоконтролю

1 Автоматизація формування і редагування технологічної документації.

2 Автоматизація побудови технологічних ескізів деталей і заготівель за допомогою систем геометричного моделювання.

3 Автоматизація розрахунку завантаження робочих місць.

4 Автоматизація побудови об'ємних моделей деталей і заготівель.

5 Автоматизація вибору методів обробки поверхонь деталі.

6 Автоматизація контролю початкової інформації при автоматизації моделювання.

7 Автоматизація контролю правильності задання конструкторських розмірів при автоматизації моделювання.

8 Автоматизація контролю правильності задання технологічних розмірів при автоматизації моделювання.

9 Автоматизація виявлення структури припусків у системах автоматизації моделювання технологічних розмірів.

10 Автоматизації виявлення технологічних розмірних ланцюгів.

11 Автоматизація приведення допусків до стандартних значень при автоматизації моделювання.

12 Автоматизація розподілу допусків у системах автоматизації розрахунку технологічних розмірів.

Типові завдання та приклади їх розв'язання

Завдання 1

Використовуючи САПР AutoCAD, виконати фрагмент креслення передньої панелі приладу, наведеного на рисунку 3.4.

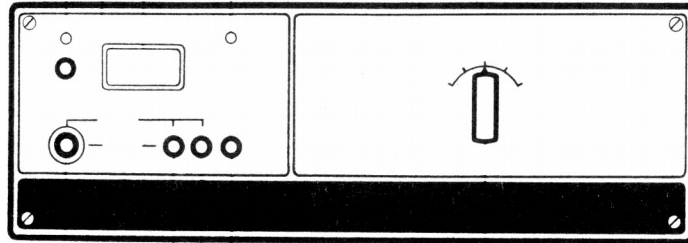


Рисунок 3.4

Розв'язання

Послідовність виконання креслення передньої панелі приладу така:

- завантажити AutoCAD;
- в головному меню AutoCAD вибрати пункт 1 (Begin a new drawing), який дозволяє створити нове креслення;
- виконати підготовчі дії такі, як: задати прив'язку графічного маркера до вузлів сітки, встановити режим виведення на екран координатної сітки, здійснити призначення шарів;
- здійснити автоматизоване виконання креслень в режимі DRAW з використанням таких команд, як LINE, CIRCLE, PLINE, TEXT;
- при необхідності відредагувати отримане креслення, використовуючи команди EDIT. В результаті отримаємо креслення, що наведено на рисунку 3.5.

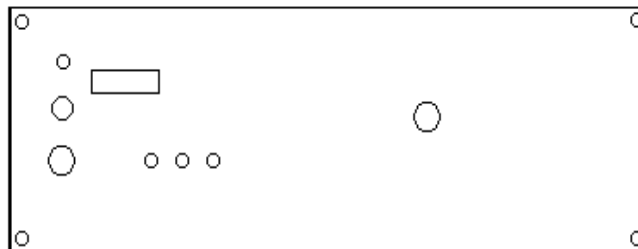


Рисунок 3.5

- виконати простановку розмірів у режимі DIM, використовуючи такі команди, як BASELINE, CONTINUOUS, DIAMETER;

- зберегти отримане креслення за допомогою команди SAVE;

- отримати копію креслення на плоттері або принтері, використовуючи відповідно команди PLOT A DRAWING або PRINTER PLOT A DRAWING головного меню.

Остаточний результат креслення передньої панелі зображено на рисунку 3.6.

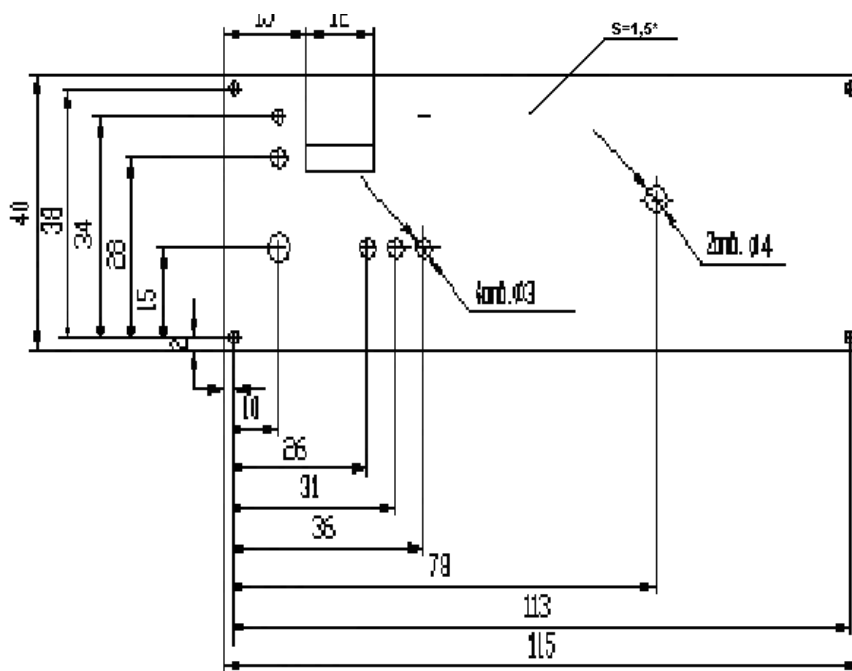


Рисунок 3.6

Завдання 2

Доопрацювати креслення друкованої плати (виконати простановку розмірів, сформувавши та внести технічні вимоги, заповнити основний напис), використовуючи файл з трасуванням з'єднань на платі TRAS.PCB. Простановку розмірів виконати, використовуючи різні шари, типи ліній, кольори.

Розв'язання

Для розв'язання цього завдання необхідно:

- завантажити AutoCAD;
- використовуючи програму PC-TRANS, трансформувати файл, отриманий в P-CAD;

- виконати простановку розмірів у режимі DIM, використовуючи такі команди, як HORIZONTAL, VERTICAL;
- сформулювати технічні вимоги до креслення та внести їх на поле креслення, використовуючи команду TEXT; заповнити основний напис;
- зберегти отримане креслення за допомогою команди SAVE;
- отримати копію креслення на принтері.

В результаті отримано креслення ДП, зображене на рисунку 3.7.

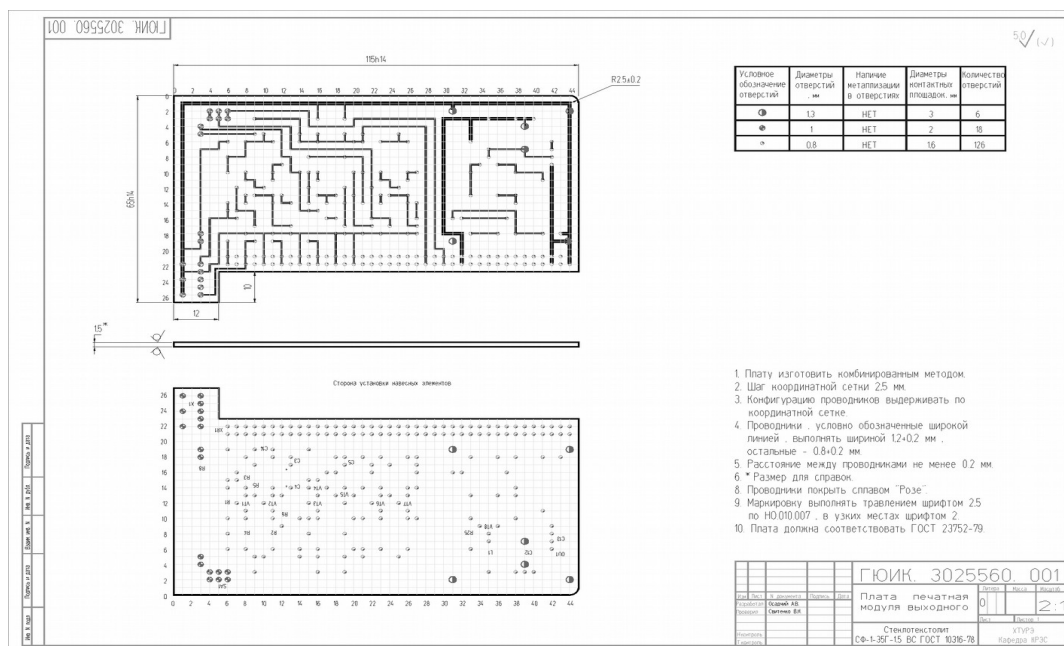


Рисунок 3.7

Завдання 3

Виконати в САПР AutoCAD складальне креслення модуля з друкованим монтажем, використовуючи файл, який містить розміщення елементів на платі RAZ.PCB. Простановку розмірів виконати, використовуючи різні шари, типи ліній, кольори.

Розв'язання

Використовуючи файл, який отримано в P-CAD, що містить розміщення елементів на платі, виконаємо складальне креслення. Послідовність дій така:

- завантажити AutoCAD;

- виконати підготовчі дії: створити формат креслення, задати прив'язку графічного маркера до вузлів сітки, встановити режим виведення на екран координатної сітки, задати масштаб, здійснити призначення шарів;
- виконати креслення в режимі DRAW з використанням таких команд, як LINE, CIRCLE, PLINE, TEXT;
- виконати постановку розмірів у режимі DIM, використовуючи такі команди, як HORIZONTAL, VERTICAL;
- сформулювати технічні вимоги до креслення і внести їх на поле креслення, використовуючи команду TEXT;
- заповнити основний напис;
- зберегти отримане креслення за допомогою команди SAVE;
- отримати копію креслення на принтері.

В результаті отримано складальне креслення, зображене на рисунку 3.8.

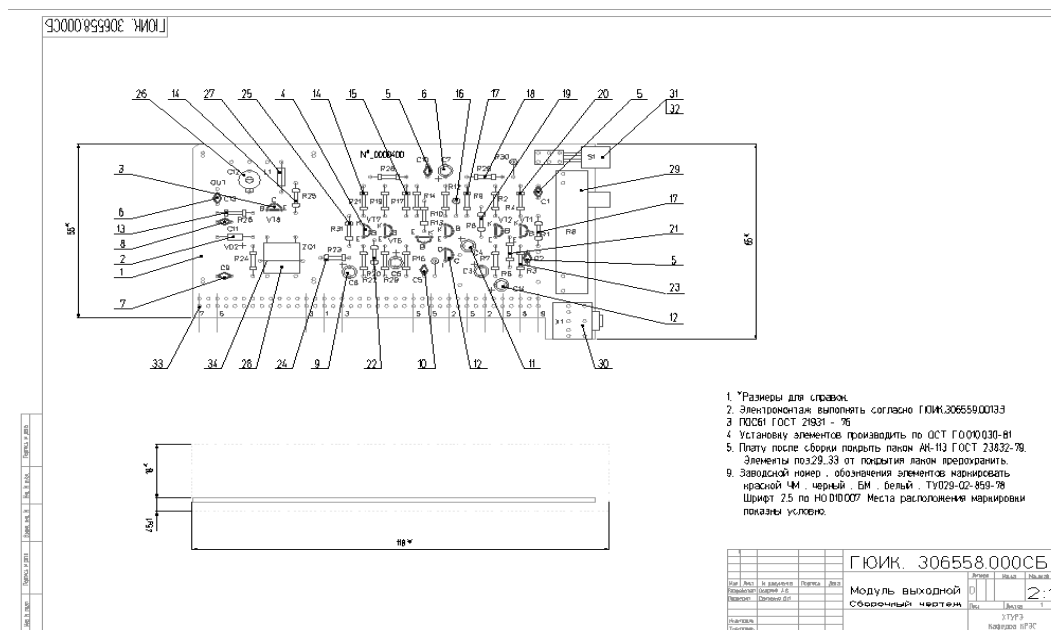


Рисунок 3.8

Індивідуальні практичні завдання

Завдання 1

Використовуючи вхідні дані конструкції радіатора та рисунок 3.9, виконати за допомогою AutoCAD його робоче креслення та об'ємне зображення.

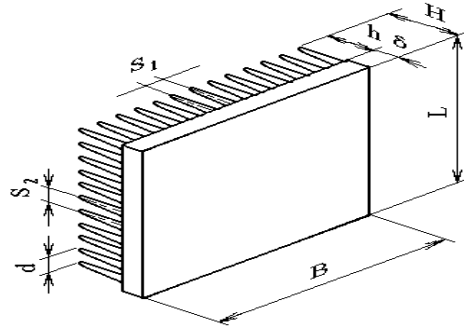


Рисунок 3.9

Завдання 2

Використовуючи дані, що містяться в складальному кресленні та схемі електричній принциповій, виконати в AutoCAD специфікацію на складальне креслення.

Завдання 3

Використовуючи файл, який отримано при розв'язанні завдання 1, виконати простановку розмірів ланцюжком, застосовуючи команду CONTINUOUS.

4 ПРОЕКТУВАННЯ ЕЛЕКТРОННИХ ПРИСТРОЇВ ТА СИСТЕМ АВТОМАТИКИ ЗА ДОПОМОГОЮ СПЕЦІАЛІЗОВАНИХ САПР

4.1 Огляд САПР друкованих плат

Будь-яка система проектування друкованих плат являє собою складний комплекс програм, що забезпечує наскрізний цикл, починаючи з прорисовки принципової схеми та закінчуючи генерацією керуючих файлів для обладнання виготовлення фотошаблонів, свердлення отворів, складання та електроконтролю. Однак умови сучасного ринку накладають додаткові вимоги на ці системи.

Компанія “Mentor Graphics” має власну систему проектування ДП Mentor BoardStation та продовжує розробку ліній продуктів “Expedition PCB” та “PADS Power PCB”. Пакет “Expedition PCB” являє собою найбільш потужне вирішення в галузі проектування плат. Основою системи є середовище “AutoActive”, яке дозволяє реалізувати такі функції, як передтопологічний аналіз цілісності сигналів, інтерактивне та автоматичне трасування з урахуванням вимог високочастотних плат та спеціальних топологічних обмежень, які накладаються при використанні сучасної елементної бази (BGA). Єдине середовище дозволяє за допомогою модуля ICX моделювати наводки в провідниках безпосередньо під час прокладання траси або шини та контролювати перевищення ними заданого рівня.

Компанія “Mentor Graphics” пропонує три варіанти видання “Expedition PCB”: Ascent – повний інтерактивний редактор топології без модуля автотрасування; Ascent LX – інтерактивний редактор з автотрасувальником без обмеження на число шарів, але без опції трасування високочастотних плат; Pinnacle – найбільш повна конфігурація, до складу якої входять усі можливості інтерактивної роботи та автотрасування. Інший продукт компанії “Mentor” – система “PADS Power PCB”, пропонує більш дешеве вирішення. Ця система обладнана кращим автотрасувальником “BlaseRouter”, що підтримує всі необхідні при трасуванні високочастотних плат функції. Пакет має модулі передтопологічного (HyperLinks LineSim) та

післятопологічного (HyperLinks BoardSim) аналізу, які тісно взаємодіють з системою контролю обмежень. Зараз ці модулі покращені за рахунок впровадження до них оригінальних алгоритмів моделювання, що раніше використовувалися у продукті ХТК, компанії “Innoveda”.

Компанія “Cadence” пропонує для верхнього рівня проектування пакет “PCB Design Studio”. Як редактор друкованих плат використовується програма Allegro, яка дозволяє розробляти багат шарові та високошвидкісні плати з високою щільністю розміщення компонентів.

Як модуль авторозміщення та автотрасування використовується програма “Specstra”, яка керується великим набором правил проектування та технологічних обмежень. Аналіз електромагнітної сумісності топології плати виконується за допомогою спеціального модуля “Specstra Quest SI Expert”, для попереднього аналізу проекту та підготовки наборів правил проектування використовується модуль SigXplorer.

Інший продукт від компанії “Cadence” – пакет “OrCAD” рекомендований як більш легке та дешеве рішення для проектування друкованих плат. Останнім часом продукт майже не розвивається, про що свідчать номери версій. Цей пакет розглядається фірмою “Cadence” як пріоритетна система вводу проектів та моделювання: модулі Capture SIC та PSpice зараз поставляються у складі пакету “PCB Design Studio”. В останню версію системи “OrCAD” увійшли нові можливості синтезу та моделювання цифрових логічних схем NC Sim. Редактор друкованих плат “OrCAD Layout” має три різні конфігурації з різними функціональними можливостями. У проекті плати може бути до 30 шарів, 16 з яких можуть бути сигнальними. Також наявні вбудовані можливості авторозміщення та автотрасування, а також інтерфейс з програмою “Specstra”.

Також можна виділити австралійську компанію “Altium”. Ця компанія випустила пакет “Protel DXP”, який є продовженням власної лінії продуктів Protel. Цей пакет забезпечує наскрізний цикл проектування змішаних аналого-цифрових плат з використанням програмованої логіки фірм “Xilinx” та “Altera”. Всі інструменти реалізовані на базі інтегрованої системи проектування “Design Explorer”, яка працює під керуванням ОС

Windows. До вже існуючих засобів післятопологічного аналізу цілісності сигналів (Signal Integrity) додалась можливість виконувати передтопологічний аналіз. Але головною новизною системи “Protel DXP” став автотрасувальник “Situs”.

Іншим, відомим насамперед у нашій країні, продуктом “Altium” є пакет “P-CAD”. До складу “P-CAD” входять два основних модулі — “P-CAD Schematic”, “P-CAD PCB” та інші додаткові програми.

Основною перевагою цієї системи є якісна підтримка вихідного формату даних ODB++. P-CAD залишається одною з найпопулярніших САПР друкованих плат, зокрема у нашій країні. Одними з нововведень в P-CAD стали функція “Design Manager”, яка спрощує процес керування даними проекту та є потужним засобом контролю його основних елементів, та інтерактивний інструмент “Visual Placement Area”, що аналізує проект на предмет пошуку областей, де може бути розташований той, чи інший компонент. Система P-CAD має можливість виділення різних електричних ланцюгів різними кольорами, що дозволяє користувачу краще орієнтуватися у контексті плати. P-CAD підтримує TTF-шрифти, що адаптовані до вітчизняних стандартів. Система забезпечує повний цикл розробки друкованих плат від створення символів елементів до розробки друкованих плат та випуску документації. Система P-CAD має можливість конвертувати файли до або з формату DXF, який доступний таким системам, як “OrCAD”, “КОМПАС”, “AutoCAD”, підтримуються формати ODB++ та Gerber.

4.2 Етапи проектування друкованої плати в САПР P-CAD

На етапі проектування топології друкованої плати здійснюється розміщення елементів на ДП і трасування з’єднань. Розв’язання цієї задачі можливо здійснити за допомогою САПР, характеристика яких була наведена в попередній главі. Розглянемо основні етапи проектування за допомогою однієї з потужних САПР – P-CAD.

Перед початком розробки ДП необхідно ознайомитись з бібліотекою компонентів принципових електричних схем (файли

з розширенням SYM) та бібліотекою їх конструктивів (файли з розширенням PRT).

Процес проектування ДП містить декілька етапів. На кожному з них використовуються окремі модулі P-CAD .

Етап 1. Створення креслення принципової електричної схеми (файл з розширенням SCH), укладання списків електричних зв'язків схеми і перевірка схеми.

Етап 2. Створення файлу бази даних ДП (файл з розширенням PKG) двома засобами:

а) на основі інформації, що міститься в файлах списків з'єднань, складених за принциповою електричною схемою (файл з розширенням NLT або XNL) і файли перехресних засилань (файл з розширенням FIL), а також файлу конструктиву ДП (файл з розширенням PCB);

б) на основі текстового файлу з розширенням ALT, в якому користувач описує склад проектованої ДП. Такий засіб застосовується менше.

Етап 3. Розміщення компонентів на ДП вручну за допомогою графічного редактора PC-CARDS або автоматично за допомогою програми PC-PLACE. В останньому випадку можна за допомогою програми PC-CARDS попередньо вручну розставити компоненти на ДП і трасувати чинність живлення і "землі".

Етап 4. Трасування з'єднань за допомогою програми PC-ROUTE або вручну за допомогою програми PC-CARDS.

Етап 5. Робота з допоміжними програмами (утилітами) для верифікації ДП, укладання креслень принципів електричних схем і внесення в них змін, випуск текстових звітів.

Етап 6. Випуск конструкторсько-технологічної документації.

На першому етапі здійснюється створення креслення принципової електричної схеми в редакторі PC-CAPS. Для створення і редагування схемотехніки друкованої плати використовується режим DETL. Після входу в режим DETL необхідно встановити робочі параметри (задання шарів) командою VLYR. Далі необхідно вивести елемент по відповідному імені з розширенням SYM. Вибравши орієнтацію, треба розмістити елемент. Далі необхідно провести електричні

з'єднання, використовуючи команду ENTR/WIRE, а для створення шин – команду ENTR/BUSS. Для створення окремих атрибутів використовується команда ATTR, для призначення схемних позначок – команда SCMD/PNUM. Далі треба виконати присвоєння імені елементу, для чого використовується команда NAME/COMP, для присвоєння імені ланцюгам – NAME/NET. Для запису файлу бази даних використовується команда – FILE/SAVE.

Для створення схемотехнічного опису елемента, якщо він не існує в базі даних, необхідно перейти в режим SYMB програми PC-CAPS. При створенні опису елементів задається: його графічне подання; точка прив'язки для подальшої його установки; графічне подання вхідних-вихідних контактів і контактів інших типів; атрибути; пакувальна інформація для подання схемного елемента в топологічному корпусі; ім'я контактів; розташування схемної позначки і номерів контактів.

Перед тим, як приступити до проектування топології, необхідно одержати список електричних кіл (файл з розширенням NLT) із схеми ЕЗ за допомогою програми PC-NODES, а потім створити файл перехресних засилань (файл з розширенням FIL), що встановлює відповідність між схемними та топологічними елементами. Файл перехресних засилань може бути створений в будь-якому текстовому редакторі. Після цього необхідно упакувати базу даних топології за допомогою програми PC-PACK.

На наступному етапі виконується розміщення елементів на ДП автоматично за допомогою графічного редактора PC-PLACE, або інтерактивне – за допомогою редактора PC-CARDS. Крім того, редактор PC-CARDS використовується для: редагування топології друкованої плати; перегляду рисунків топології; рисування полігонів і внутрішніх вирізів у них; автоматичного контролю введення провідників; запам'ятовування і відновлення елементів топології як блоків; відображення нерозведених зв'язків.

При створенні в PC-CARDS опису топологічних елементів (якщо вони не існують в бібліотеці топологічних елементів) спочатку вводяться всі контакти елемента і задається ім'я, тип і значення еквівалентності для кожного контакту. Після цього

команди PC-CARDS використовуються для створення графіки елемента. Може бути введено ім'я елемента, призначена точка розташування схемної позначки та додана інформація про упаковку. В рисунок елемента може бути доданий будь-який текст.

Альтернативний варіант розміщення компонентів на ДП перебуває у використанні програми автоматичного розміщення PC-PLACE. Перед початком автоматичного розміщення елементів необхідно вказати правила і параметри розміщення, використовуючи команди LATP, LATC, ASSG, CLR.

Після розміщення компонентів може бути виконане трасування з'єднань на платі інтерактивно в PC-CARDS або автоматично в PC-ROUTE. Програма PC-ROUTE дозволяє задати параметри і правила трасування, сукупність яких утворює стратегію трасування, після чого воно виконується автоматично. Якщо PC-ROUTE не завершує трасування, можна використати редактор PC-CARDS для коректування і завершення її.

4.3 Система проектування пристроїв за допомогою САПР OrCAD

Цілий ряд задач, які виникають при розробці схем таких пристроїв, можна вирішувати, використовуючи проблемно-орієнтовані пакети програм.

Розглянемо *два* види задач – *розрахунок транзисторного підсилювача потужності і трасування друкованої плати цього підсилювача.*

Розглядається, як за допомогою цієї програми підбирають параметри кола узгодження на виході і вході підсилювача, тобто як вирішують питання параметричного синтезу та відбувається пакування з наступним трасуванням друкованої плати медичного підсилювача потужності на сучасних SMD елементах.

Розрахунок транзисторного електронного підсилювача потужності. Використаємо програму OrCAD для розрахунку періодичного режиму в транзисторному підсилювачі потужності.

За допомогою програми OrCAD можна не тільки аналізувати підсилювач потужності із заданими параметрами кіл узгодження на транзисторах, але і здійснити параметричний

синтез підсилювача, тобто обрати параметри узоджуючих кіл із заданою структурою. Спочатку користувач створює свій проект з його персональною назвою, що виконується за командами: File>New Project, де далі вказує ім'я – Name з розширенням *.orj. Тип проекту вказується галочкою в рядку з назвою – Analog or Mixed – Signal Circuit що дає змогу в подальшому проводити аналіз, оптимізацію і трасування друкованої плати проектованого підсилювача потужності (рисунок 4.1).

Сам проект складається з декількох окремих файлів – файлу схеми принципової з розширенням *.dsn, файлу з'єднать елементів на схемі з розширенням *.net і т.ін.

Відповідно до майбутньої схеми підсилювача, підбираємо елементи з бібліотек, які люб'язно пропонуються розробниками програми. Користуючись довідниками з параметрів та назв імпортих і вітчизняних активних елементів, знаходимо повну чи часткову відповідність параметрів елемента вітчизняному.

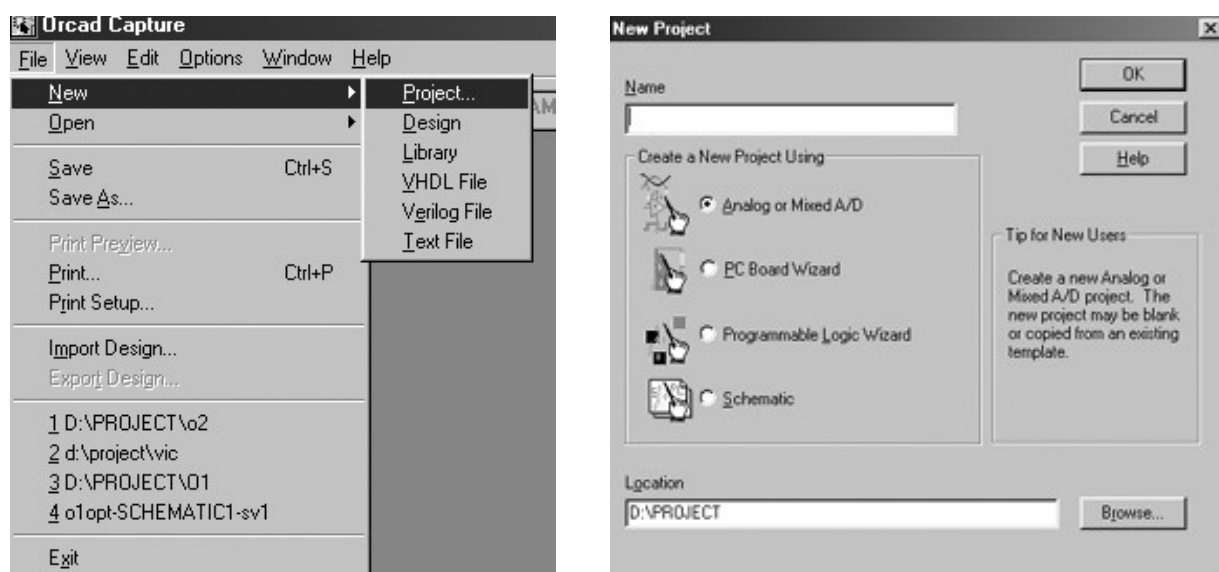


Рисунок 4.1 – Вибір та тип схеми нового проекту

Розміщення символів елементів відбувається за командою: Place>Part, при цьому зображення елемента з'являється на екрані монітора, де за допомогою миші він може бути розміщений у зручному місці. Під час такого “конструювання” схеми принципової медичного підсилювача потужності є змога, при виборі команди Edit Properties, проводити допоміжне корегування. За командою Part Reference встановлюється своє

позиційне позначення елемента за схемою, коли автоматичне позначення не задовольняє користувача. За командою PCB Footprint можна вибрати або скорегувати ім'я корпусу елемента для розміщення на друкованій платі. По закінченні процесу створення схеми обов'язково треба вказати хоча б один заземлений елемент.

За командою Place>Wire проводиться з'єднання елементів схеми згідно зі схемою принциповою (рисунок 4.2) за допомогою миші.

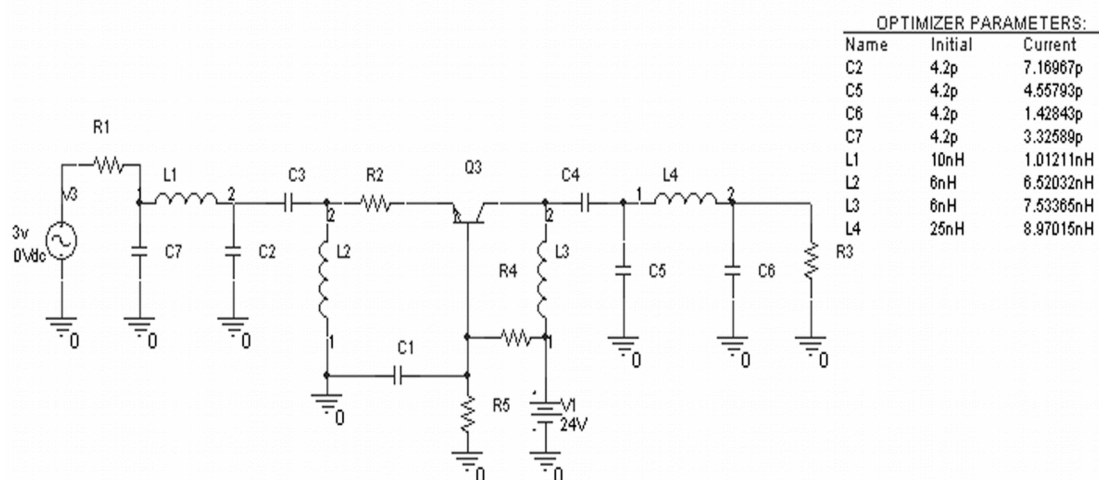


Рисунок 4.2 – Вигляд схеми принципової підсилювача потужності на екрані монітора програми OrCAD Capture

Якщо на зовнішній вигляд схема готова, то можна перейти до наступного етапу – складання списку з'єднань елементів схеми. За командою Tools> Design Rules Check (DRC) відбувається перевірка схеми на відсутність грубих помилок – розірваних кіл, відсутніх з'єднань і т. ін., а далі за командою Tools>Create Netlist складається перелік усіх з'єднань у схемі і створюється файл з розширенням *.mnl, який далі передається до програми Layout Plus для подальшого створення друкованої плати (рисунок 4.3).

Одночасно відбувається формування даних для моделювання схеми. Спочатку за командою Pspice>New Simulation Profile (рисунок 4.3) користувач створює завдання на моделювання, де вказує вид аналізу – в часі (Transient), в частотній області (AC Sweep), на постійному струмі (DC Sweep) і

т.ін. Після цього за командою Pspice >Run (рисунок 4.3) відбувається саме моделювання, при цьому дані пересилаються до програми Pspice де і проходять головні розрахунки. Результати розрахунків висвітлюються на екрані монітора, це уже робота наступної програми Probe, де є можливість корегувати графіки – робити їх більш зручними для користувача.

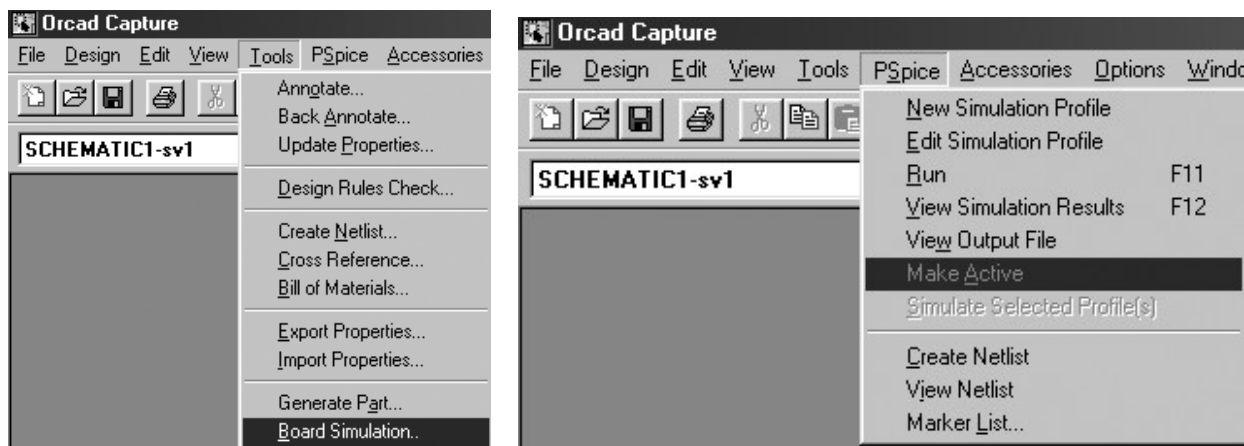
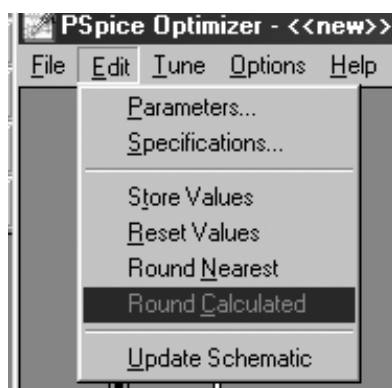


Рисунок 4.3 – Етапи складання завдання для створення друкованої плати

Оптимізацію параметрів елементів схеми виконують за допомогою програми Pspice Optimizer, яка за потребою може бути викликана з основної програми OrCAD або самостійно (рисунок 4.4).



OPTIMIZER PARAMETERS:		
Name	Initial	Current
C2	4.2p	7.16967p
C5	4.2p	4.55793p
C6	4.2p	1.42843p
C7	4.2p	3.32589p
L1	10nH	1.01211nH
L2	6nH	6.52032nH
L3	6nH	7.53365nH
L4	25nH	8.97015nH

Рисунок 4.4 – Програма оптимізації параметрів елементів схеми підсилювача потужності

Тут необхідно вказати за командою `Edit>Parameters` параметри тих елементів, які будуть змінними при оптимізації. Це вказується у вікнах, які з'являються за цією командою, де надається також ім'я цьому змінному елементу. Попередньо на схемі принциповій треба також зробити відповідну вказівку – за командою `Pspice>Place Optimizer Parameters` (рисунок 4.5) розмістити спеціальний напис і відредагувати його.

Для цього мишею роблять відповідні дії, натискуючи на ліву кнопку і записуючи початкові і поточні параметри цих елементів. Далі необхідно на схемі вказати, які елементи вважати змінними при оптимізації. Це зручно зробити, натиснувши два рази лівою кнопкою миші на зображення елемента, при цьому відкриється відповідна панель з переліком усіх властивостей, де в графі `Value` замість номінала елемента вказати в дужках його ім'я в завданні на оптимізацію, про що було сказано раніше. Далі треба визначитись із цільовими функціями, за якими будемо проводити оптимізацію. Це відбувається за командою `Edite>Specifications` (рисунок 4.5), де задається знову `Name` - ім'я, але вже складової цільової функції, `Weight` – вага цієї складової в загальній цільовій функції схеми, `Target` – саму мету до якої повинна прямувати ця складова цільової функції, `Range` – діапазон можливих відхилень цієї складової, тип аналізу `AC`, `DC`, `Transient`, ім'я файлу, де записані дані про електричні параметри схеми і створеного програмою `Pspice`, який має розширення `*.sim`, сам математичний вираз складової цільової функції. Багато математичних виразів – підпрограм, складених згідно з математичними виразами, розміщено в файлі `Pspice.prb`, які мають позначку `Global Goal Function` і призначені для відповідних видів аналізу – `AC`, `DC`, `Transient`. Користувач може сам задавати математичні вирази для його цільових функцій, але це повинно відповідати правилам складення виразів в системі `OrCAD`. Кількість складових цільової функції вибирається користувачем відповідно до задачі, яку він вирішує за допомогою програми оптимізації. В нашому випадку їх три, це `CenterFreq` – складова, відповідна за центральну частоту, до якої повинна прямувати цільова функція, `BandWidht` – складова, відповідна за ширину смуги частот, `max (V)` – складова, відповідна за одержання максимальної напруги на навантаженні підсилювача.

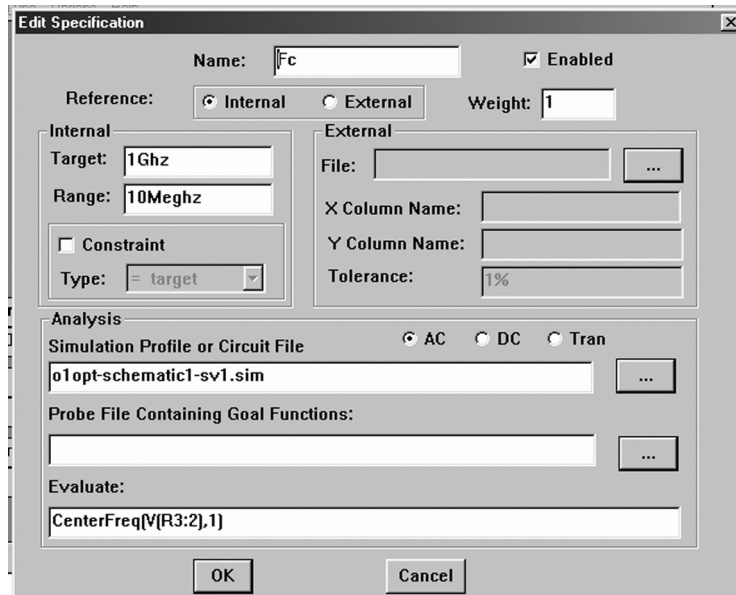


Рисунок 4.5 – Вибір значень складових цільової функції

Підготовлене таким чином завдання на оптимізацію, при відсутності помилок, може бути приведене в дію за допомогою команд Tune>Auto>Start (рисунок 4.6).

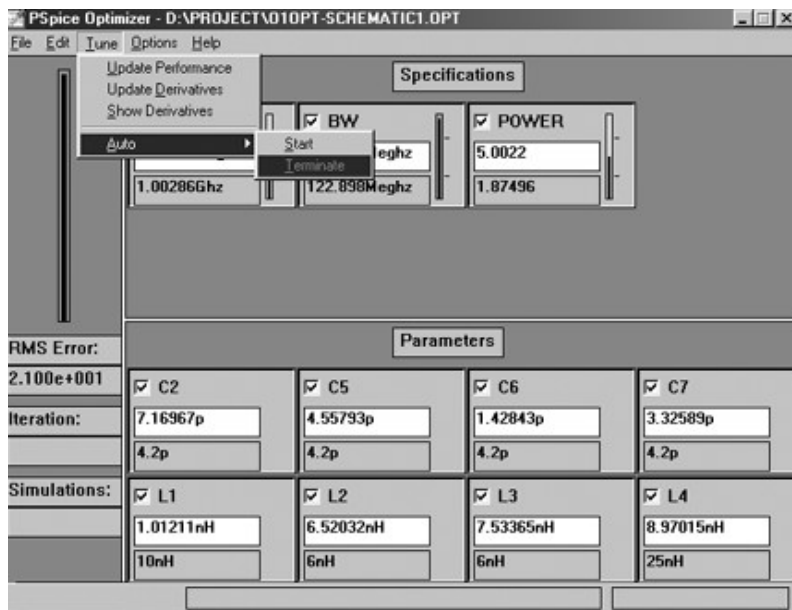


Рисунок 4.6 – Початок процесу оптимізації параметрів елементів схеми підсилювача потужності

Перед очима користувача висвітлюється інформація про хід оптимізації, кількість ітерацій, загальне відхилення розрахунків від заданої цільової функції, розраховані параметри елементів, що

перебираються програмою і т. ін. По закінченні оптимізації всі параметри можна передати схемі принципів, а можна записати окремо до файлу з розширенням *.opt для подальшої роботи з ним.

4.4 Редактор топології друкованих плат OrCAD Layout

Найбільш поширений спосіб розробки друкованої плати, це коли маємо складену схему за допомогою одного із графічних редакторів, наприклад, OrCAD Capture і переліку з'єднань схеми в форматі, який зрозумілий OrCAD Layout Plus.

Розробка нової друкованої плати починається після завантаження OrCAD Layout і виконання команди File>New. Спочатку запитується ім'я технологічного шаблону друкованої плати (розширення імені файла *.tch або *.tpl. Потім – ім'я файла списку з'єднань *.mnl. В кінці вказується ім'я файлу майбутньої друкованої плати *.max).

Перелік помилок, що з'явилися при завантаженні списку з'єднань, відтворюється в текстовому файлі з розширенням *.err.

За час завантаження списку з'єднань для кожного символу схеми у бібліотеках корпусів компонентів *.llb (Footprint Libraries) розшукується відповідний корпус. Ця відповідність вказується або за допомогою атрибута символу PCB Footprint , або в файлі System.prt.

Якщо при завантаженні списку з'єднань був знайдений компонент, який не має інформації щодо його корпусу, то видається повідомлення у вигляді діалогового вікна (рисунок 4.7) для його визначення.

Після натискання на панель Link existing footprint to component (Укажіть ім'я існуючого типового корпусу) з'являється діалогове вікно (рисунок 4.7), в якому треба вибрати ім'я бібліотеки, а потім ім'я корпусу, зображення якого видно в правій половині вікна.

Технологічні шаблони мають розширення файлів *.tch або *.tpl і мають початкову інформацію щодо друкованої плати: проміжки та сітки трасування, розміри контактних площин і т.ін. Далі всі ці установлення можна змінювати окремо або завантажити новий шаблон після утворення друкованої плати.

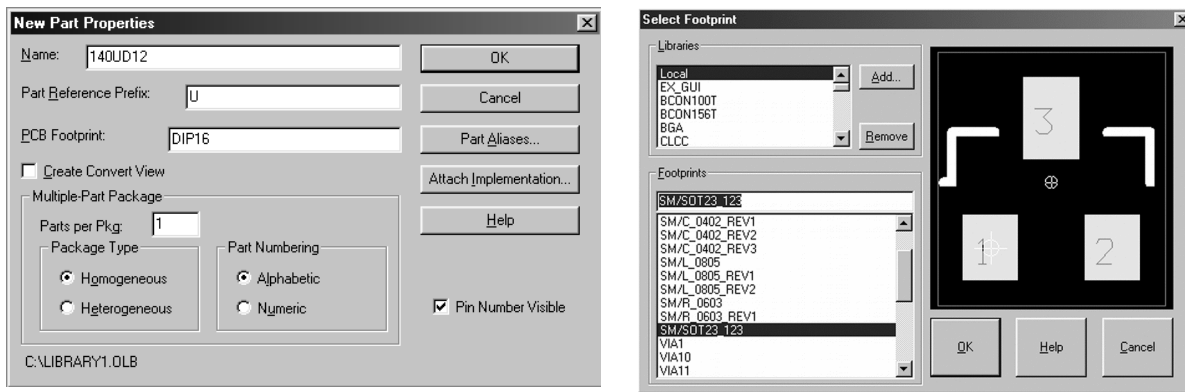


Рисунок 4.7 – Визначення корпусу компонента

Після завантаження технологічного шаблону до поточного проекту вносяться такі зміни:

- завантажуються стратегія розміщення компонентів і трасування провідників, заміщуючи попередні дані;
- встановлюється нова структура шарів друкованої плати;
- змінюються розміри кроків всіх сіток;
- змінюються параметри всіх стеків контактних площин (СКП); виводів компонентів і перехідних отворів (ПО).

Для утворення файлу технологічного шаблону необхідно утворити файл друкованої плати і зберегти його з явно вираженим ім'ям та розширенням *.tch.

Після вдалого завершення завантаження списку з'єднань на робочому екрані OrCAD Layout зображаються корпусу компонентів поточного проекту з вказівкою їх електричних з'єднань (рисунок 4.8).

В OrCAD є можливість передавати дані з OrCAD Layout до програми Specstra за командою File>Export>Layout to Specstra, як для виконання розміщення компонентів, так і для трасування провідників у ручному, інтерактивному або автоматичному режимах. Дані про розроблену плату передаються знову до OrCAD Layout за командою File>Import>Specstra to Layout для виготовлення документів і утворення керуючих файлів для фотоплотерів і свердлильних станків з ЧПК.

Але усі ці операції можна виконати і за допомогою OrCAD Layout. На початку за командою Options>System Setting у діалоговому режимі, як показано на рисунку 4.9, перевіряють і при необхідності, редагують глобальні параметри проекту.

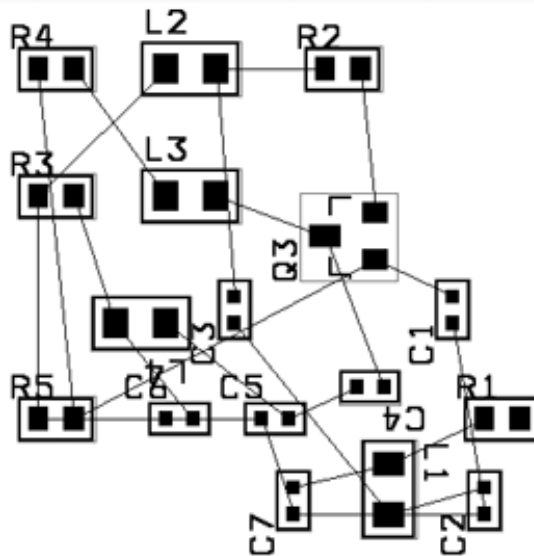


Рисунок 4.8 – Результат завантаження списку з'єднань підсилювача потужності на друковану плату

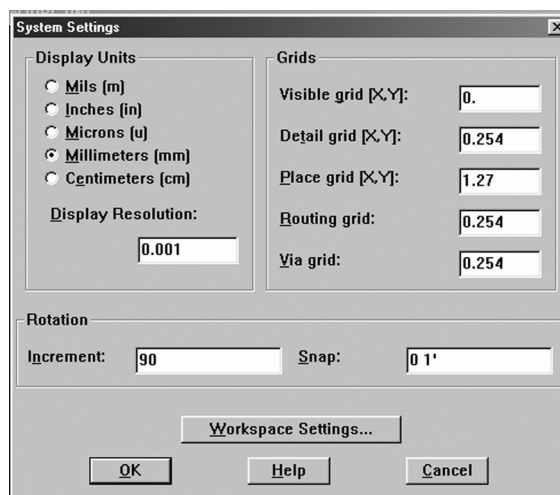


Рисунок 4.9 – Редагування параметрів проекту

Після цього за командою `Tool>Layer>Select From Spreadsheet` у діалоговому режимі перевіряють і при необхідності редагують кількість шарів друкованої плати і їх призначення. Редагування шарів відбувається за допомогою меню, яке відкривається при натискуванні правої кнопки миші. Далі, при необхідності, за командою `View>Database Spreadsheets>Padstacks` виконують в діалоговому режимі редагування стеків контактних площин (СКП) та перехідних отворів (ПО).

Перед розміщенням компонентів або трасуванням провідників необхідно накреслити межу майбутньої друкованої плати (Board outline). Для цього в меню Tool необхідно вибрати команду Obstacle, де накреслюється багатокутник, товщина ліній, межі якого вказуються в діалоговому вікні. За цією командою також визначають і інші межі розміщення або трасування і області металізації, як видно з рисунка 4.10.

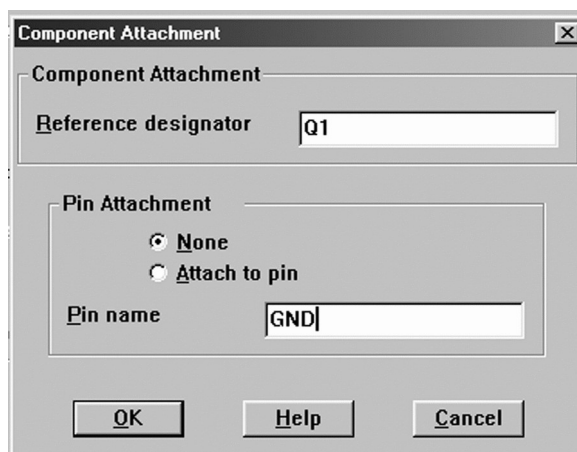


Рисунок 4.10 – Задання області металізації, підключеної до виводу одного компонента

Розміщення компонентів виконують в OrCAD Layout вручну, по черзі вибираючи, переміщуючи, повертаючи компоненти на поверхні зовнішнього шару друкованої плати і переміщуючи планарні компоненти на протилежний бік друкованої плати. В інтерактивному режимі розміщення відбувається за командою Auto>Place. Трасування провідників виконується вручну за командою Tool>Track, Track Segment, Via, Jumper, або в інтерактивному режимі за допомогою команди Auto>Autoroute.

Автоматичне розміщення компонентів відбувається за командою Auto> Place>Board тільки в програмі OrCAD Layout Plus. Авторозміщення виконується за декілька проходів (до 11) згідно з правилами, які редагуються за командою Options>Placement Strategy.

Редагування окремого параметра одного з проходів трасування або всіх його параметрів виконується подвійним натиском лівої кнопки миші.

Допоміжно за командою Option>Place Setting (рисунок 4.11) визначаються параметри стратегії авторозміщення компонентів.

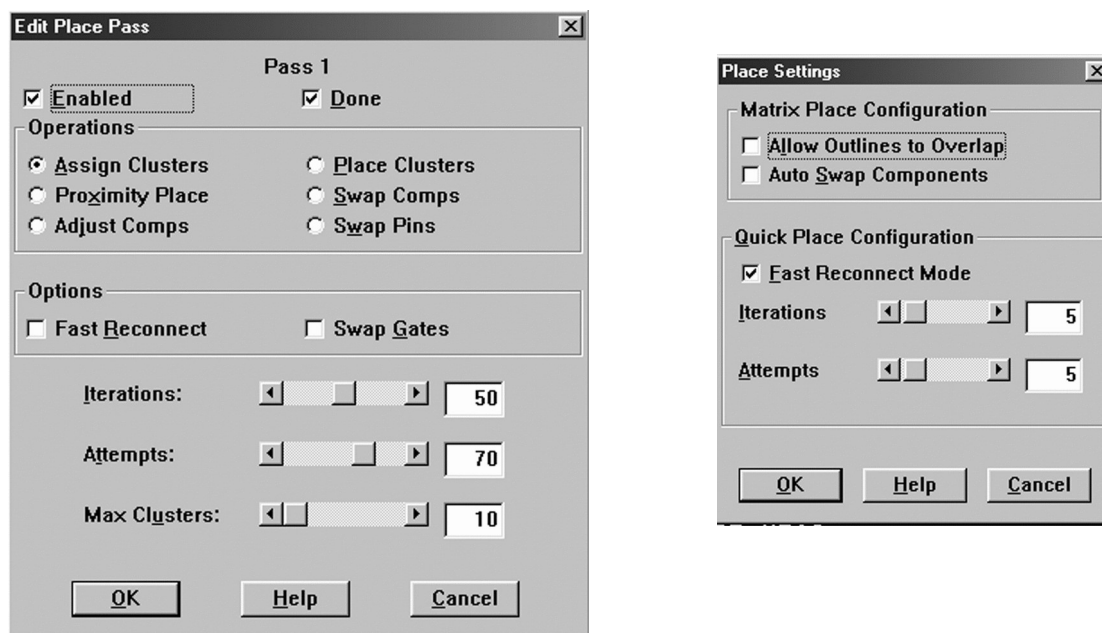


Рисунок 4.11 – Редагування параметрів вибраного проходу авторозміщення та параметрів швидкого і матричного розміщення компонентів

Автоматичне трасування провідників виконується в OrCAD трояко. По-перше, OrCAD Layout має в меню Auto групу відповідних команд, достатніх для розробки простіших друкованих плат. По-друге, для розробки більш складних плат призначена окрема програма SmartRoute, яка використовує алгоритм нейронних мереж. По-третє, з метою розміщення компонентів і трасування провідників вручну, в інтерактивному та автоматичному режимах можна використовувати програму Specstra, яка не входить безпосередньо до складу системи OrCAD. Розглянемо коротко перші два способи.

1-й спосіб. Часткові параметри стратегії автотрасування визначаються в діалогових вікнах, які викликаються за командами Option>Route Strategy>Manual Route, Route Layers, Route Sweeps, Route Passes.

Глобальні параметри стратегії автотрасування визначаються в діалоговому вікні, яке відкривається за командою Option>Route Setting.

2-й спосіб. Програма SmartRoute викликається за командою smart.exe із того місця, де вона розташована.

Спочатку завантажується файл друкованої плати *.max, а потім, згідно командами меню Options, визначаються параметри трасування в діалогових вікнах.

Кінець кінцем, за командою Routing Passes призначається перелік проходів трасування (Routing Passes) та проходів покращення технологічності плати (Manufacturing Passes). Автотрасування починається за командою Auto>AutoRoute>Board. Інформація про поточний стан трасування подається в нижній частині екрану, де видно хід трасування, процент виконання і т.ін. Програма має також режим ручного трасування, який викликається за командою Tools>Manual Route. Розведена однобічна друкована плата підсилювача потужності за програмою SmartRoute (рисунок 4.12) записується в файл програми OrCAD за іншим ім'ям із розширенням *. max.

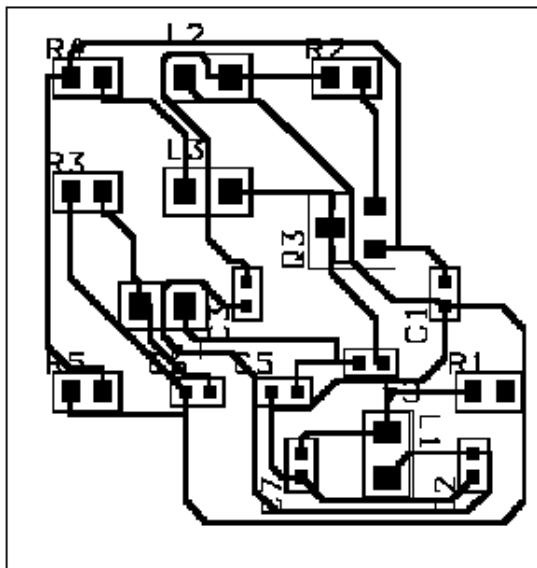


Рисунок 4.12 – Однобічна друкована плата електронного підсилювача потужності з використанням SMD елементів

Питання, тести для самоконтролю

1 Автоматизація розрахунку системи технологічних розмірів при моделюванні.

2 Автоматизація побудови технологічних ескізів деталей і заготівель за допомогою систем геометричного моделювання.

3 Автоматизація розрахунку завантаження робочих місць.

4 Автоматизація побудови об'ємних моделей деталей і заготівель.

5 Автоматизація вибору методів обробки поверхонь деталі.

6 Автоматизація контролю початкової інформації при автоматизації моделювання.

7 Автоматизація контролю правильності задання конструкторських розмірів при автоматизації моделювання.

8 Автоматизація контролю правильності задання технологічних розмірів при автоматизації моделювання.

9 Автоматизація виявлення структури припусків у системах автоматизації моделювання технологічних розмірів.

10 Автоматизації виявлення технологічних розмірних ланцюгів.

11 Автоматизація приведення допусків до стандартних значень при автоматизації моделювання.

Типові завдання та приклади їх розв'язання

Завдання 1

Використовуючи САПР P-CAD, виконати варіант розміщення елементів і трасування електричних з'єднань плати модуля, схема якого зображена на рисунку 4.13. Розміщення елементів виконати автоматично за допомогою редактора PC-PLACE. Трасування елементів виконати в редакторі PC-ROUT, використовуючи стандартну стратегію.

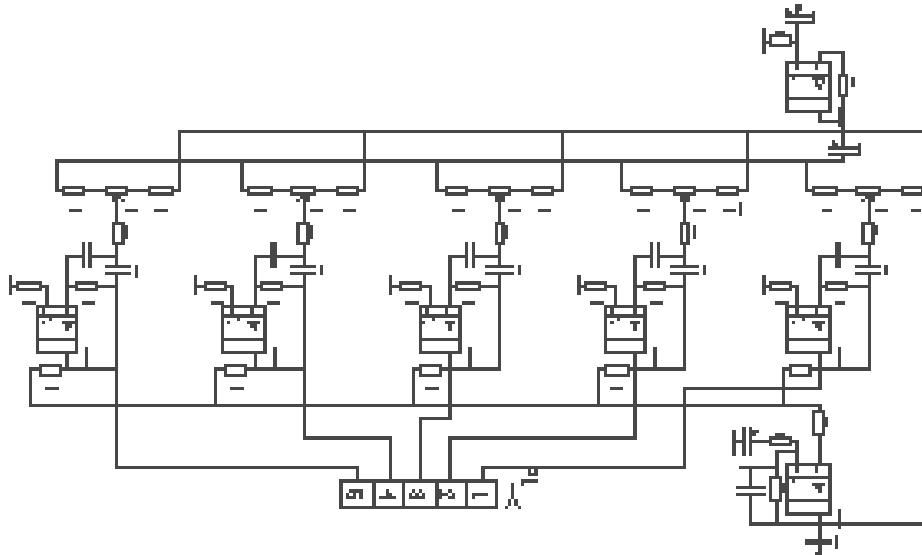


Рисунок 4.13

Розв'язання

1-й етап. Створення креслення принципової схеми (розширення SCH). Основні кроки зі створення принципової електричної схеми такі:

КРОК	Команди, що використовуються
1 Встановити режим DETL	
2 Встановити шари бази даних	VLYR
3 Розмістити УГП відповідно до схеми ЕЗ	ENTR/COMP
4 Побудувати зображення електричних зв'язків	ENTR/WIRE ENTR/BUSB
5 Ввести унікальні схемотехнічні позначки для всіх ЕРЕ	SCMD/PNUM
6 Присвоїти ім'я всім ЕРЕ	NAME/COMP
7 Присвоїти ім'я ланцюгам	NAME/NET
8 Зберегти створену схему ЕЗ	FILE/SAVE

2-й етап. Для створення файлу бази даних ДП необхідно:

а) витягти список ланцюгів (.NLT) зі схеми ЕЗ за допомогою програми PC-NODES;

б) створити файл перехресних засилань (.FIL), що описує відповідність між схемотехнічними і топологічними описами ЕРЕ;

в) виконати упаковку бази даних топології за допомогою програми PC-PACK.

3-й етап. Автоматичне розміщення елементів на полі конструктива ДП за допомогою редактора PC-PLACE. Вхідними даними для PC-PLACE є:

- а) база даних, що упакована в програмі PC-PACK (.PKG);
- б) параметри, які визначаються користувачем (сітка, у вузлах якої розміщуються EPE, параметри асоціювання, які визначають позиції дискретних EPE відносно основних елементів, відстань між основними EPE, положення фіксованих EPE, заборонені зони для розміщення).

Вихідними даними є:

- а) файл бази даних з розміщенням EPE (.PLC);
- б) файли звітів про початкове розміщення (.PLR), гістограму (.HIS), переміщення (.RPT) та покращення розміщення EPE;
- в) файл креслення для виведення на плоттер або принтер (.PLT);
- г) файл протоколу команд (.CMD).

Основні кроки з виконання автоматичного розміщення такі:

крок	Команди, що використовуються
1 Задати сітку для розміщення EPE	LATP;
2 Визначити основні і дискретні EPE	LATC;
3 Задати параметри об'єднання (асоціювання) дискретних EPE по відношенню до основних	ASSG;
4 Визначити зазори основних EPE	CLR;
5 Запустити автоматичне розміщення	PLCE;
6 Оцінити отримане розміщення	HIST
7 Покращити автоматичне розміщення	IMPR/COMP IMPR/GATE MOVE, ROT, SWAP.

4-й етап. Автоматичне трасування з'єднань. Вхідними даними для PC-ROUTE є: файл бази даних топології, створений в PC-PLACE та параметри і правила трасування (стратегія трасування), які містяться в файлі з розширенням (.CTL). Програма PC-ROUTE надає можливість вибрати стандартну стратегію (PCAD1.CTL, PCAD2.CTL, PCAD3.CTL) або створити власну. Вибираємо стандартну стратегію трасування PCAD1.CTL.

Перед запуском PC-ROUTE необхідно додати контур ДП в базу даних за допомогою команди DRAW редактора PC-CARDS. Контур плати необхідно створити в режимі DETL на шарі BRDOUT.

Після запуску автоматичного трасування в процесі виконання PC-ROUTE на екрані з'являються повідомлення, що відображають етапи трасування.

У випадку успішного завершення трасування з'являється повідомлення ROUTE COMPLETE. Якщо в процесі виконання були помилки, PC-ROUTE перерве свою роботу і внесе відповідну інформацію в файл повідомлень (файл з розширенням REP)

В результаті отримаємо варіант розміщення елементів на друкованій платі, який наведено на рисунку 4.14, трасування електричних з'єднань – відповідно на рисунках 4.15, 4.16.

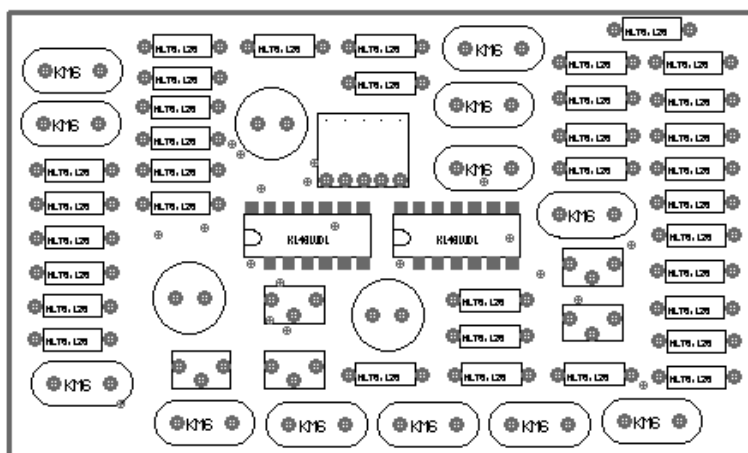


Рисунок 4.14

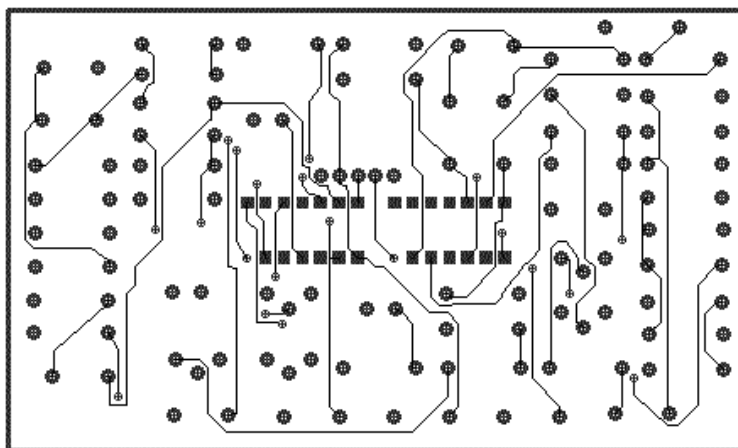


Рисунок 4.15

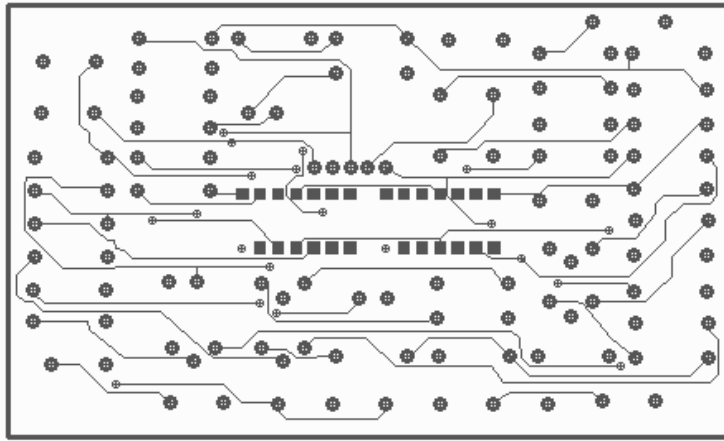


Рисунок 4.16

Завдання 2

В результаті аналізу електромагнітної сумісності модуля на ДП, зображеного на рисунку 4.14 виникла необхідність змінити його компоноване рішення. Резистори змінного опору та з'єднувач необхідно встановити вздовж протилежних широких сторін ДП. Виконати необхідне доопрацювання розміщення та трасування.

Розв'язання

Розв'язання завдання аналогічно до розв'язання завдання 1 за винятком роботи в РС-PLACE. Необхідно зафіксувати резистори змінного опору та з'єднувач. Для цього треба перемістити EPE на визначене місце за допомогою команди MOVE або MOVE/COMP, а потім зафіксувати це положення за допомогою команди FIX (рисунок 4.17).

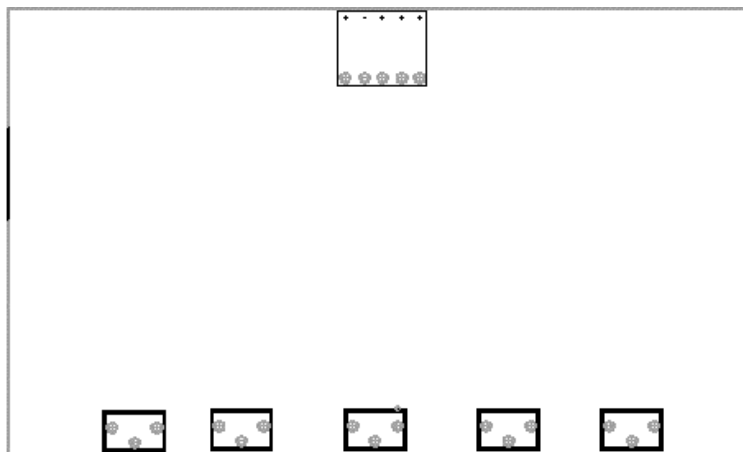


Рисунок 4.17

Після запуску автоматичного розміщення за допомогою команди PLCE отримуємо новий варіант розміщення, який наведено на рисунок 4.18.

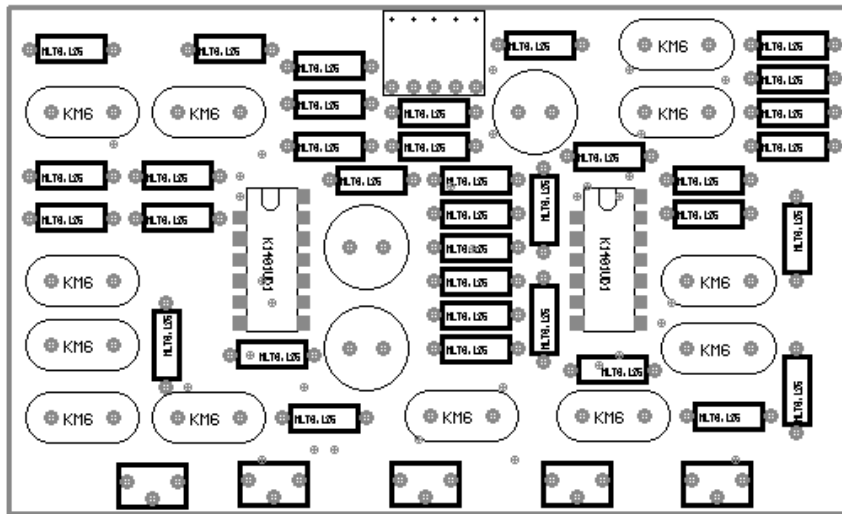


Рисунок 4.18

Використовуючи редактор PC-ROUTE, одержуємо варіант трасування з'єднань, що наведено на рисунках 4.19 та 4.20.

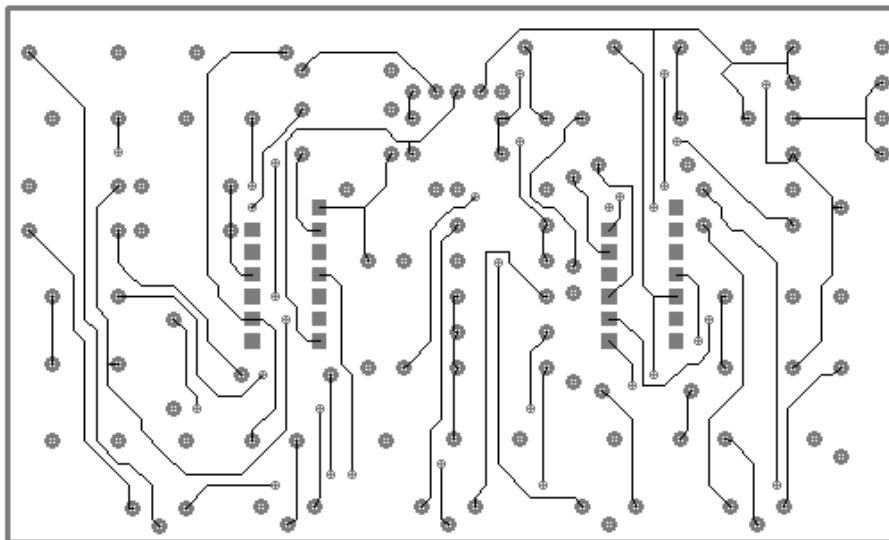


Рисунок 4.19

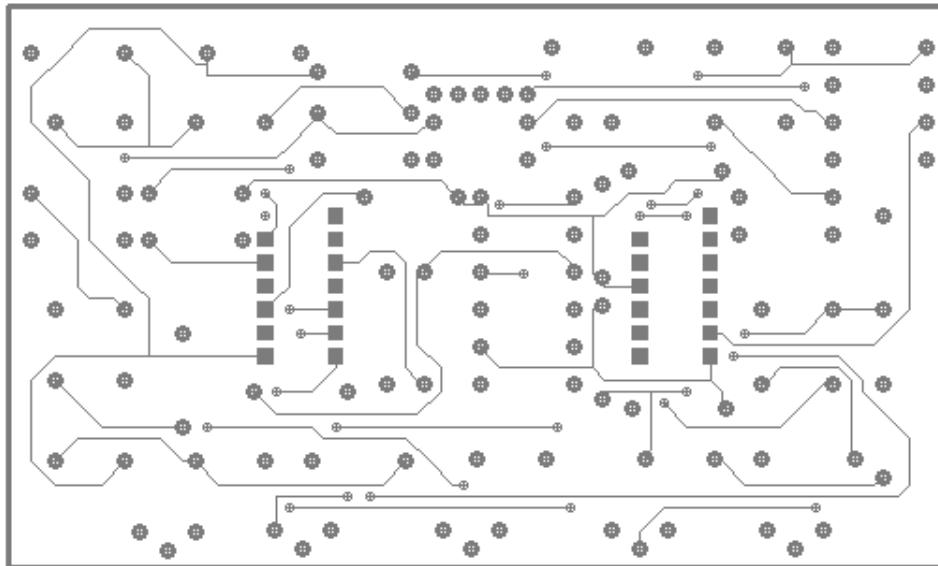


Рисунок 4.20

Завдання 3

Виходячи з технологічних міркувань, виконати розміщення елементів та трасування з'єднань за вимогами мінімуму перехідних отворів для модуля, схема якого зображена на рисунку 4.21.

Розв'язання

Хід розв'язання завдання аналогічний до розв'язання завдання 2 за винятком роботи в PC-ROUTE. При застосуванні стратегії трасування для виконання заданого обмеження необхідно вибрати параметр “Perform via minimization” та встановити опцію Yes. Результат розв'язання буде мати вигляд, зображений на рисунках 4.21, 4.22, з яких слідує, що кількість перехідних отворів зменшилась з 30 до 14.

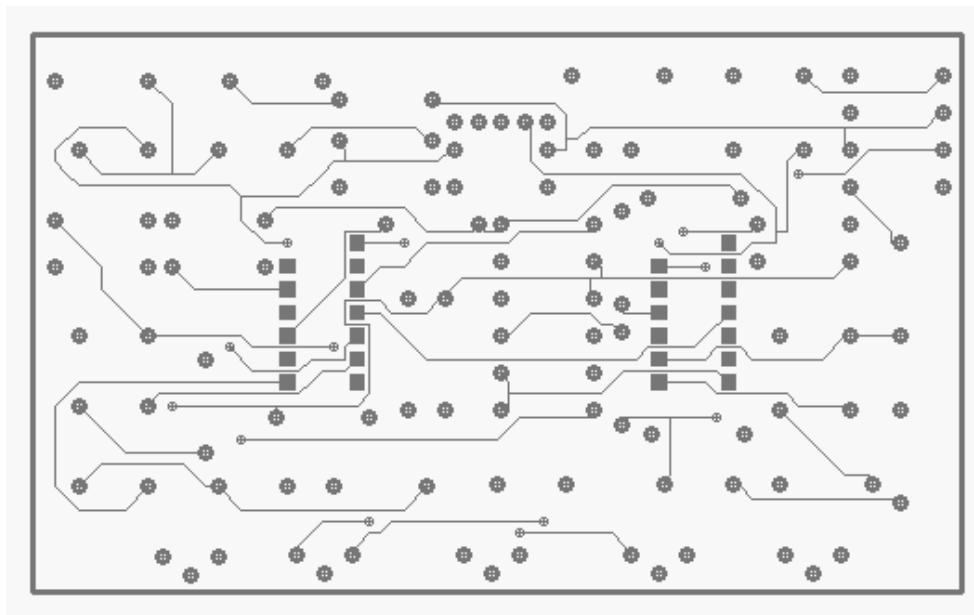
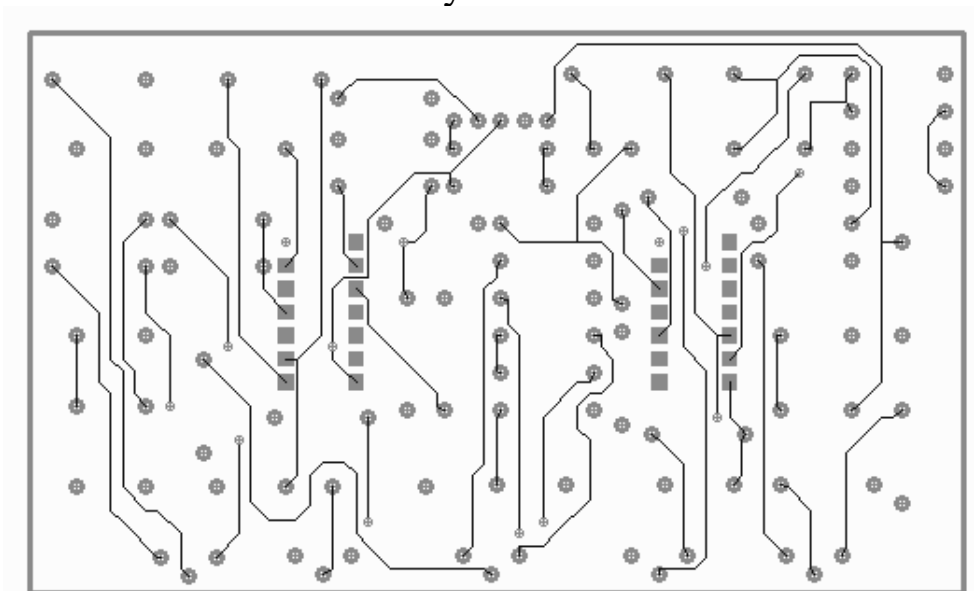


Рисунок 4.21



⊠ – контактні майданчики, ○ – перехідні отвори

Рисунок 4.22

Завдання 4

Використовуючи САПР P-CAD, виконати варіант трасування одношарової плати модуля, схема якого зображена на рисунку 4.23.

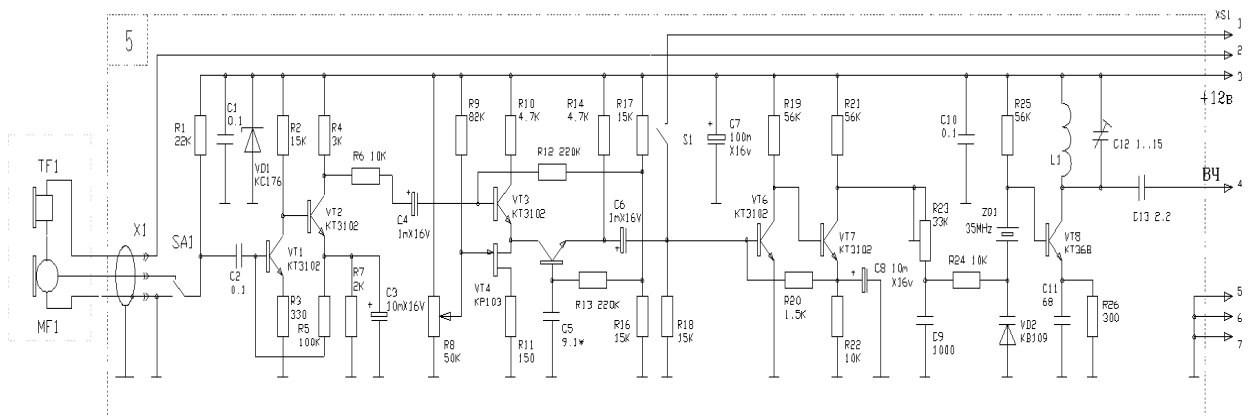


Рисунок 4.23

Розв'язання

Розв'язання завдання зі створення ДП аналогічно розв'язанню завдання 1, за винятком роботи в редакторі PC-ROUTE. Для того, щоб автотрасувальник зробив трасування на одному шарі, необхідно створити власну стратегію трасування. При виборі пункту редагування стратегії розводки головного меню PC-ROUTE з'являється меню редагування стратегії, в якому необхідно вибрати пункт редагування параметрів розводки. Вибираючи параметр "Number of routing layers", необхідно встановити кількість шарів, яка дорівнює 1. В результаті отримано варіант трасування електричних з'єднань, зображений на рисунку 4.24.

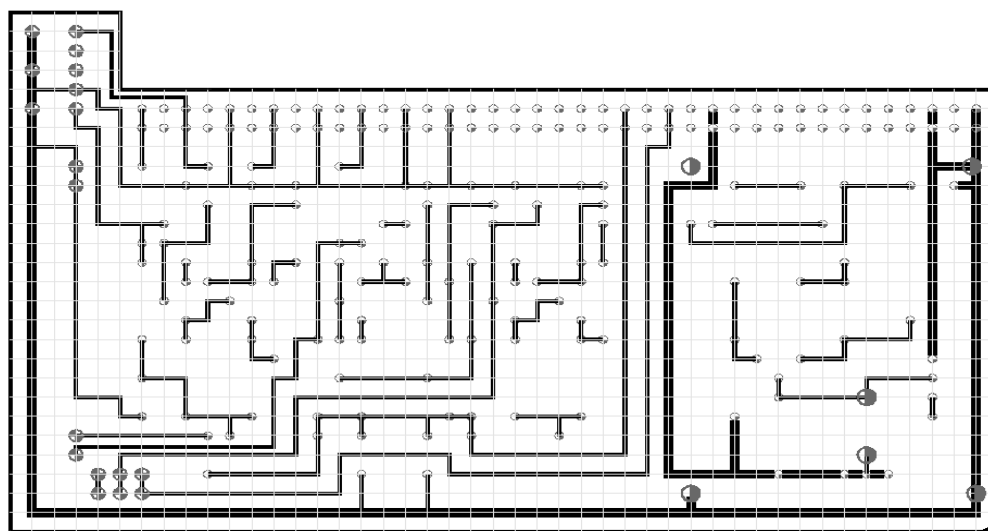


Рисунок 4.24

Індивідуальні практичні завдання

Завдання 1

Виконати варіант автоматичного розміщення елементів і трасування електричних з'єднань модуля плати, схема якого зображена на рисунку 4.25. Створити власну стратегію трасування, в якій кількість шарів дорівнює 1.

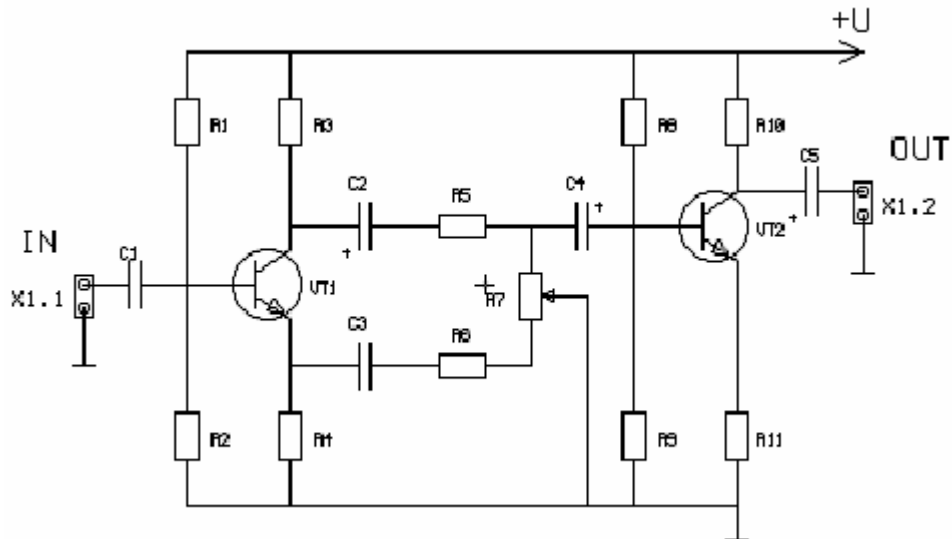


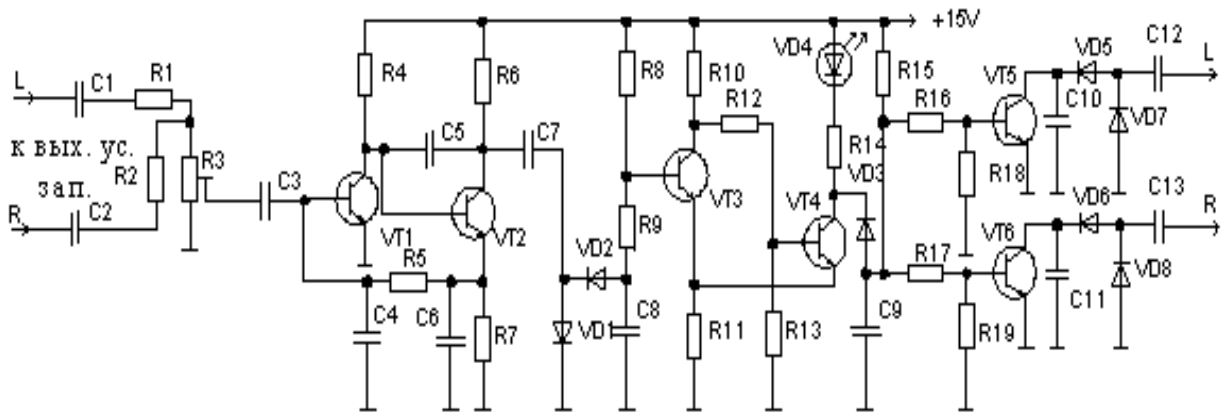
Рисунок 4.25

Завдання 2

Використовуючи вхідні дані завдання 1, виконати в САПР P-CAD варіант розміщення елементів і трасування електричних з'єднань. Розміщення елементів виконати за допомогою редактора PC-CARDS.

Завдання 3

Виконати варіант розміщення елементів на одношаровій платі і трасування електричних з'єднань в САПР P-CAD модуля, що зображений на рисунку 4.26, використовуючи пріоритет вертикального прокладання трас.



Тип елементів: R16-R19- МЛТ-0,125; VT5-VT6-КТ315; C10-C13-КМ5; VD5-VD8-КД523

Рисунок 4.26

Завдання 4

Використовуючи вхідні дані завдання 3, виконати варіант розміщення елементів на платі і трасування з'єднань з горизонтальним пріоритетом прокладання трас.

Завдання 5

Виконати варіант розміщення елементів на платі і трасування електричних з'єднань в САПР P-CAD модуля, що зображений на рисунку 4.27, за вимогами мінімізації перехідних отворів. Крок координатної сітки встановити відповідно відстані між виводами мікросхеми.

Рисунок 4.27

Завдання 6

Використовуючи вхідні дані завдання 5, зробити заміну елемента R2 на підстроювальний резистор (тип елемента вибрати самостійно) та виконати доопрацювання трасування з'єднань на друкованій платі.

Завдання 7

Використовуючи САПР P-CAD, одержати список електричних кіл зі схеми електричної принципової, що зображена на рисунку 4.28.

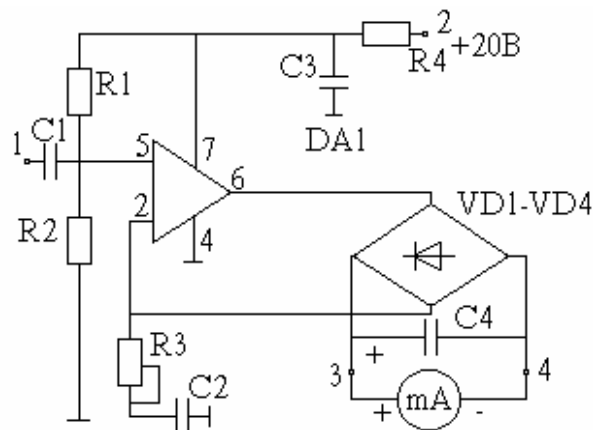


Рисунок 4.28

Завдання 8

Використовуючи файл, в якому наведено об'єднаний вигляд розміщення та трасування з'єднань (рисунок 4.29), вилучити з нього окремий вигляд розміщення елементів на ДП та двох шарів трасування.

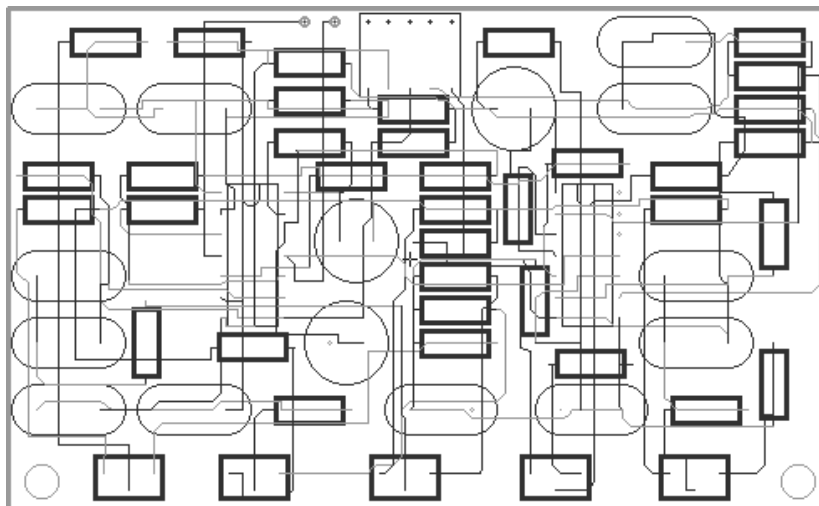


Рисунок 4.29

ВИСНОВКИ

Автоматизація проектування займає особливе місце серед комп'ютерних інформаційних технологій. Оскільки вона - є синтетична дисципліна, її складовими частинами є багато інших сучасних інформаційних технологій. Технічне забезпечення систем автоматизованого проектування засновано на використанні обчислювальних мереж і телекомунікаційних технологій, в автоматизованому проектуванні використовуються персональні комп'ютери й робочі станції, є приклади застосування

мейнфреймів. Математичне забезпечення автоматизованого проектування відрізняється багатством і розмаїтістю використовуваних методів обчислювальної математики, статистики, математичного програмування, дискретної математики, штучного інтелекту. Програмні комплекси автоматизованого проектування відносяться до числа найбільш складних сучасних програмних систем, заснованих на операційних системах Unix, Windows, мовах програмування C, C++, Java та інших, сучасних CASE-технологіях, реляційних і об'єктно-орієнтованих системах керування базами даних (СУБД), стандартах відкритих систем та обміну даними в комп'ютерних середовищах.

Знання основ автоматизації проектування та уміння працювати із засобами автоматизованого проектування потрібно практично будь-якому інженерові-розроблювачеві. Комп'ютерами насичені проектні підрозділи, конструкторські бюро та офіси.

Сучасні програми автоматизованого проектування призначені не тільки для створення креслень – вони володіють набагато ширшими можливостями та використовуються практично у всіх галузях техніки. З їхньою допомогою можна створювати моделі різних конструкцій і перевіряти властивості моделі на комп'ютері до початку виробництва, що істотно знижує кількість помилок проектування та прискорює появу виробу на ринку.

СПИСОК ЛІТЕРАТУРИ

1 Норенков И.П. Основы автоматизированного проектирования: Учебник. – М.: Издательство МГТУ им. Н.Э. Баумана, 2002.

2 Болдин А.Н., Задиранов А.Н. Основы автоматизированного проектирования: Учеб. пособие. – М: МГИУ, 2006. – 104 с.

3 Ли К. Основы САПР (CAD/CAM/CAE). – С.Пб.: Питер, 2004. – 560 с.

4 Комп'ютерні технології автоматизованого виробництва: Навч. посібник / М.А. Бережна. – Харків: ТОВ «Компанія СМІТ», 2007. – 368 с.

5 Компьютерное проектирование и подготовка производства конструкций: Учеб. пособие для вузов / С.А. Куркин, В.М. Ховов, Ю.Н. Аксенов и др.; Под ред. С.А. Куркина, В.М. Ховова. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. – 464 с.

6 Автоматизация проектирования радиоэлектронных средств: Учеб. пособие для вузов / под ред. О.В. Алексеева. — М.: Высш. шк. 2000.— 479 с.

7 Норенков И.П. Разработка систем автоматизированного проектирования. – М.: Изд-во МГТУ им. Н.Э.Баумана, 1994. – 207 с.

8 Быков В.П. Методическое обеспечение САПР в машиностроении. – Л.: Машиностроение, 1989. – 255 с.

9 САПР технологических процессов, режущих инструментов и приспособлений; Под. ред. С.Н. Корчака, – М.: Машиностроение, 1988. – 350 с.

10 Жермен-Лакур П., Жорж П.Л., Пистр Ф., Безье П. Математика и САПР: В 2-х кн. – Кн. 2 / Пер. с франц. – М.: Мир, 1989. – 264 с.