

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІНЖЕНЕРНИЙ НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ім. Ю.М. ПОТЕБНИ  
КАФЕДРА ЕЛЕКТРОНІКИ, ІНФОРМАЦІЙНИХ СИСТЕМ  
ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

**ЕЛЕКТРООБЛАДНАННЯ ВЕРСТАТІВ**  
**З ЧИСЛОВИМ ПРОГРАМНИМ КЕРУВАННЯМ**

Конспект лекцій  
для здобувачів ступеня вищої освіти бакалавра  
спеціальності 171 «Електроніка»  
освітньо-професійної програми «Електроніка»

Затверджено  
вченою радою ЗНУ  
Протокол №\_\_  
від \_\_\_\_\_р.

2022 рік

---

## ЗМІСТ

Вступ.....	6
1. Загальні відомості .....	7
1.1. Історія винаходу верстата з ЧПК .....	10
1.2. Основні сучасні системи числового програмного керування	11
1.3. Органи керування сучасного верстата з ЧПК.....	12
1.4. Основні режими роботи верстата з ЧПК.....	15
Запитання до розділу 1 .....	17
2. Підсистема керування верстата з ЧПК .....	18
2.1. Складові підсистеми керування .....	18
2.2. Класифікація систем ЧПК.....	21
2.3. Система ЧПК фірми Fanuc.....	22
2.4. Система ЧПК фірми Siemens .....	23
2.5. Системи ЧПК фірми Heidenhain.....	24
2.6. Система ЧПК фірми Haas.....	27
Запитання до розділу 2 .....	29
3. Підсистема приводів.....	30
3.1. Кроковий двигун із змінним магнітним опором .....	30
3.2. Кроковий двигун із постійним магнітним опором.....	31
3.3. Сервоприводи в сучасних високоточних верстатах з ЧПК...	32
3.4. Синхронні серводвигуни.....	33
3.5. Асинхронні серводвигуни.....	35
3.6. Ходовий гвинт .....	36
Запитання до розділу 3 .....	39
4. Підсистема зворотного зв'язку.....	40
4.1. Датчики лінійних переміщень .....	42
4.2. Датчики кутових переміщень .....	51
Запитання до розділу 4 .....	56

---

5. Функціонування системи числового програмного керування .	57
5.1. Система координат верстата з ЧПК .....	59
Запитання до розділу 5 .....	63
6. Нульова точка верстата та заготовки, напрямки переміщень...	64
6.1. Нульова точка верстата .....	64
6.2. Нульова точка деталі та робоча система координат .....	64
6.3. Компенсація довжини інструменту .....	67
6.4. Визначення робочої системи координат .....	68
6.5. Контактний спосіб вимірювання інструменту .....	70
6.6. Безконтактний спосіб вимірювання інструменту .....	77
Запитання до розділу 6 .....	80
7. Основи програмування обробки на верстатах з числовим програмним керуванням.....	81
Запитання до розділу 7 .....	85
8. Етапи створення керуючої програми.....	86
8.1. Абсолютні та відносні координати .....	86
8.2. Створення керуючої програми на ПК.....	88
8.3. Методи перевірки керуючої програми .....	89
Запитання до розділу 8 .....	93
9. Структура керуючої програми.....	94
9.1. Підготовчі та допоміжні коди.....	94
9.2. Слово даних, адреса та число .....	98
9.3. Рядок безпеки .....	100
9.4. Формат програми .....	101
Запитання до розділу 9 .....	107
10. Особливості використання базових кодів та постійних циклів.....	108
10.1. G-коди .....	108

---

10.2. Допоміжні коди – М-коди.....	116
Запитання до розділу 10 .....	124
11. Постійні цикли верстата з ЧПК.....	126
11.3. Цикли розточування .....	136
11.4. Приклад програми на свердління з допомогою постійних циклів.....	137
Запитання до розділу 11 .....	139
12. Автоматична корекція радіусу інструмента .....	141
Запитання до розділу 12 .....	148
13. Підпрограми .....	150
Список використаної літератури.....	157

---

## Вступ

Одним із найважливіших засобів підвищення продуктивності праці, забезпечення повторюваності випуску якісних виробів є автоматизація виробничих процесів на основі використання верстатів з числовим програмним керуванням (ЧПК).

Головна відмінність цього обладнання в тому, що оброблювання виконується за розробленою програмою, що дозволяє забезпечувати гнучкість виробничого процесу і можливість використовувати програму багаторазово.

На сьогоднішній день верстати з ЧПК знаходять широке застосування як на великих підприємствах так і на малих, що вимагає від фахівця розуміння будови, можливостей цих верстатів та вміння використовувати це обладнання з максимальною ефективністю.

Запропонований посібник створений відповідно до лекційних курсів «Інноваційне обладнання автоматизованого виробництва – Верстати з ЧПК» та «Обладнання автоматизованого виробництва», що викладаються на кафедрі «Конструювання машин» в НТУУ «Київський політехнічний інститут імені Ігоря Сікорського».

В посібнику представлені матеріали по історії створення верстатів з ЧПК, конструктивні особливості та системи програмного керування, що використовуються в сучасних верстатах з ЧПК, детально розглядаються основні підсистеми: підсистема керування, підсистема приводів, підсистема зворотного зв'язку та засоби їх ефективного функціонування.

Також велика увага приділена функціонуванню системи програмного керування, послідовності та особливостей створення програм оброблювання, їх структури та особливостей реалізації на верстатах з ЧПК.

Головна мета посібника – розширити знання та вміння майбутніх фахівців в області створення та використання сучасних верстатів з ЧПК, а також освоїти систему програмування з використанням G та M кодів відповідно до положень Міжнародної організації по стандартизації (ISO) та Асоціації електронної промисловості (EIA).

---

## 1. Загальні відомості

На сьогоднішній день практично кожне підприємство використовує верстати з числовим програмним керуванням.

Переваги верстатів з числовим програмним керуванням у порівнянні з універсальними наступні:

Верстати з числовим програмним керуванням **можуть працювати практично автономно**, день за днем випускаючи продукцію стабільно високої якості.

Один оператор може обслуговувати декілька верстатів, оскільки його участь у виготовленні деталі мінімальна і зводиться до запуску програми та реалізації допоміжних операцій - установка заготовки і зняття готової деталі, налагодження інструментів і таке інше, що забезпечує високий рівень автоматизації виробництва.

3. Наступною перевагою є **виробнича гнучкість**. Для обробки різних деталей необхідно лише замінити програму, а вже перевірена і відпрацьована програма може бути використана в любий момент і необмежену кількість разів.

4. Четвертою перевагою є **висока точність та повторюваність обробки**. Використовуючи одну і ту ж програму можна виготовити тисячі практично ідентичних деталей

5. Числове програмне керування дозволяє оброблювати такі деталі, які неможливо виготовляти на звичайному обладнанні. Це **деталі складної просторової форми**, наприклад, штампи і прес-форми, лопатки газотурбінних двигунів і таке інше.

6. Сама методика роботи за програмою дозволяє достатньо точно визначати час обробки конкретної партії деталей і, відповідно, більш повно завантажувати обладнання.

Верстати з числовим програмним керуванням достатньо дорогі, але їх висока продуктивність і стабільна якість продукції перебивають усі затрати при грамотному використанні та відповідних об'ємах виробництва.

Конструкція верстатів з ЧПК повинна забезпечувати, як правило, поєднання різних видів обробки (точіння-фрезерування; фрезерування-свердління-нарізання різьби, наприклад), автоматичну заміну інструментів, зручність завантаження заготовок та вивантаження готових деталей, Останнє особливо важливе при використанні промислових роботів,

Сучасні верстати з ЧПК та панель керування ЧПК Siemens Sinumeric показані на наступних чотирьох рисунках (рис 1.1; рис1.2; рис.1.3; рис.1.4).



Рис. 1.1. Токарный верстат з ЧПК



Рис.1.2. Токарно-фрезерний верстат з ЧПК



Рис.1.3. Фрезерно-свердильний верстат з ЧПК



Рис.1.4. Панель керування ЧПК Siemens Sinumeric

Підвищення точності обробки досягається високою точністю виготовлення та жорсткістю верстата, що вища ніж у звичайних верстатів такого ж призначення. Для цього використовують короткі кінематичні ланцюги, скорочуючи кількість механічних передач та автономні приводи виконавчих механізмів.

Підвищенню точності сприяє також усунення зазорів в передавальних механізмах приводів подач, зменшення втрат на тертя в



---

напрямних та інших механізмах, підвищення вібростійкості, зниження теплових деформацій, використання датчиків зворотного зв'язку.

Базові деталі (станини, колони) виконують більш жорсткими за рахунок додаткових ребер жорсткості та використовуючи матеріали підвищеної вібростійкості.

Шпинделі виконують точними, жорсткими з підвищеною зносостійкістю базових поверхонь.

Конструкція шпинделя значно ускладнюється через вбудовані пристрої автоматичного затиску та розтискування заготовок, використання датчиків при адаптивному керуванні та автоматичній діагностиці.

Допоміжні механізми для заміни інструментів, прибирання стружки, змащування, затискні пристосування також мають специфічні відмінності.

Так, пристрої для заміни інструментів (магазини, автооператори, револьверні головки) повинні забезпечувати мінімальні втрати часу на заміну інструментів, високу надійність, стабільність положення інструментів, тобто постійність величини вильоту і положення осі інструменту при повторних замінах. В сучасних верстатах час заміни інструментів визначається декількома секундами.

Напрямні верстатів з ЧПК мають підвищену зносостійкість і малу силу тертя, що дозволяє знизити потужність слідкувального приводу і підвищити точність переміщень.

Інтенсивний розвиток мікропроцесорної техніки дозволяє застосовувати цифрові приводи головного руху та подач з повним мікропроцесорним керуванням.

Конструктивно перетворювачі частоти, сервоприводи та пристрої пуску та реверсу є окремими електронними блоками керування.

Декілька верстатів з числовим програмним керуванням можуть об'єднуватись в гнучку автоматизовану виробничу систему (ГВС), яка може входити в роботизований технологічний комплекс (РТК), або в склад гнучкої автоматичної лінії (ГАЛ) дільниці або цеху.

## **1.1. Історія винаходу верстата з ЧПК**

Винахідником першого верстата з числовим програмним керуванням є американець **Джон Персонс**, що працював у фірмі свого батька **Parsons Inc.** Фірма випускала гвинти для гвинтокрилів і Джон Персонс вперше запропонував використовувати верстат, що працював за програмою, яка вводилась з перфокарт.

---

В **1949** році військово-повітряні сили США профінансували розробку цією фірмою верстата для контурного фрезерування складних за формою деталей авіаційної техніки. Самостійно виконати таке складне завдання фірма не мала можливості, тому звернулась за допомогою до Масачусетського технологічного університету. Це співробітництво продовжувалось недовго. Вже в **1950** році університет викупив компанію по виробництву фрезерних верстатів **Hydro-Tel** та відмовився від співробітництва з **Parsons Inc.** Університет заключив самостійний контракт з військово-повітряними силами США на створення фрезерного верстата з програмним керуванням і вже у вересні **1952** року верстат, що управлявся від перфострічки, вперше продемонстрували широкому загалу. Все ж він був досить складний і не міг використовуватись у виробничих умовах.

Перший серійний пристрій з числовим програмним керуванням був створений у **1954** році, а з **1955** року почав встановлюватись на верстатах.

Широке впровадження верстатів з числовим програмним керуванням проходило дуже повільно через недовіру виробників. Міністерство оборони США змушене було випустити **120** верстатів за свої кошти і передати їх в аренду приватним компаніям.

Першими вітчизняними верстатами з числовим програмним керуванням для промислового використання були токарно-гвинторізний верстат **1К62ПУ** та токарно-карусельний **1541П**. Вони були створені в першій половині 60-х років. Потім були розроблені вертикально-фрезерні верстати на базі верстата мод. **6Н13** з системою програмного керування «**Контур-3П**». В наступні роки найбільше поширення для токарних верстатів отримали системи числового програмного керування вітчизняного виробництва **2Р22** та «**Електроніка НЦ-31**».

## **1.2. Основні сучасні системи числового програмного керування**

Сучасні системи числового програмного керування, що позначаються як **CNC (Computer numerical control)**, засновані на мікропроцесорі з оперативною пам'яттю та операційною системою.

Програма для верстатів з числовим програмним керуванням може завантажуватись із зовнішніх носіїв, або звичайних чи спеціалізованих флеш-накопичувачів. До того ж сучасне обладнання та верстати з числовим програмним керуванням можуть підключатись до комп'ютерних

---

мереж підприємства, що дозволяє виконувати завантаження програми за допомогою передачі по промисловій мережі.

Основна мова програмування описана документом **ISO6983 Міжнародного комітету стандартів** і називається «**G-код**». Вона найбільш поширена для металообробного обладнання. Програмування виконується в **G та M-кодах**.

Мова G та M-кодів базується на положеннях Міжнародної організації по стандартизації (ISO) та Асоціації електронної промисловості (EIA).

Офіційно ця мова вважається стандартом для американських та європейських виробників обладнання з ЧПК і інколи її називають «ISO 7біт».

Все ж виробники систем ЧПК хоча і притримуються цих стандартів для описування основних функцій, але допускають і відхилення від правил, коли мова йде про якісь спеціальні можливості своїх систем.

Системи ЧПК Fanuc (Японія) були одними з перших, що адаптовані для роботи з G та M-кодами ISO. Вони найбільш повно використовують цей стандарт.

Системи Fanuc на сьогоднішній день дуже популярні та найбільш поширені.

Стійки ЧПК інших відомих виробників, наприклад, Heidenhain та Sinumerik (Siemens) також мають можливості роботи з G та M-кодами, але деякі коди можуть відрізнитись.

Немає необхідності знати усі коди всіх систем ЧПК.

Достатньо знати набір основних G та M-кодів, а при виникненні специфічних функцій можна скористатись документацією конкретної системи.

Деякі виробники систем ЧПК пропонують діалогову мову програмування. Ця мова спрощує спілкування з системою, оскільки основою для неї слугують англійські речення, скорочення, запитання та графічні елементи, які вводяться оператором верстата в інтерактивному режимі.

### **1.3. Органи керування сучасного верстата з ЧПК**

Більшість органів керування сучасного верстата з ЧПК зосереджено на передній панелі стійки ЧПК.

Стійка ЧПК має клавіатуру або аналогічну клавіатурі звичайного персонального комп'ютера, або обмежену, яка дозволяє вводити лише основні символи та знаки програмування.

Дисплей монохромний або кольоровий.

Всі клавіші, перемикачі та рукоятки верстата можна умовно розділити на декілька функціональних груп:

**Клавіші для введення різних символів, букв та цифр** дають можливість оператору створити програму обробки прямо на екрані, вводячи Gкоди, різні слова даних та спеціальні символи програмування (наприклад, знак кінця кадру).

У випадку обмеженої клавіатури одна клавіша може відповідати за декілька символів.



	Привод ввімкнено		Привод вимкнено
	Пуск ЧПК		Зупинка ЧПК
	Блокування шпинделя		Подача ЗОР вручну
	Подача ЗОР через систему ЧПК		Блокування заміни інструменту
	Заміна інструменту		Шпиндель ввімкнуто. Обертання вправо
	Зупинка шпинделя		Напрямок по осі X+.

Рис.1.5. Панель керування та клавіші стійки ЧПК фрезерно-свердлильного верстата

**Клавіші редагування** дозволяють оператору змінювати склад керуючої програми.

**Програмні або екранні клавіші** використовуються для виконання різних функцій в залежності від програмного забезпечення системи ЧПК та поточного екранного режиму.

Як правило, ці клавіші розташовані прямо під дисплеєм, а їхні поточні функції відтворюються в нижній частині дисплея.

**Клавіші та перемикачі режимів роботи верстата** дозволяють переходити із одного режиму в інший.

**Кнопки прямого керування осьовими переміщеннями** дозволяють оператору переміщувати виконавчі органи на робочій або прискореній подачі.

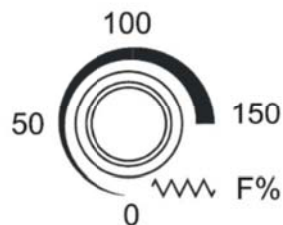


Рис. 1.6. - Ручна корекція величини подачі

Рукоятки керування подачею та обертанням шпинделя

Багато верстатів мають засоби для прямого (без програмування G та M кодів) вмикання та вимикання шпинделя та керування частотою його обертання.

Система ЧПК дає можливість оператору коректувати запрограмованою подачею та частотою обертання шпинделя у визначених діапазонах.

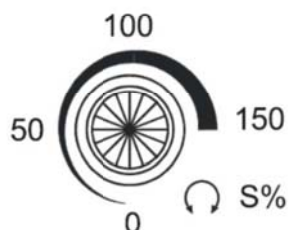


Рис.1.7. Ручна корекція частоти обертання шпинделя

**Клавіші та перемикачі для роботи із спеціальними функціями верстата** відповідають за вмикання та вимикання освітлення робочої зони верстата, керування системою видалення стружки та інші допоміжні дії.

**Клавіші циклу програмування**

За пуск керуючої програми відповідає кнопка «Старт циклу», а за її зупинку – кнопка «Зупинка подачі» або «Скидання програми». До цієї групи відносяться також клавіші для активації функцій вибіркової зупинки

«M01», пропуску кадру «/», виконання програми за окремими кадрами, пробного прогону та блокування осьових переміщень.

Частина органів керування може бути розташована не на самому пульті пристрою ЧПК. Наприклад, вимикач електроживлення часто розташовується на тильній стороні корпусу верстата, а клавіші керування інструментальним магазином рядом з віконцем для завантаження інструментів. Велика червона кнопка «Екстренна зупинка» знаходиться на самому видному та доступному місці.

Практично усі верстати з ЧПК мають маховики, які дозволяють оператору переміщувати виконавчі органи вручну. Як правило, цими маховиками оператор користується для виконання точних операцій, таких, як пошук нульової точки або вимірювання довжини інструменту.

Крім різноманітних органів керування верстат з ЧПК має набір індикаторів (світлодіоди або лампочки), які можуть показати наступне: прийшли чи не прийшли виконавчі органи в нульову точку, чи ввімкнена подача ЗОР. Вони також сигналізують у випадку аварійної ситуації.

#### **1.4. Основні режими роботи верстата з ЧПК**

**Режим автоматичного керування** є основним для верстата з ЧПК. В цьому режимі виконується оброблювання деталі за програмою. Для запуску керуючої програми на її виконання необхідно спочатку вибрати активну програму, а потім натиснути кнопку «Старт циклу».

В режимі автоматичного керування оператор може впливати на запрограмовану величину подачі та частоту обертання шпинделя. Рукоятка корекції прискореного ходу дозволяє змінювати швидкість холостих ходів виконавчих органів верстата в діапазоні від 0 до 150 відсотків.

**Режим редагування** дозволяє оператору вводити нову або редагувати існуючу програму оброблювання вручну, використовуючи клавіатуру пристрою ЧПК.

Можливості редагування керуючої програми у різних стійок ЧПК можуть суттєво відрізнитись. Найпростіші системи дозволяють вставляти, видаляти та копіювати слова даних. Сучасні системи ЧПК мають функції пошуку та заміни даних (аналогічно текстовим редакторам на ПК), копіювання, видалення та переносу визначеного програмного діапазону, здатні редагувати керуючу програму у фоновому режимі.

Функція фонового редагування даних дозволяє оператору верстата створювати або редагувати одну програму при одночасному виконанні

іншої програми. Для фонового редагування систему керування необхідно переключити в автоматичний режим.

Зазвичай в режимі редагування виконується введення або виведення керуючої програми з персонального комп'ютера або іншого зовнішнього пристрою. Тут можна також перевірити розмір вільної пам'яті системи ЧПК та кількість зареєстрованих програм.

**Режим ручного введення даних MDI** дозволяє оператору ввести та виконати один або декілька кадрів, не записаних в пам'яті системи ЧПК. Цим режимом користуються для введення даних **G** та **M-кодів**, наприклад, для заміни інструменту або вимикання обертів шпинделя. Введені команди та слова даних після виконання або скидання програми видаляються.

**Поштовховий режим (старт-стопний)** забезпечує ручне переміщення виконавчих органів верстата натисканням на відповідні клавіші на панелі верстата.

**Режим повернення в нульову точку** є стандартною процедурою по синхронізації верстата та системи керування при вмиканні верстата.

**Режим прямого числового керування DNC** дозволяє виконувати програму обробки безпосередньо з комп'ютера або іншого зовнішнього пристрою, не записуючи її в пам'ять системи. Як правило, в цьому режимі виконуються керуючі програми великого розміру, для реалізації яких недостатньо пам'яті системи ЧПК.

**Режим редагування параметрів** дозволяє виконувати редагування параметрів системи ЧПК.

Параметри користувачів відповідають за настроювання поточної дати та часу, роботу в різних режимах і т.п.

Системні параметри відповідають за функціонування верстата в цілому.

Все ж таки не рекомендується самотійно змінювати значення системних параметрів. Інколи вхід в область параметрів заблокований і для редагування потрібно ввести спеціальний код, що встановлюється виробником.

**Тестові режими** з відповідною кількістю тестових функцій є в кожному верстаті з ЧПК. До них відносять, наприклад, пробний прогін та відпрацювання керуючої програми за окремими кадрами. Деякі верстати дозволяють проводити графічну перевірку траєкторії переміщення інструменту.

**Індикація системи координат** дає можливість під час виконання обробки за програмою, або ручного переміщення виконавчих органів

верстата, спостерігати за їхнім поточним положенням в різних координатних системах.

Таблиця 1.1

Регістри положення виконавчих органів верстата

<b>ABSOLUTE</b>	Абсолютне положення в робочій системі координат ( <b>G54-G59</b> )
<b>MACHINE</b>	Поточне положення відносно нуля верстата
<b>DISTANCE TO GO</b>	Залишок відстані переміщення в кадрі

За координатами реєстрів ABSOLUTE та MACHINE (табл. 1.1) можна судити про відповідність переміщень до керуючої програми.

Координати реєстра DISTANCE TO GO використовуються оператором для знаходження нуля деталі та установки робочої системи координат.

Для більшості сучасних верстатів система числового програмного керування може бути умовно розділена на три підсистеми:

- підсистема керування;
- підсистема приводів;
- підсистема зворотного зв'язку.

## Запитання до розділу 1

1. Які переваги верстатів з ЧПК у порівнянні з універсальними?
2. За рахунок яких заходів забезпечується більша точність верстатів з ЧПК в порівнянні з універсальними?
3. Історія винаходу верстата з числовим програмним керуванням.
4. Яка основна мова програмування використовується в сучасних верстатах з ЧПК?
5. Які міжнародні фірми є основними виробниками систем ЧПК?
6. Назвіть основні органи керування верстатом з ЧПК, що зосереджені на передній панелі стійки ЧПК.
7. Назвіть основні режими роботи верстата з ЧПК.
8. Охарактеризувати режим автоматичного керування верстатом з ЧПК.
9. Особливості режиму редагування та його можливості.
10. З яких підсистем складається система числового програмного керування?



## 2. Підсистема керування верстата з ЧПК

Підсистема керування верстата з ЧПК є центральною частиною усієї системи числового програмного керування.

З однієї сторони вона читає керуючу програму та віддає команди різноманітним агрегатам верстата на виконання тих чи інших операцій. З іншої сторони взаємодіє з людиною-оператором, дозволяючи контролювати процес обробки.

Системи керування можуть бути закритими та відкритими, так званими ПК-сумісними.

Закриті системи керування мають власні алгоритми та цикли роботи, власну логіку. Виробники таких систем, як правило, не розповсюджують інформацію про їхню архітектуру. Тому самостійно оновлювати програмне забезпечення та редагувати налаштування такої системи практично неможливо.

Але в системах закритого типу є важлива перевага – висока надійність, оскільки усі компоненти системи проходять тестування на сумісність.

ПК-сумісні системи керування більш відкриті. Їх апаратні складові такі ж, як і у домашнього персонального комп'ютера. Перевага такого методу – в доступності та відносно невеликій вартості електронних компонентів, та їх доступності. Все ж вважається, що надійність таких систем нижча ніж закритих систем керування.

### 2.1. Складові підсистеми керування

Серцем підсистеми керування є контролер або процесор, який найчастіше розташовується в корпусі стійки з ЧПК, що має клавіатуру та монітор для вводу і виводу необхідної інформації.

**Контролер** – комп'ютеризований пристрій, що вирішує наступні задачі:

- формування траєкторії руху різального інструменту;
- реалізацію технологічних команд керування пристроями автоматика верстата;
- загальне керування;
- редагування керуючих програм;
- діагностику та допоміжні розрахунки, наприклад, режимів різання і інше.

В якості контролера використовують мікропроцесор, на якому побудована система або логічний контролер, що програмується, чи більш складну систему – промисловий комп'ютер.

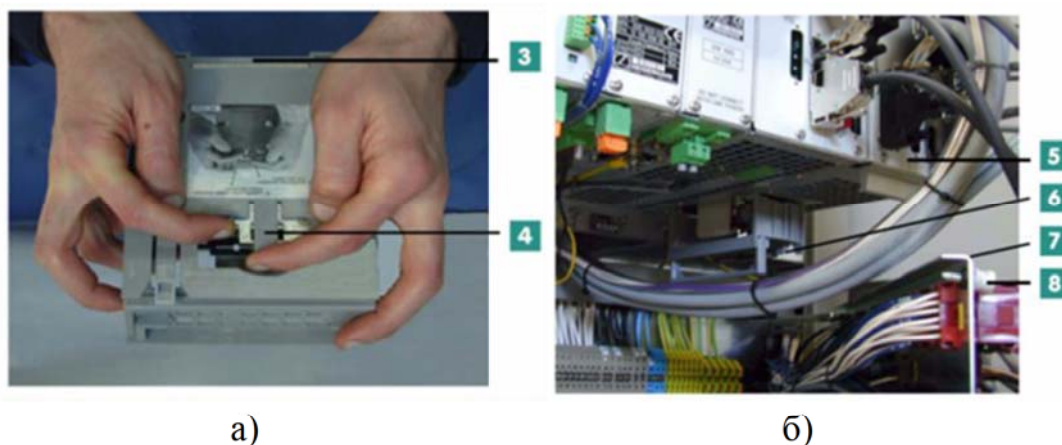


Рис.2.1. Основний контролер фрезерно-свердлильного верстата:  
а)- жорсткий диск в корпусі (3 – штекер; 4 – запобіжна пластина);  
б) – вид на логічний пристрій (5 – головний контролер; 6 - запобіжна пластина; 7- плата штекерного роз'єму; 8 – гвинт кріплення)

Структурно до складу ЧПК також входять **постійна пам'ять** (ПЗП) у вигляді карти пам'яті або жорсткого диску для довготривалого зберігання інформації (роки і десятки років) та **оперативна пам'ять** (ОЗП) для тимчасового зберігання керуючих та системних програм, що використовуються в даний момент.

Все разом називається інтерфейсом користувача.

Розробка керуючих програм на сьогоднішній день виконується з використанням спеціальних модулів для систем автоматичного проектування (**САПР**) або окремих систем автоматизованого програмування (**CAM**), які у відповідності до електронної моделі генерують програму обробки.

В процесі створення або після введення керуючої програми оператор може відредагувати її, включивши в роботу системну програму редактора та виводячи на дисплей всю або потрібні частини керуючої програми і вносячи потрібні зміни.

Під час роботи в режимі виготовлення деталі керуюча програма кадр за кадром поступає на виконання.

У відповідності до команд керуючої програми контролер викликає із постійної пам'яті відповідні системні підпрограми, які і заставляють працювати підключене до ЧПК обладнання в заданому режимі.

Результат роботи контролера у вигляді електричних сигналів поступає на виконавчий орган – привод подач, головний привод чи пристрій керування автоматикою верстата.

Для визначення потрібної траєкторії переміщення робочого органу у відповідності до керуючої програми використовується **інтерполятор**, що розраховує положення проміжних точок траєкторії за заданими в програмі кінцевими точками.

Завдяки системі керування верстата з ЧПК мають розширені технологічні можливості при збереженні високої надійності.

За характером руху виконавчих органів системи ЧПК розділяють на:

- позиційні;
- контурні;
- синхронні;
- універсальні.

При позиційному керуванні засоби ЧПК забезпечують переміщення виконавчого органу в задану координату. Привод подач повинен забезпечувати високу швидкість переміщення для зменшення часу холостих ходів. Оброблювання при цьому не виконується і вид траєкторії переміщення не задається. Точність вимагається лише при зупинці в заданій координаті.

При контурному керуванні переміщення виконавчих органів виконується по заданій траєкторії з установленою швидкістю для отримання потрібної форми оброблюваної поверхні.

Розрізняють контурні прямокутні системи, контурні криволінійні та синхронні системи ЧПК.

Контурні прямокутні системи забезпечують рух лише по одній координаті для оброблювання поверхні, що паралельна даній вісі. Програмуються кінцеві координати точок переміщення як і в позиційній системі, але вказується швидкість переміщення відповідно до заданих режимів різання. Переміщення виконуються послідовно по кожній із координатних вісей. Прямокутні системи використовують в токарних, фрезерних та шліфувальних верстатах.

Контурні криволінійні системи забезпечують формоутворення при оброблюванні заготовки з одночасним узгодженим рухом виконавчого органу по декількох координатах. Програму переміщення виконавчих органів розробляють відповідно до заданої форми обробленої поверхні та результуючої швидкості, що розрахована для заданих режимів різання. Ці

системи найбільш складні як по створенню програм так і по вимогах до приводу подач.

Синхронні системи використовуються в основному в зубооброблювальних верстатах, де необхідно витримувати постійне співвідношення швидкостей не менше ніж по двох координатних вісях. Формоутворення реалізується завдяки конфігурації інструменту.

Універсальне керування об'єднує принципи позиційного та контурного криволінійного, що дозволяє виконувати позиціонування в задану координату та рух виконавчих органів по відповідній траєкторії. Воно найбільш ефективно для багатоцільових верстатів.

## **2.2. Класифікація систем ЧПК**

В залежності від рівня використання засобів обчислювальної техніки системи ЧПК класифікують наступним чином:

1. Системи типу NC (Numerical Control) виконують адресування команд, інтерполяцію проміжних координат, реалізацію типових циклів за жорстко заданими алгоритмами. Інформація в систему ЧПК вводиться з керуючої програми окремими кадрами.

2. Системи типу MNC (Memory NC) або SNC (Stored NC) оснащуються додатковим блоком оперативної пам'яті, що дозволяє зберігати інформацію про керуючу програму. Програма в пристрій ЧПК вводиться зразу, перевіряється та видається для оброблювання окремими кадрами. Перевага в порівнянні з системою NC – це висока надійність в роботі, оскільки відпадає необхідність в складному фотозчитувальному пристрої для зчитування кожного кадру.

3. Системи типу HNC (Hand NC) дозволяють задавати програму вручну на пульті керування. В цьому випадку відпадає необхідність підготовки керуючої програми технологом-програмістом.

4. Системи типу CNC (Computer NC) – це системи з вбудованими однією або декількома мікропроцесорами та з програмною реалізацією алгоритмів, які записуються в постійний запам'ятовуючий пристрій при виготовленні системи ЧПК. Ці системи дозволяють формувати типові цикли оброблювання для різних технологічних задач. Програмно-математичне забезпечення для реалізації цієї можливості зберігається в запам'ятовуючому пристрої, що постійно перепрограмується.

5. Система DNC (Direct Numerical Control) керує групою верстатів від однієї ЕОМ, що має загальну пам'ять для зберігання програм і

розподілу їх за запитами окремих верстатів. Такі системи використовують в гнучких виробничих системах (ГВС) для організації узгодженої роботи окремих технологічних об'єктів

6. Системи PCNC (Personal Computer NC) будуються на основі персонального комп'ютера в промисловому виконанні. Комп'ютер має спеціальну інтерфейсну плату, що забезпечує зв'язок з приводами, датчиками та електроавтоматикою верстата. Така будова дозволяє легко адаптувати систему ЧПК до верстатів з різним функціональним призначенням за рахунок корекції відповідного програмного забезпечення, що в свою чергу дозволяє виконувати модернізацію застарілих систем ЧПК.

7. Система STEP-NC (покрокова система керування) побудована на основі системи PCNC і її основна задача – виключити участь людини в підготовці процесу оброблювання. До складу програмного забезпечення обов'язково входять пакети CAD, САПР, САМ. Функціонування системи виконується в наступній послідовності:

а) Система CAD забезпечує автоматизацію розроблювання креслення деталі та підготовку геометричної та технологічної інформації для передачі в САПР та системи САМ.

б) Система САПР створює технологічний процес оброблювання (встановлює режими різання, визначає різальний та допоміжний інструменти, послідовність та склад переходів).

в) Система САМ за результатами попередніх кроків виконує розрахунок траєкторії переміщення інструменту, визначає послідовність керування приводами та автоматикою верстата, тобто створює програму керування, яка в подальшому реалізується на верстаті. Розроблюються також системи САМ, що безпосередньо керують системою ЧПК без формування керуючої програми.

Лідерами на світовому ринку в області розробки та поставки високоефективних систем ЧПК для верстатобудування є фірми Fanuc, Siemens Heidenhain, Haas.

### **2.3. Система ЧПК фірми Fanuc**

Системи ЧПК фірми Fanuc (рис. 2.2) розраховані на широкий спектр операцій, таких як точіння, фрезерування, свердління, шліфування та інші. Високі експлуатаційні характеристики та надійність дозволяють обслуговувати найрізноманітніші верстатні системи.

Мініатюрна печатна плата, що вбудована в блок, дозволяє повністю використовувати можливості сучасних великих інтегральних схем та технологій поверхневого монтажу. Плата монтується за рідиннокристалічним дисплеєм. Система ЧПК також має високошвидкісну сервошину послідовної дії та компактний розподільчий модуль вводу-виводу, що дозволяє підключати до одного волоконно-оптичного кабелю декілька сервопідсилювачів.



Рис. 2.2 . Стійка фірми Fanuc

Пристроєм ЧПК та верстатом можна керувати за допомогою графічного інтерфейсу персонального комп'ютера. Крім того, можливості мереж можуть використовуватись для обміну інформацією, а програмні засоби та бази даних для керування сервісними програмами.

Програмне забезпечення (бібліотека CNC) використовується для вводу-виводу внутрішньої інформації ЧПК. Таке програмне забезпечення може підтримувати стандартний набір мов програмування Microsoft (Visual Basic або Visual C++™), а також інтерфейс OLE/DDE.

## 2.4. Система ЧПК фірми Siemens

Системи ЧПК фірми Siemens є високоякісними системами керування для металорізальних верстатів. Це цілісний комплекс взаємодіючих компонентів:

- пристрій ЧПК – центр керування;
- компоненти керування для спілкування оператора з системою;
- контроль електроавтоматики;

- приводи та двигуни в різних виконаннях в залежності від призначення;

- зворотні зв'язки вимірювальних систем;

- силові та сигнальні кабелі і програмне забезпечення.

Фірма Siemens представляє на ринку пристрої ЧПК:

-SINUMERIK 802C, 802S, 802D для токарних та фрезерних верстатів з обмеженою кількістю осей, достатньою для реалізації функціональних можливостей цих верстатів;

-SINUMERIK 810D для верстатів з невеликими робочими зусиллями;

-SINUMERIK 840D – для широкої гами верстатів та технологічних задач.

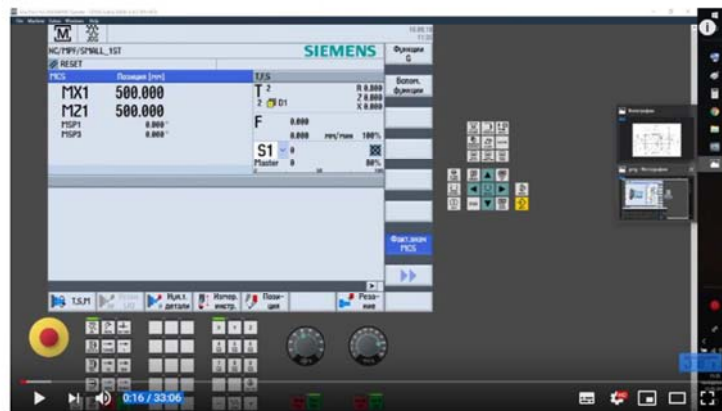


Рис. 2.3. Стійка фірми Siemens

## 2.5. Системи ЧПК фірми Heidenhain

Системи ЧПК фірми Heidenhain TNC через свою багатофункціональність є ідеальними для автоматизованого виробництва. Зокрема, вони є сумісними одна з одною від самої ранньої версії до більш нової системи ЧПК. Завдяки цим якостям вони знаходяться серед всесвітньо визнаних лідерів в галузі виробництва.

Системи ЧПК Heidenhain TNC дозволяють цехове керування та програмування на відстані.

Найбільш широкий набір можливостей є в системах TNC640 та iTNC530. Вони ідеальні для автоматизованого виробництва завдяки короткому часу обробки кадру і можуть бути застосовані в безлюдному виробництві.

Система iTNC530 дозволяє програмування вільного контуру (FK), що було недоступно в більш ранніх версіях.

В системі TNC640 з'явилася нова можливість – програмування фрезерної та токарної обробки на одному верстаті або їх комбінацію в

одній програмі. Токарний контур можна задавати стандартними функціями або у вигляді вільного контуру (FK). Поряд із токарними циклами попереднього та остаточного оброблювання можна задавати цикли проточування канавок, нарізання різьби та інше.

Особливістю системи TNC 640 та iTNC530 є опція DXF, за допомогою якої можна відкривати САД-дані та отримувати із них контури безпосередньо в TNC.

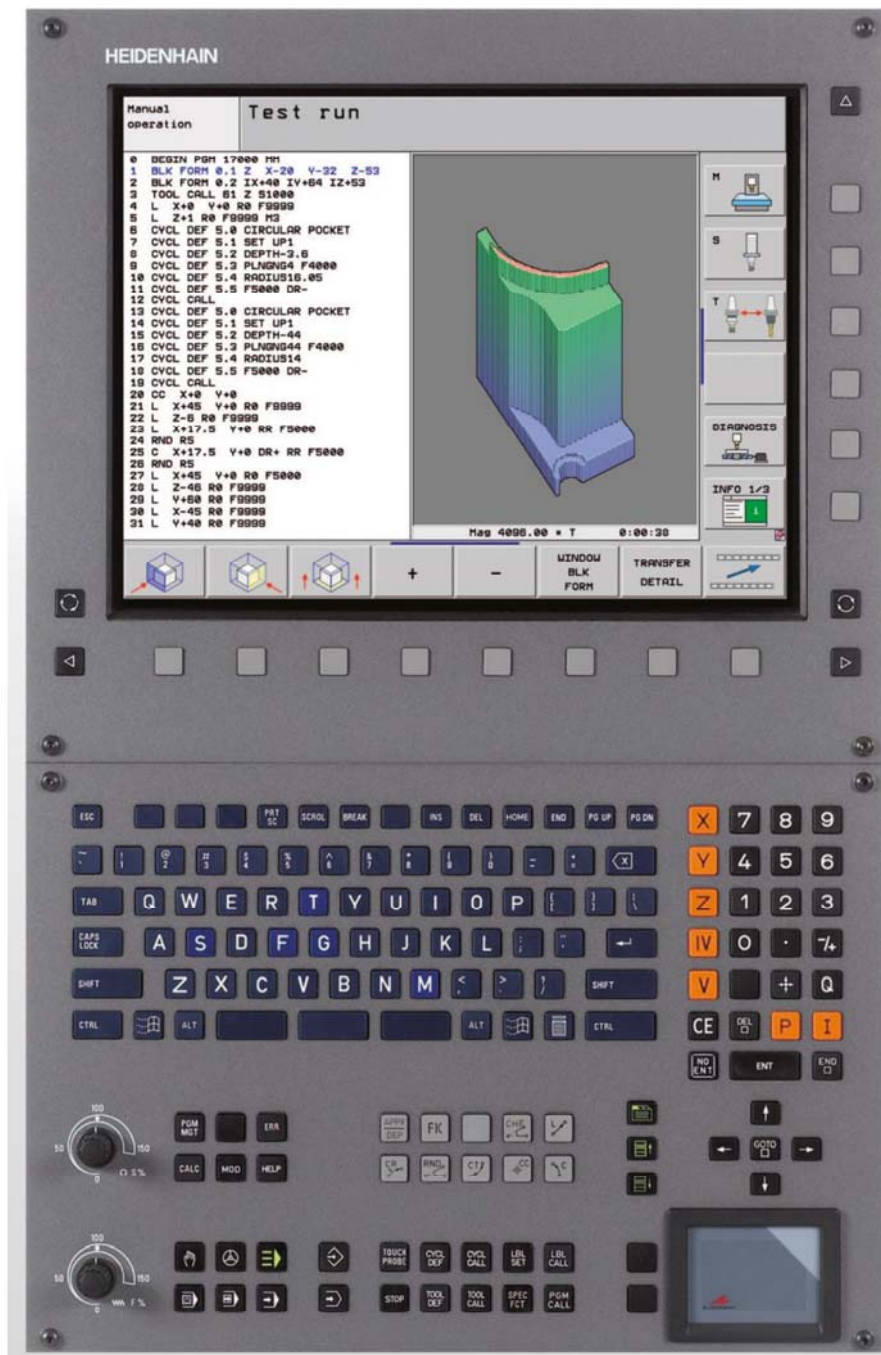


Рис. 2.4. Сійка фірми Heidenhain



Структурно системи керування фірми Heidenhain можна представити наступним чином (рис. 2.5):

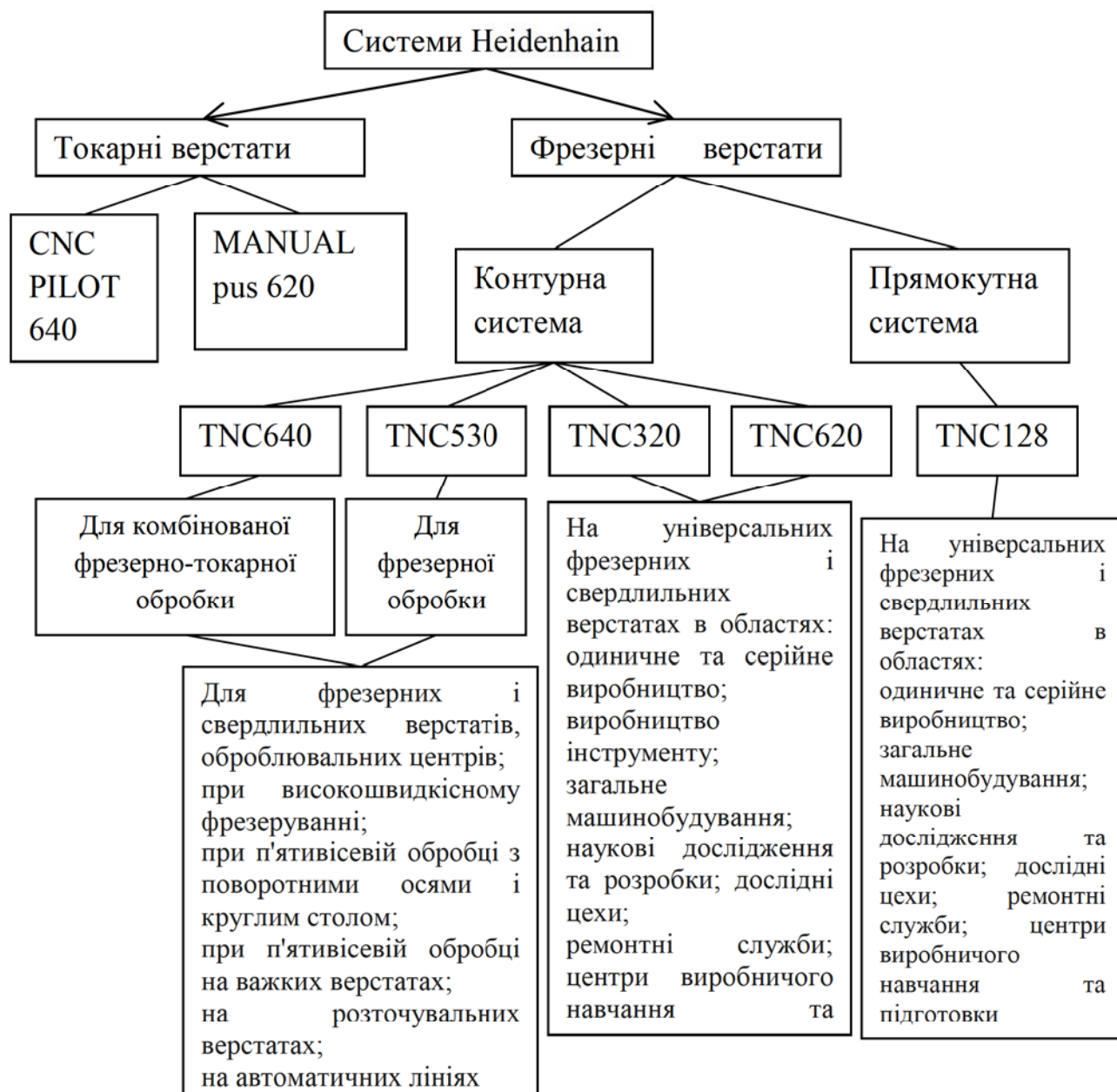


Рис. 2.5 . Структурна схема використання програмного забезпечення фірми Heidenhain

## 2.6. Система ЧПК фірми Наас

Фірма Наас створила закриту систему числового програмного керування на базі G та M-кодів, що оптимізована спеціально для їхніх верстатів та забезпечує незалежність від сторонніх постачальників систем ЧПК.

Такий підхід створює умови високої надійності та довговічності системи керування при відносній простоті використання з 15-дюймовим кольоровим рідинно-кристалічним дисплеєм, який завдяки високій контрастності та високій інтенсивності дозволяє користувачам ефективно реалізовувати систему керування з трьома режимами та доступом до кожної функції керування.

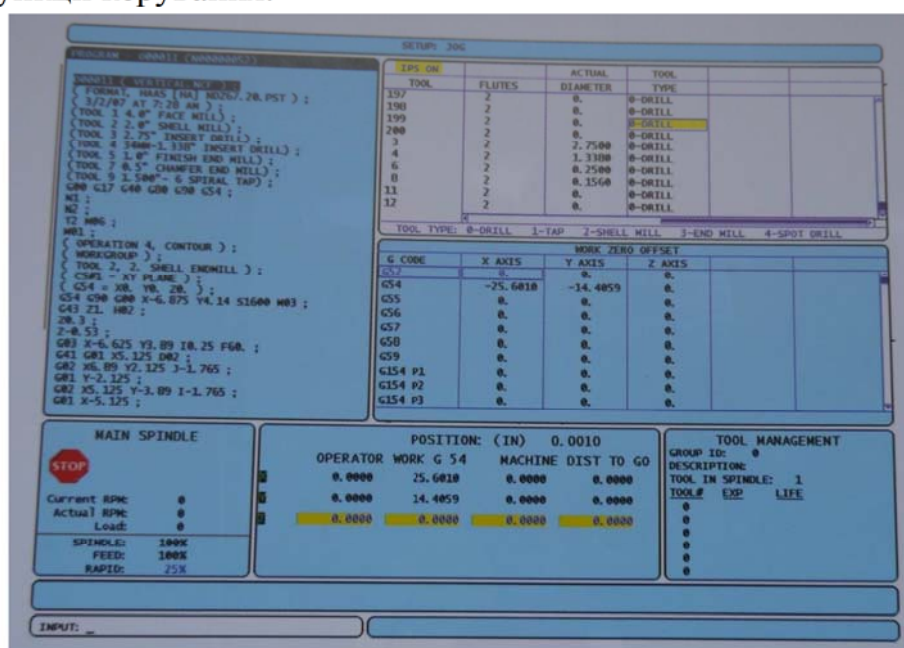


Рис. 2.6. Режим налагодження

В режимі налагоджування (Setup) користуються розташованим в лівому куті екрану вікном Active Program (Активна програма). В правому верхньому куті Tool Offsets (Зміщення інструменту), під яким розташовується Work Zero Offsets (Нульове зміщення деталі). Знизу зліва вікно - Spindle Information (Дані шпинделя), в якому відображається частота обертання шпинделя та кожні її зміни а також величина подачі. Рядом з даним вікном розташовано вікно Position (Положення), в якому відображається ім'я оператора, зміщення деталі, верстат та координати заданого переміщення. З правої сторони від цього вікна знаходиться інформація – «Керування інструментом».

Таке раціональне і компактне розміщення вікон та наступна активація необхідного у потрібний момент вікна дозволяє оператору переміщувати курсор у вікні за допомогою клавіш-стрілок на клавіатурі.

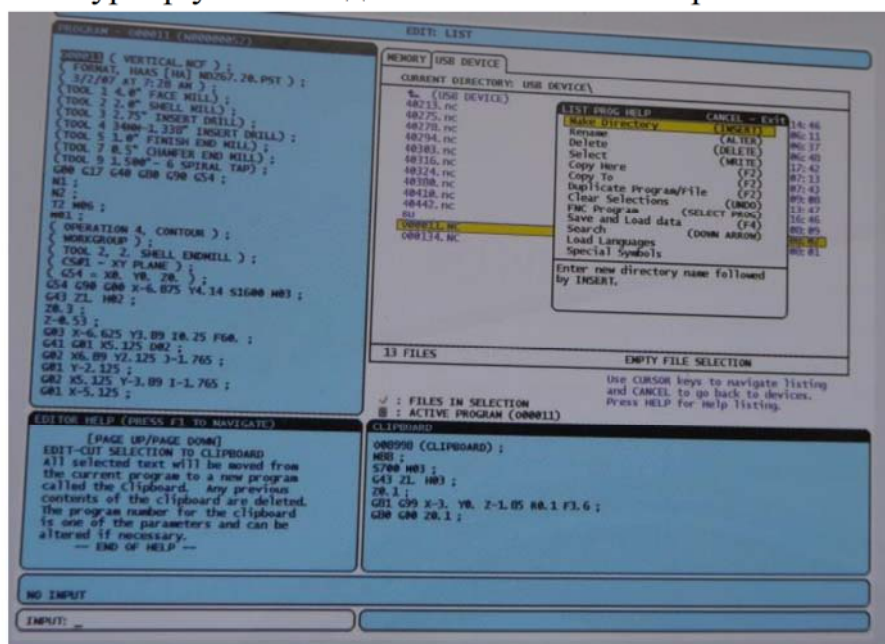


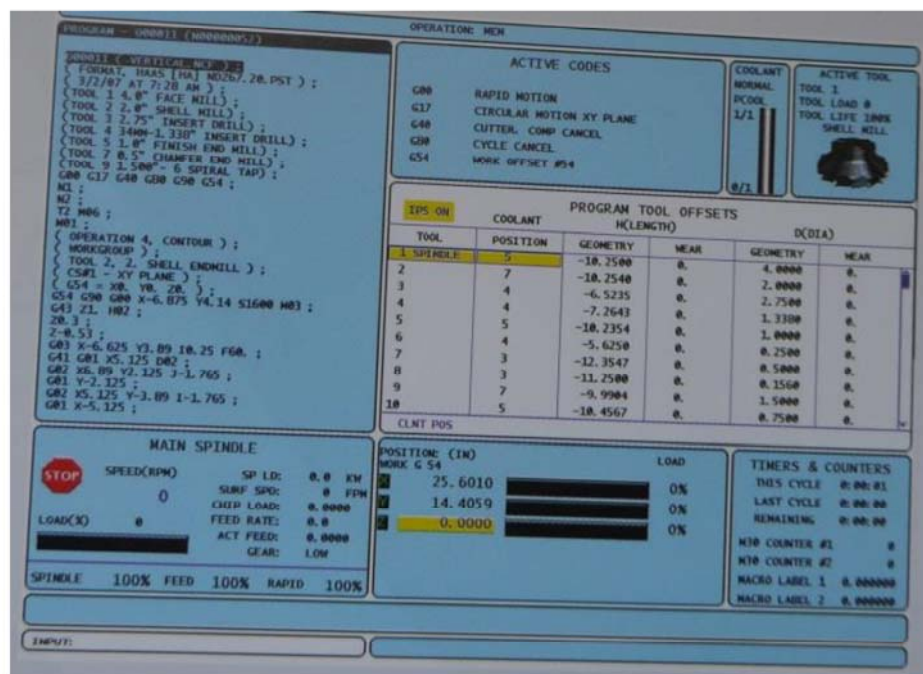
Рис. 2.7. Режим редагування

Завантаження програми виконують після установаження необхідних даних в режимі налагодження використовуючи попередньо режим Edit (Редагування), який викликається натисканням клавіші List Programs (Список програм). В режимі редагування оператор отримує доступ до програм, що знаходяться в кожному із підключених до верстата пристроїв. В цьому режимі використовується потужна функція – довідкове меню Quick-Key, яке має повний список і опис усіх функцій та переміщень у файлової системі. В довідковому меню оператор бачить назву команди та відповідну клавішу для кожної функції а також повний опис виконання кожної функції.

В режимі редагування використовуються також вікна Editor Help (Довідкова інформація для редактора) та Clipboard (Буфер обміну) і інші функції, що відносяться до редагування програми і спрощують роботу оператора.

В режимі Operation (Робота) представляється уся необхідна інформація для керування верстатом під час його роботи та швидкий доступ до неї. В верхньому лівому вікні екрану розташоване вікно Program Display (Дисплей програми), в якому відображається не тільки основна програма а також і підпрограми при їх наявності. Правіше розташовані «Активні G-коди» з текстовим супроводом та інформація про активний

інструмент, що включає і графічне зображення його типу. В центрі екрану знаходиться Offset Window (Вікно зміщень), в якому можна регулювати зміщення під час роботи верстата.



### 3. Підсистема приводів

Підсистема приводів включає різноманітні двигуни та гвинтові передачі. Вони виконують команди підсистеми керування для переміщення виконавчих органів верстата.

В конструкціях верстатів з ЧПК використовують **крокові двигуни та сервоприводи.**

Кроковий двигун – це електромеханічний пристрій, що перетворює електричний сигнал керування в дискретне механічне переміщення.

Розрізняють кілька основних видів крокових двигунів, що відрізняються конструктивним виконанням:

- крокові двигуни із змінним магнітним опором;
- крокові двигуни із постійним магнітним опором;
- гібридні двигуни.

#### 3.1. Кроковий двигун із змінним магнітним опором

Кроковий двигун із змінним магнітним опором (рис. 3.1) має кілька полюсів на статорі та ротор з магнітно-м'якого матеріалу (реактивний ротор).

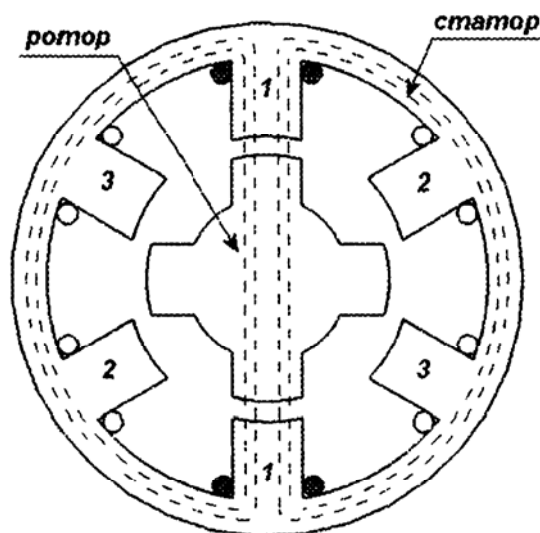


Рис. 3.1. Схема крокового двигуна зі змінним магнітним опором

В двигуні зі змінним магнітним опором (рис.3.1) статор має шість полюсів, а ротор – чотири виступи. В статорі є три незалежні обмотки, кожна з яких охоплює протилежні полюси.

При подачі електричного струму в одну із обмоток, ротор прагне зайняти положення, при якому магнітний потік повинен бути замкненим,

тобто виступи ротора будуть знаходитись навпроти тих полюсів статора, на обмотки яких подається струм. Якщо в цій обмотці струм вимкнути, а подати на наступну, то ротор знову повернеться, щоб замкнути магнітний потік своїми виступами. Для безперервного обертання ротора необхідно послідовно подавати струм на першу, другу та третю обмотки. При цьому крок обертання для даного двигуна буде  $30^\circ$ .

### 3.2. Кроковий двигун із постійним магнітним опором

Кроковий двигун з постійними магнітами (рис. 3.2) складається із статора з обмотками та ротора з постійними магнітами.

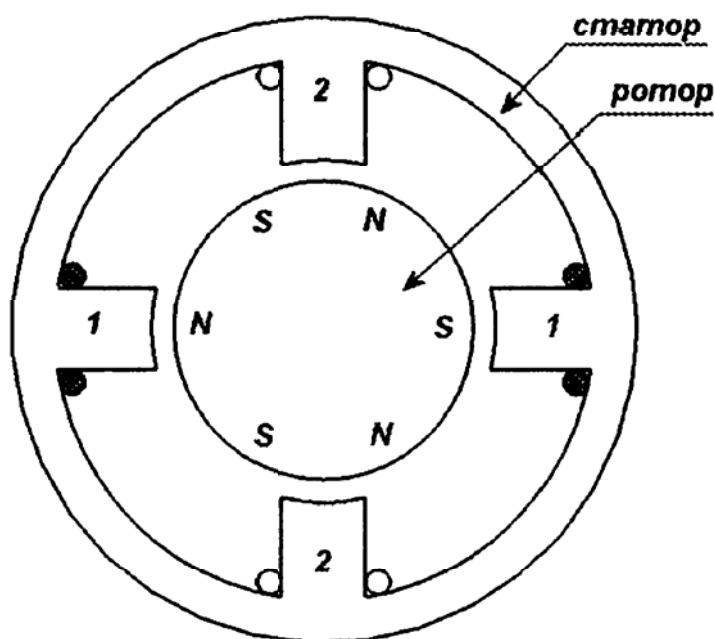


Рис. 3.2. Схема крокового двигуна із постійними магнітами

При подачі електричного струму на одну з обмоток, ротор займе положення, при якому різнойменні полюси статора та ротора будуть співпадати.

Для безперервного обертання ротора необхідно послідовно подавати електричний струм на першу та другу обмотки. При цьому крок обертання ротора складає  $30^\circ$ .

Більшість сучасних крокових двигунів є гібридними, тобто об'єднують переваги двигунів із змінним магнітним полем та двигунів з постійними магнітами, мають велике число полюсів статора та виступів ротора, що забезпечує малу дискретність обертання.

Коли підсистема керування посиляє кроковому двигуну електричний імпульс, то ротор повертається на кут, який залежить від конструкції двигуна (наприклад,  $0,72^\circ$ ). Якщо ходовий гвинт має крок 2мм, то один електричний імпульс заставить виконавчий орган верстата лінійно переміститись на величину:

$$\Delta = \frac{0,72^\circ}{360^\circ} \times 2 = 0,004 \text{ мм}$$

Ця величина називається розпізнаванням системи, або ціною імпульсу.

Неможливо перемістити виконавчий орган на величину, що менша ціни імпульсу.

Простота конструкції та легкість керування забезпечили високу популярність кроковим двигунам.

Основним недоліком цих двигунів є дискретність роботи, що може погіршити якість остаточної обробки та викликати ефект «ступінчастості» при обробці похилих та криволінійних поверхонь. Але вони можуть працювати без використання складного та дорогого зворотного зв'язку, що дозволяє створювати недорогі верстати.

### **3.3. Сервоприводи в сучасних високоточних верстатах з ЧПК**

Сучасні високоточні верстати з ЧПК використовують сервоприводи, що призначені для відпрацювання швидкості, моменту, позиції із заданою точністю та динамікою. До складу сервоприводу входить двигун, позиційний датчик та система керування, де присутнє трьохконтурне регулювання: позиція, швидкість, струм.

Сервопривод, отримуючи на вході величину параметра, що управляється, створює і підтримує цю величину на виході виконавчого органу, виходячи з показань датчика зворотного зв'язку.

Серводвигун – це двигун, що отримує команду від перетворювача частоти (конвертора). Сучасні серводвигуни – це компактні пристрої, що забезпечують оптимальний розгін та гальмування та створюють великі прискорення і крутний момент.

До переваг сервоприводів та серводвигунів можна віднести:

- плавні та точні переміщення при низьких швидкостях;
- висока потужність при мінімальних розмірах;

- безшумність роботи;
- надійність та безвідмовність, що дозволяє використовувати їх у відповідальних вузлах;
- підвищена швидкість переміщення елементів в порівнянні з іншими приводами;
- можливість вибору розпізнавання системи для виконання конкретної задачі.

До недоліків можна віднести більш високу складність системи забезпечення роботи системи ніж у крокових двигунах та високу вартість.

Якщо необхідно забезпечити плавний розгін або плавне гальмування, то з метою зменшення динамічних навантажень електродвигуна реалізують схеми керування на мікропроцесорах з використанням контролерів ЧПК. Сервоприводи на таких контролерах досягають потужності 15Квт та можуть розвивати крутний момент до 50Нм.

Серводвигуни обертального руху бувають синхронними з можливістю високоточного задавання швидкості обертання, кута повороту та прискорення і асинхронні, в яких швидкість дуже точно підтримується також і на низьких обертах.

### **3.4. Синхронні серводвигуни**

Синхронні серводвигуни – це трьохфазні двигуни із збудженням від магнітів та датчиком обертання ротора.

Основною перевагою синхронного серводвигуна є низький інерційний момент ротора відносно моменту обертання. Це дає можливість реалізації високої швидкодії, часу розгону до номінальної частоти обертання за мілісекунди та реверсування з повної швидкості в межах одного оберту ротора двигуна.

Головні переваги асинхронного серводвигуна від звичайного загальнопромислового – це мала вага, низький інерційний момент та високі максимальні швидкості, що дозволяє використовувати їх в високо динамічних системах.

Термін експлуатації асинхронного серводвигуна суттєво подовжує примусова вентиляція, яка дозволяє використовувати його при високих швидкостях.

Малу контурну похибку дозволяють отримати високі динамічні характеристики, що досягаються за рахунок зниження динамічного та



статичного розбалансування при використанні асинхронного сервоприводу в системі ЧПК.

Ціна також прийнятна, завдяки чому асинхронний привод є наймасовішим двигуном в промисловості.

Ринок пропонує велику кількість конструкцій серводвигунів для різних областей використання.

Так, фірмою NCT Kft розроблені та виготовляються синхронні (рис.3.3) та асинхронні (рис. 3.4) серводвигуни, що в першу чергу призначені для використання у верстатах з ЧПК. Завдяки спеціальному виконанню вони мають відмінні динамічні характеристики, які відповідають найвищим вимогам до сучасних верстатів і не вимагають обслуговування протягом довготривалого періоду експлуатації.

Синхронні серводвигуни серії А – це трьохфазні двигуни із збудженням від постійних магнітів з восьмиполусною обмоткою статора, з'єднаною за схемою «зірка» та восьмиполусним ротором. Двигун також оснащується інкрементальним або абсолютним датчиком положення ротора відносно статора. Виконання повністю закрите з герметично вбудованими підшипниками вала та електричними роз'ємами. Відведення тепла виконується за рахунок звичайної конвенції тому немає необхідності в примусовій вентиляції. Збудження обмоток статора а також магнітна індукція в повітряному зазорі є практично синусоїдальними величинами, що дозволяє дуже точно регулювати величину крутного моменту незалежно від положення ротора і таким чином забезпечувати надзвичайно точне регулювання швидкості. Номінальна потужність таких двигунів складає 310...5230Вт в залежності від виконання а частота обертання 2000об/хв або 3000об/хв.



Рис. 3.3 . Синхронні серводвигуни фірми NCT Kft

Синхронні серводвигуни серії Аі ідентичні за конструкцією двигунам серії А але для постійних магнітів ротора використовують рідкоземельний метал (неодимій або самарій-кобальт), що дозволило підвищити потужність та частоту обертання.

### **3.5. Асинхронні серводвигуни**

Асинхронні двигуни NCT були спеціально розроблені для верстатів з ЧПК і за своїми характеристиками відповідають високим динамічним навантаженням в сучасних верстатах. Це забезпечує довготривалу експлуатацію без додаткового технічного обслуговування. Ці серводвигуни мають наскрізний отвір у валу ротора, що дозволяє подачу змащувально-охолоджувальної рідини через інструмент при прямому підведенні.

Асинхронні серводвигуни серії АіS мають фланцеве кріплення та забезпечують частоту обертання від 1500об/хв до 15000об/хв при потужності від 10,5Квт до 22Квт.

Асинхронні серводвигуни серії DA можуть мати фланцеве кріплення або встановлюватись на лапах. Забезпечують частоту обертання від 1000об/хв до 5000об/хв при потужності від 11Квт до 29Квт.

Асинхронні серводвигуни серії AMS – це компактні мотор-шпинделі, оскільки вони інтегровані в шпинделі токарних верстатів. Завдяки високій динамічній жорсткості та низькому рівні вібрацій вони забезпечують високу точність оброблювання.

Переваги мотор-шпинделя перед традиційною пасовою передачею наступні:

- високі динамічні характеристики завдяки малій інерційній масі а також менші втрати енергії та менші витрати електричного струму;
- менші похибки форми обробленої деталі завдяки відсутності деформацій, що виникають при натягуванні пасової передачі;
- завдяки низькому рівню вібрацій при оброблюванні досягається висока якість обробленої поверхні та збільшується стійкість різального інструменту.
- підвищується точність геометричних параметрів обробленої деталі та повторюваність розмірів завдяки інтенсивному рідинному охолодженню, що також збільшує і термін служби підшипників.



Рис. 3.4 . Асинхронні серводвигуни та мотор-шпинделі фірми NCT Kft

### 3.6. Ходовий гвинт

Другою складовою підсистеми є ходовий гвинт.

Виконавчий орган при зрушенні з місця починає рух не одночасно з дією керуючого сигналу, а після того, як будуть вибрані зазори в передачах, виникне пружна деформація елементів і зусилля, що діє на виконавчий орган, перевищить опір сил тертя та сил різання. При конструюванні ходових гвинтів, що є найважливішими ланками при передачі команди на переміщення виконавчим органам, дуже важливо враховувати дію цих факторів. Тому у верстатах з ЧПК використовують передачі «гвинт-гайка кочення», що характеризуються високою точністю, зносостійкістю та жорсткістю і збільшеним діаметром ходового гвинта.

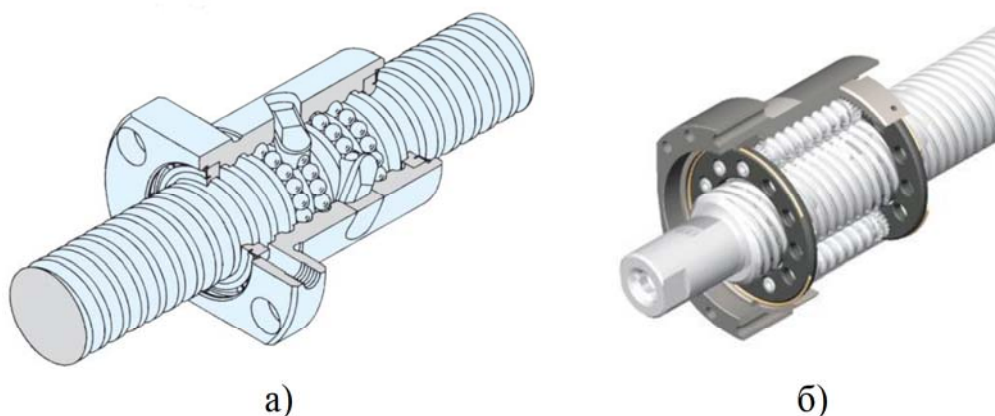


Рис. 3.5. Передача гвинт – гайка кочення ( а-кулькова; б – роликова)

Ходовий гвинт жорстко кріпиться в осьовому напрямку з використанням упорних підшипників з попереднім натягом.

Удосконалений ходовий гвинт верстата з числовим програмним керуванням дозволяє виконувати переміщення виконавчого органу з мінімальним тертям і практично без зазорів. Беззазорні гвинтові передачі необхідні для високоточного позиціонування. Попутне фрезерування також неможливе при наявності в приводі подач зазорів.

Для підвищення коефіцієнту корисної дії передачі гвинт-гайка застосовують кулькові передачі з циркулюючими кульками (рис. 3.6).

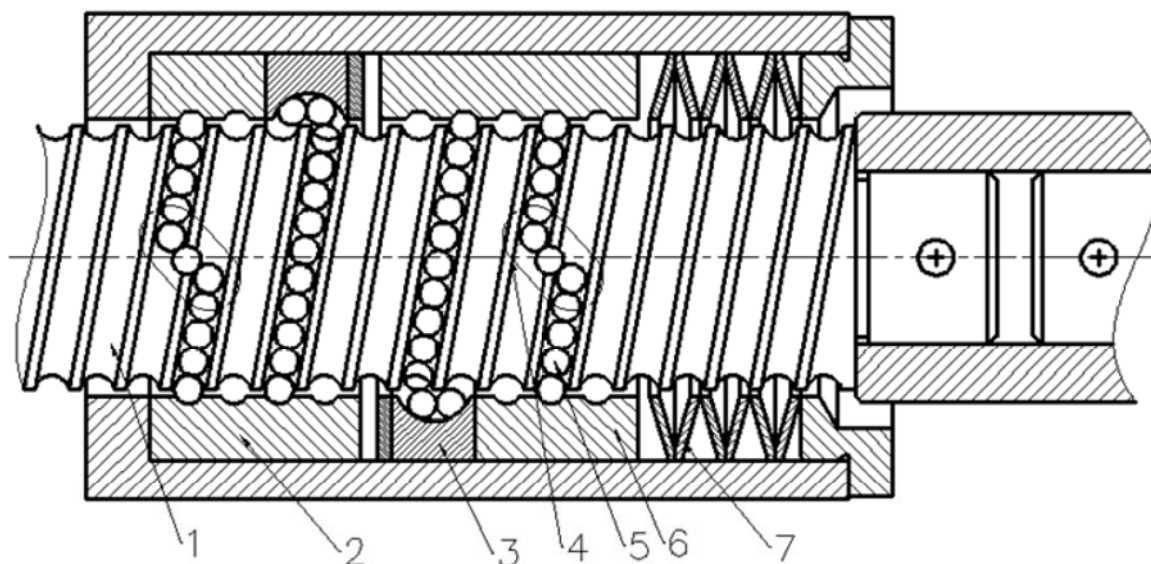


Рис.3.6. Кульково-гвинтова передача

Гвинт 1 виготовлений з канавкою дугової форми. Аналогічну форму канавки мають обидві частини 2 та 6 гайки. В простір, створений витками гвинта та гайки, закладені кульки 5. Кульки розташовані на довжині одного витка, кінці якого з'єднуються вкладишами 3 та 4, що врізані в тіло гайки. При обертанні гвинта кульки рухаються по канавці гвинта, входять в канавку вкладиша і по ній повертаються знову в канавку гвинта. Таким чином реалізується безперервна циркуляція кульок. Кожна з частин гайки має по два замкнутих витки. На кресленні два вкладиші показані в перерізі, а два – умовно в плані. Обидві частини гайки направляються шпонкою, а для усунення зазорів притискаються тарільчастими пружинами 7.

Така схема жорсткого вибирання зазорів потребує високої точності виготовлення профілю та кроку гвинтової канавки.

Тому кульково-гвинтові пари трудомісткі у виготовленні, але широко використовуються у приводах подач верстатів з ЧПК завдяки низькому коефіцієнту тертя та відсутності зазорів. Найчастіше при їх виготовленні використовують для гвинта – сталь ХВГ (HRCe 60), а гайки – сталь 9ХС.

Крім жорсткої схеми широко застосовують також пружинне або силоне вибирання зазору, що охоплює як кульковий так і весь кінематичний ланцюг від двигуна до гвинта.

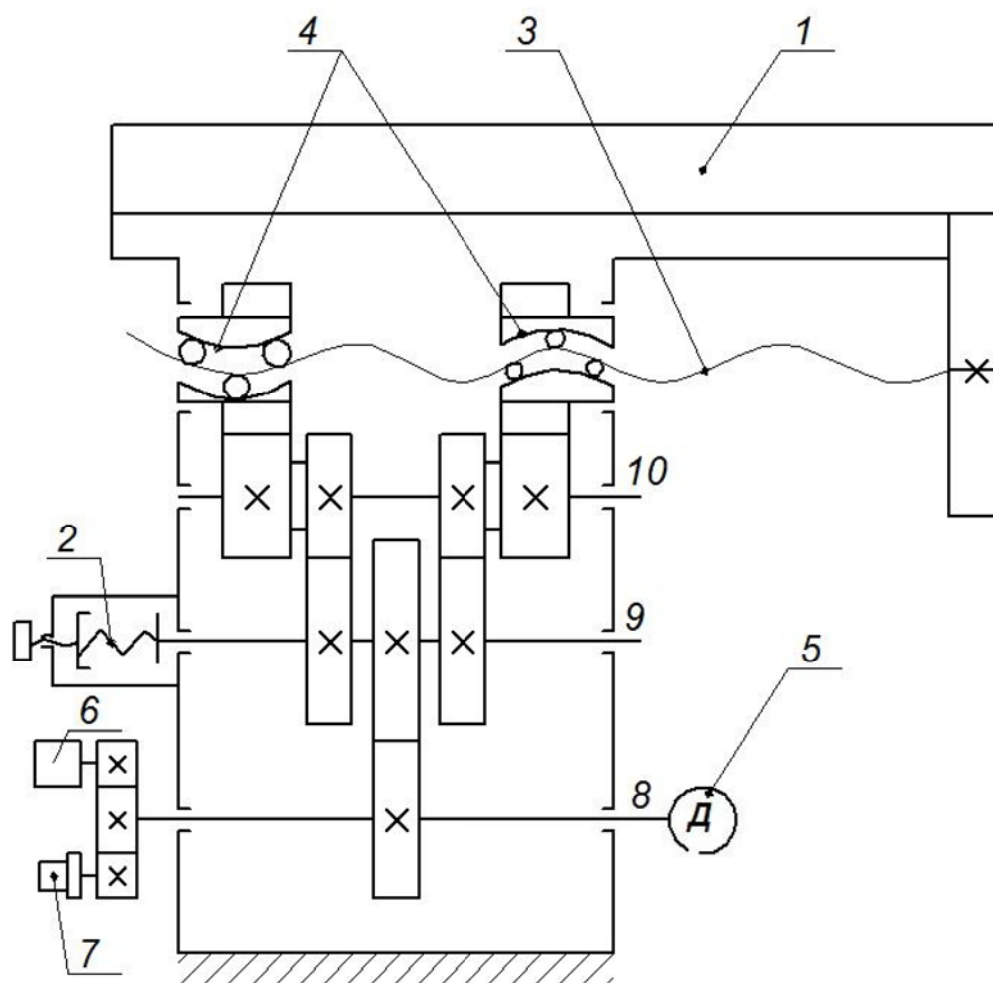


Рис.3.7. Силоний кінематичний ланцюг

Силоний кінематичний ланцюг (рис. 3.7) слугує для переміщення робочого органу-столу 1 за допомогою ходового гвинта 3. Після другого проміжного валу 9 кінематичний ланцюг розгалужується. Два зубчасті блоки на валу 10 одночасно обертають дві половини кулькової гайки 4. За допомогою косозубих коліс на валу 9 і можливості їх зміщення під дією пружини 2, що створює постійне зусилля, обидві половини кулькової гайки можуть не тільки одночасно обертатися, але і повертатись одна щодо іншої на невеликий кут. Цей кут достатній для вибирання зазору і забезпечення деякого натягу, що задається зусиллям пружини 2. Від першого робочого валу 8, що обертається двигуном 5 через зубчасту передачу, обертається тахогенератор 6 та датчик зворотного зв'язку 7.

### **Запитання до розділу 3**

1. Які складові елементи підсистеми приводів верстатів з ЧПК?
2. Які особливості роботи крокового двигуна і два основні варіанти його виконання?
3. Які особливості крокових двигунів із змінним магнітним полем?
4. Які особливості крокових двигунів із постійними магнітами?
5. Що таке ціна імпульсу і як вона визначається для верстатів з ЧПК?
6. Які переваги серводвигунів в порівнянні з кроковими?
7. Охарактеризувати конструктивні особливості синхронного серводвигуна фірми NCT Kft та його переваги.
8. Охарактеризувати конструктивні особливості асинхронного серводвигуна фірми NCT Kft та його переваги.
9. Які конструктивні особливості кулькової гвинтової передачі?
10. Які схеми вибирання зазорів використовують в кулькових гвинтових передачах?

#### 4. Підсистема зворотного зв'язку

Підсистема зворотного зв'язку головним чином повинна забезпечувати підсистему керування інформацією про реальне положення виконавчого органу та швидкість двигунів.

Підсистема зворотного зв'язку може бути відкритою або замкнутого типу.

Системи відкритого типу реєструють лише наявність або відсутність сигналу із підсистеми керування. Вони не можуть дати інформацію про реальне положення виконавчого органу та швидкості двигунів, тому в сучасних верстатах з ЧПК практично не використовуються.

Системи замкнутого типу використовують зовнішні датчики для перевірки необхідних параметрів.

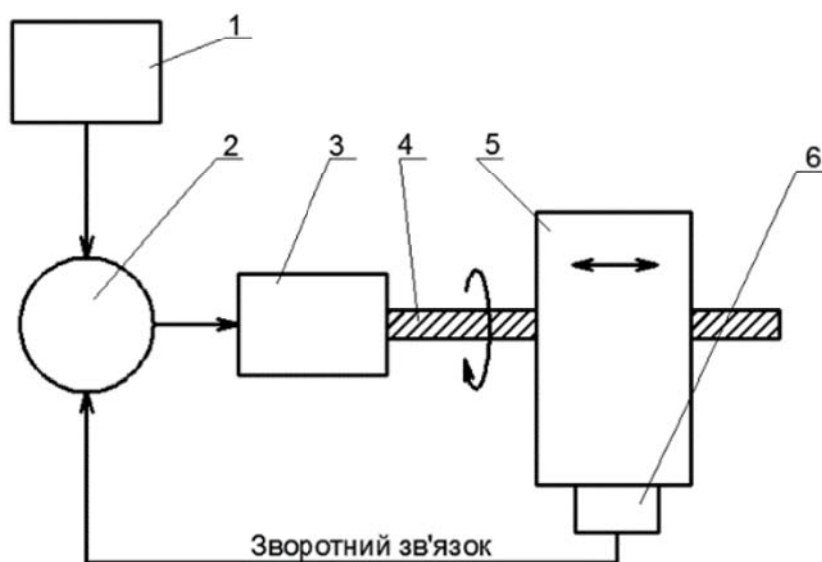


Рис.4.1. Схема зворотного зв'язку на верстаті з ЧПК: 1 – система числового програмного керування; 2 – контролер; 3 – двигун; 4 – ходовий гвинт; 5 – виконавчий орган; 6 – датчик.

У верстатах з ЧПК використовують два типи датчиків положення: лінійні датчики та датчики обертання.

Лінійні датчики положення (рис.4.2) використовуються практично в усіх сучасних верстатах з ЧПК для точного визначення абсолютного або відносного положення виконавчих органів. Датчики об'єднують два взаємозв'язаних елементи: растрову шкалу та головку зчитування.

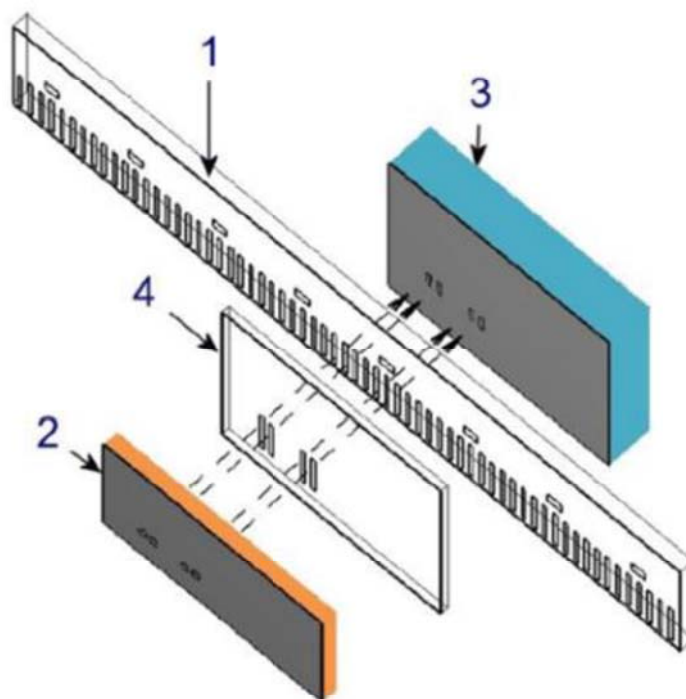


Рис.4.2. Схема лінійного датчика положення

Растрова шкала 1 (рис. 4.2), що розташовується вздовж напрямних верстата, виготовляється у вигляді лінійки з вузькими прорізами (растрами). Головка зчитування переміщується разом з виконавчим органом і складається з освітлювача 2, фотоприймачів 3 та індикаторної пластини 4.

Освітлювачі з індикаторною пластинною та фотоприймач знаходяться по різні сторони від лінійки.

На індикаторній пластині також мають місце дві растрові ділянки із зміщеним кроком для формування двох сигналів.

Коли зчитувальна головка переміщується вздовж растрової шкали, то світлові сигнали проходять через індикаторну пластину, шкалу і реєструються фотоприймачем. Отримані сигнали дозволяють визначити напрямок та величину переміщення.

На растровій шкалі може знаходитись додаткова доріжка референтних міток для задавання власного початку відліку.

Для гарантування плавного і точного керування переміщеннями виконавчих органів датчики положення повинні мати високу розпізнавальну здатністю по всіх координатних осях.



#### 4.1. Датчики лінійних переміщень

На сьогоднішній день однією з передових фірм на світовому ринку по розробці і виготовленню датчиків лінійних та кутових переміщень, датчиків обертання і пристроїв цифрової індикації та систем числового програмного керування є фірма HEIDENHAIN.

Висока якість продукції забезпечується використанням спеціального виробничого обладнання та засобів вимірювання для виготовлення і вимірювання лінійних та кутових шкал а також копіювальні пристрої, що розроблюються і виготовляються на власному виробництві.

Основною складовою вимірювальних приладів фірми HEIDENHAIN є прецизійні шкали з поділками у вигляді штрихової сітки з періодом від 0,25 до 10мкм.

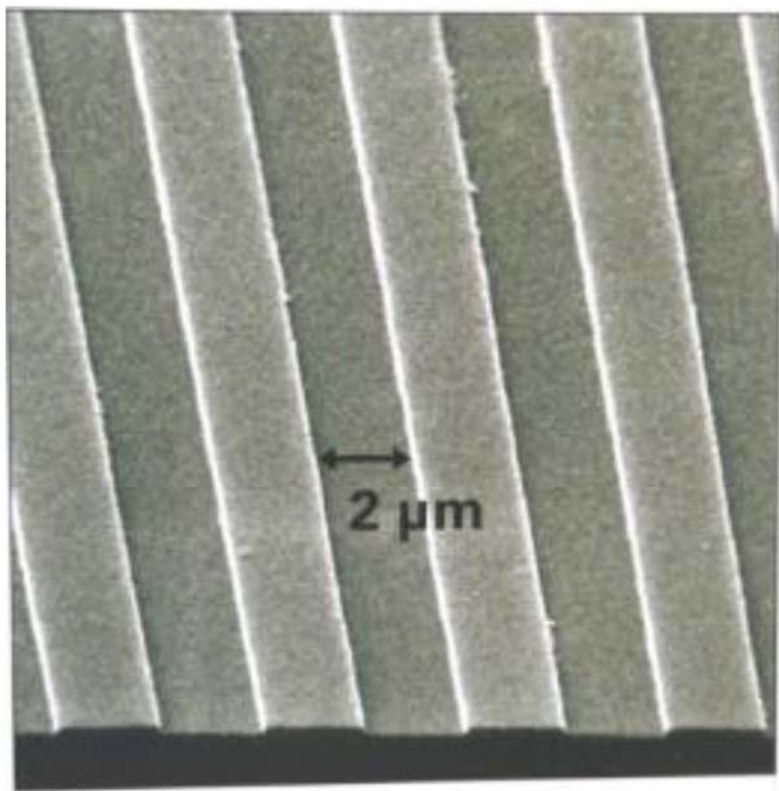


Рис.4.3. Прецизійна шкала з поділками у вигляді штрихової сітки

Для нанесення поділок використовують спеціальні методи.

Прецизійні шкали, що виконані методом DADUR, виготовляються шляхом нанесення тонкого шару хрому на носій із скла або склокераміки. Точність поділок знаходиться в межах мікрометрів та вище.

Шкала, що виготовляється методом AURODUR, складається із рефлектуючих золотих штрихів та витравлених матових зазорів на носіях із спеціальної сталі.

Шкала, що виготовляється методом METALLUR, має практично гладку поверхню завдяки особливій оптичній структурі із рефлектуючих золотих шарів, що забезпечує її нечутливість до забруднення.

В трьохмірних решітках, що мають особливі властивості і виготовлені за спеціальними технологіями, ширина поділок знаходиться в межах від кількох мікрметрів до четверті мікрметра.

Метод SUPRADUR дозволяє виготовляти поділки, що виглядають як і трьохмірні фазові решітки, але мають гладку поверхню, завдяки чому вони не чутливі до забруднень.

Перераховані методи створення вимірювальних шкал дозволяють забезпечувати високу точність вимірювання при їх використанні у відповідних пристроях.

Так, шкала типу DADUR використовується в закритих та відкритих датчиках лінійного переміщення, розроблених для металорізальних верстатів, які визначають положення лінійної вісі без додаткових механічних передавальних елементів. Датчиком визначається похибка переміщення виконавчого органу по лінійній вісі і коректується системою керування верстатом.

Такий спосіб дозволяє уникнути впливу різних джерел на похибки:

- похибку позиціонування, що викликана нагріванням кульково-гвинтової пари (КГП);
- похибку внаслідок наявності зазорів у кульково-гвинтовій парі;
- кінематичну похибку внаслідок похибки монтажу кульково-гвинтової пари.

Сучасні верстати працюють з високими швидкостями і при високих режимах різання, що викликає підвищення температури в закритому робочому просторі верстата та великі динамічні навантаження. При розробці та використанні лінійних датчиків ці особливості експлуатації повинні враховуватися.

В ідеальному випадку термічні властивості датчика повинні відповідати термічним властивостям вимірюваного об'єкту. При змінах температури датчик повинен відповідним чином розтягуватись або стискатись і ці зміни повинні бути повторюваними. В лінійних датчиках фірми HEIDENHAIN носії шкали мають визначені коефіцієнти теплового розширення, що дозволяє підібрати датчик для кожного конкретного випадку використання у верстаті.

Для забезпечення ефективної роботи в умовах динамічних навантажень виконавчих органів верстата лінійні датчики виконані з

високою жорсткістю в напрямку вимірювань, а невелика маса рухомих частин датчика гарантує його високі динамічні властивості.

Використання високоякісних матеріалів для виготовлення окремих елементів датчиків та конструктивні особливості дозволяють забезпечувати їх довговічність та точність на протязі усього терміну експлуатації верстатів, де лінійні переміщення по осях досягають значних величин. Так, при експлуатації металообробних верстатів з ЧПК величини цих переміщень можуть складати більше 3000км за рік.

В закритих лінійних датчиках відсутній безпосередній контакт між шкалою та елементом зчитування, а захист сигналу від впливу зовнішніх джерел досягається екрануванням. При необхідності, в захисний кожух може подаватися стиснене повітря для надійного захисту від забруднення.

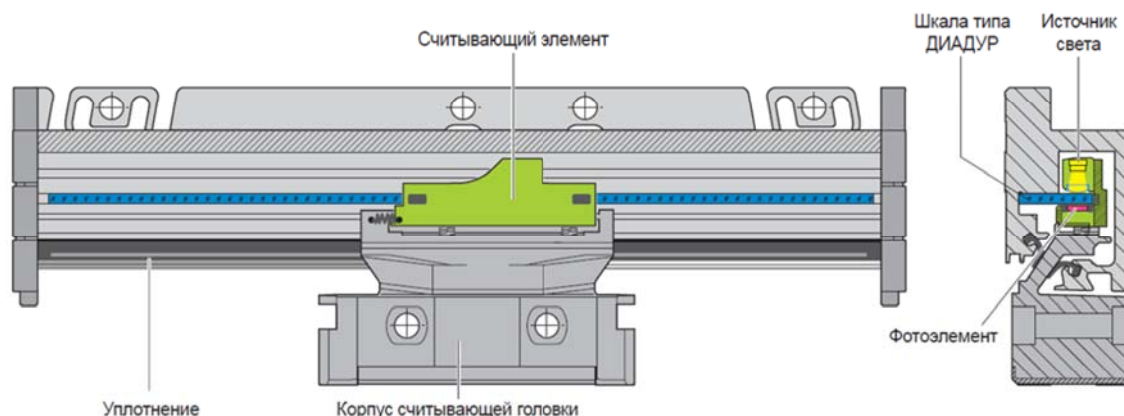


Рис.4.4. Принципова схема закритого лінійного датчика LS 183

Елемент зчитування переміщується вздовж шкали без контакту з нею завдяки використанню фотоелектричного способу зчитування, що дозволяє розпізнавати штрихи шириною в декілька мікрметрів і генерувати вихідний сигнал з дуже малим періодом.



Рис.4.5. Датчики лінійних переміщень закритого типу фірми HEIDENHAIN

При експлуатації лінійних датчиків фірми HEIDENHAIN використовують абсолютний або інкрементальний метод вимірювання.

При абсолютному методі вимірювання після включення апаратури зразу ж стає доступним абсолютне значення положення вісі в даний момент, яке може бути зчитано відповідною системою зчитування. Інформація про фактичне положення зчитується із закодованої спеціальним чином шкали.

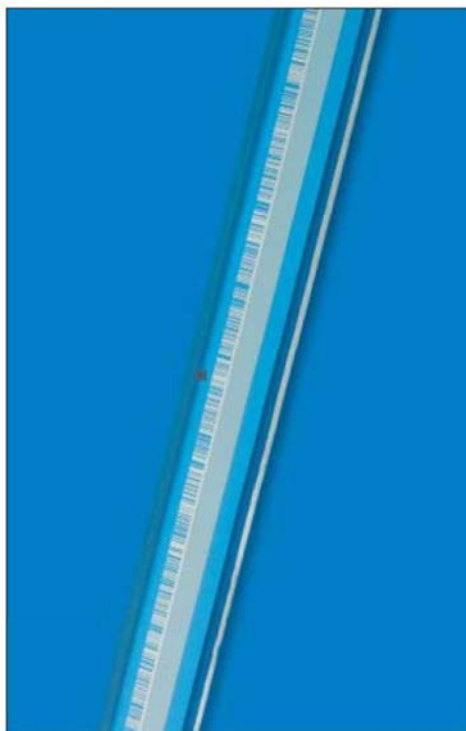


Рис. 4.6. Шкала абсолютного лінійного датчика

В інкрементальному методі вимірювання шкала складається тільки з одного ряду рівномірних штрихів. Дані про положення отримуються шляхом розрахунку окремих інкрементів (кроків вимірювання) відносно вибраної нульової точки, в якості якої на шкалі використовується окремий ряд штрихів, що несуть референтну мітку. Референтна мітка має такий же період сигналу як і інкрементальний сигнал. Щоб встановити заново або відновити нульову точку необхідно проїхати референтну точку.

В самому невигідному варіанті для перетину референтної мітки необхідно проїхати більшу частину вимірюваної відстані. Щоб зменшити цю ділянку більшість датчиків мають кодовані референтні мітки – це додатковий ряд штрихів, що має багато референтних міток на різній відстані одна від іншої. Система керування визначає положення вже після проходження двох сусідніх міток, що складає всього декілька міліметрів.

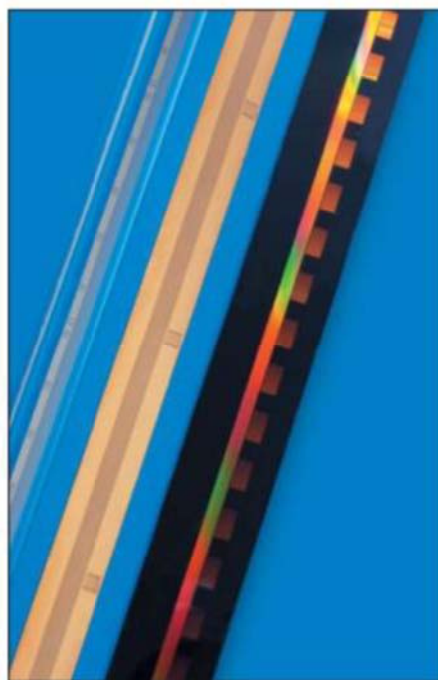


Рис. 4.7. Шкали інтерферентних датчиків

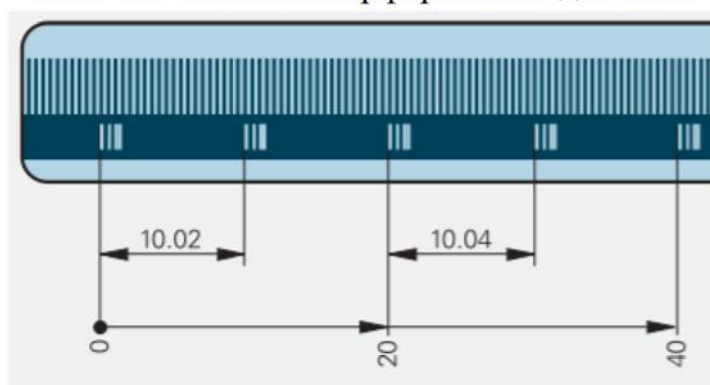


Рис. 4.8. Схематичне представлення інкрементальної шкали з кодованими референтними мітками

Нульова точка в кодованих датчиках також визначається розрахунком інкрементів між двома референтними мітками за розробленою окремою методикою.

Точність вимірювання довжини визначають наступні параметри:

- точність штрихів шкали;
- якість зчитування сигналу;
- якість обробки сигналу;
- похибки взаємного розташування зчитувальної головки та шкали.

Точність штрихів шкали визначається методами їх виготовлення. Датчики фірми HEIDENHAIN використовують прецизійні шкали з поділками у вигляді штрихової сітки з періодом від 0,25мкм до 10мкм.

Поділки наносяться спеціальним методом, що забезпечує кінцеву точність вимірювальних приладів.

Більшість приладів фірми HEIDENHAIN використовують фотоелектричний спосіб зчитування, який реалізується без контакту і виключає зношування робочих елементів. Цей спосіб дозволяє розпізнавати штрихи шириною в декілька мікрометрів та генерувати сигнали з дуже малим періодом.

**Метод відображення** в спрощеному вигляді – це генерація сигналу на основі значень світло-тінь (рис. 4.9).

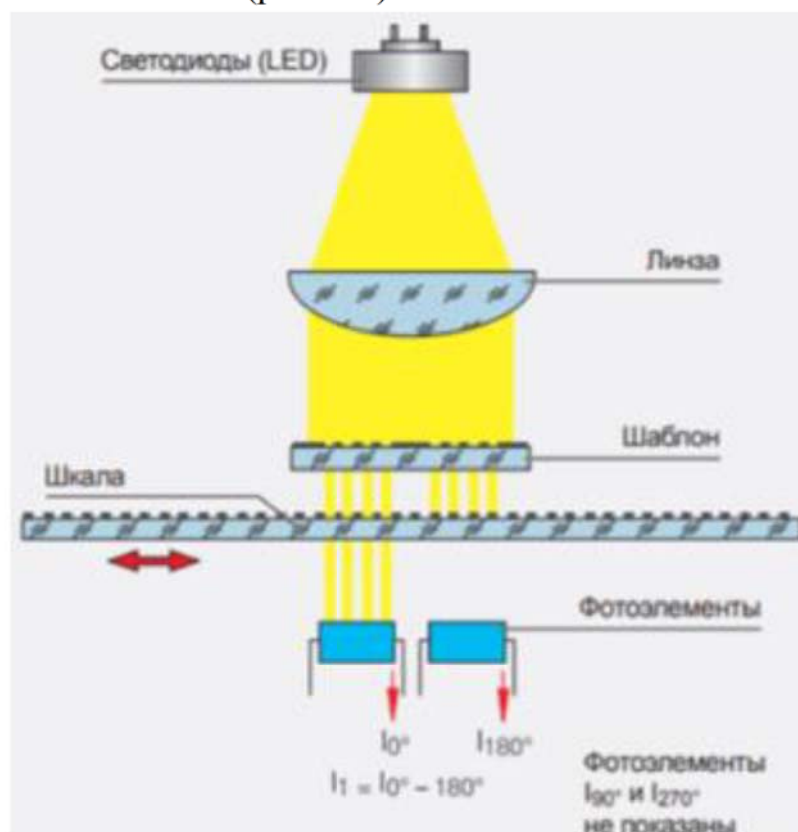


Рис.4.9. Схематичне зображення зчитування методом відображення

Дві шкали із штрихами одного або схожого періоду (шкала і шаблон) рухаються одна відносно іншої. Носій штрихів шаблону виготовляється із прозорого матеріалу, а носій самої шкали також може бути прозорим або мати металічну рефлектувальну поверхню.

Після проходження паралельних променів світла від світлодіодів через фокусуючу лінзу і шаблон створюється відповідна світло-тіньова послідовність. Потім промені, що пройшли через шаблон, попадають на шкалу. При переміщенні шаблону вздовж шкали штрихи на шаблоні можуть співпадати зі штрихами на шкалі, створюючи на виході в місцях просвітів «світло». Якщо ж штрихи накладаються на просвіти, то на виході

створюється тінь. Ряд фотоелементів перетворює цей світловий сигнал в електричний. Штрихи на шаблоні структуровані спеціальним методом завдяки чому вони фільтрують світловий потік таким чином, щоб він наближався до синусоїдальної форми.

Чим менша відстань між штрихами тим меншою і точнішою повинна бути відстань між шкалою та зчитувальним елементом.

На методі відображення працюють датчики лінійних переміщень LC, LS та LB.

**Інтерферентний метод** зчитування заснований на дифракції та інтерференції світла при проходженні його крізь прецизійні штрихи. Із отриманого сигналу розраховують величину переміщення.

Шкалою слугує дифракційна ступінчаста решітка, яка виготовляється нанесенням рефлектуючих штрихів висотою 0,2мкм на рефлектуючу поверхню (рис. 4.10).

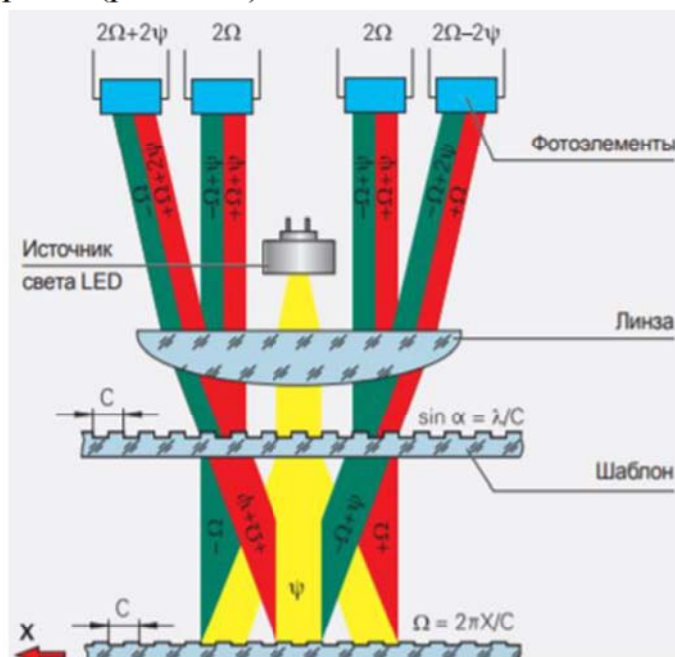


Рис. 4.10. Оптична схема інтерферентного методу зчитування:  $c$  – період шкали;  $\psi$  - зміщення фази світлової хвилі при проходженні через шаблон;  $\Omega$  – зміщення фази світлової хвилі при переміщенні шкали вздовж вісі

При попаданні світлової хвилі на шаблон вона розділяється на три хвилі: 1; 0 та -1 порядків з відносно однаковою інтенсивністю. Від шкали з фазовою решіткою вони відбиваються таким чином, що найбільша інтенсивність у хвиль 1 та -1 порядків. Ці хвилі знову зустрічаються на шаблоні та, накладаючись одна на іншу, огинають його штрихи. При

цьому створюються дві групи хвиль, які покидають шаблон під різними кутами. Фотоелементи перетворюють інтенсивність хвиль в електричний сигнал.

При переміщенні шаблону відносно шкали на один період фронт хвилі першого порядку зміщується на одну довжину хвилі в плюс, а фронт хвилі -1 порядку на одну довжину хвилі в мінус. Оскільки ці дві хвилі інтерферують після шаблону, то їх зміщення досягає двох довжин хвиль. Таким чином отримують два періоди сигналу при одному відносному зміщенні на один період.

Прилади з інтерферентним методом зчитування працюють найчастіше зі шкалами, період яких складає 8мкм та 4мкм.

Датчики лінійного переміщення, що використовують інтерферентний метод, позначаються LF.

При визначенні точності вимірювання розрізняють похибку шкали, що віднесена до всієї довжини шкали та похибку, що віднесена до одного періоду сигналу.

Похибка вимірювання (рис. 4.11), що віднесена до всієї довжини шкали використовується при визначенні точності лінійних датчиків закритого типу, яка задається в класах. Вони мають наступне визначення: Граничні значення  $\pm F$  кривої похибки вимірювань для будь-якого шляху вимірювання (максимум  $l_m$ ) знаходиться в межах класу точності  $\pm a$ .



Рис. 4.11. Похибка вимірювання  $a$ , що віднесена до всієї довжини шкали

В закритих лінійних датчиках ці величини задаються для всієї вимірювальної системи, що включає і зчитувальну головку, Вони



називаються точністю системи і визначаються при заключному контролі датчика та заносяться в протокол вимірювань.

Похибка вимірювання, що віднесена до одного періоду сигналу (рис. 4.12), визначається як величиною періоду шкали так і якістю штрихів та засобів зчитування. В будь-якій точці вимірювання вона не перевищує  $\pm 2\%$ , а для датчиків лінійних переміщень LC та LS не перевищує  $\pm 1\%$ .

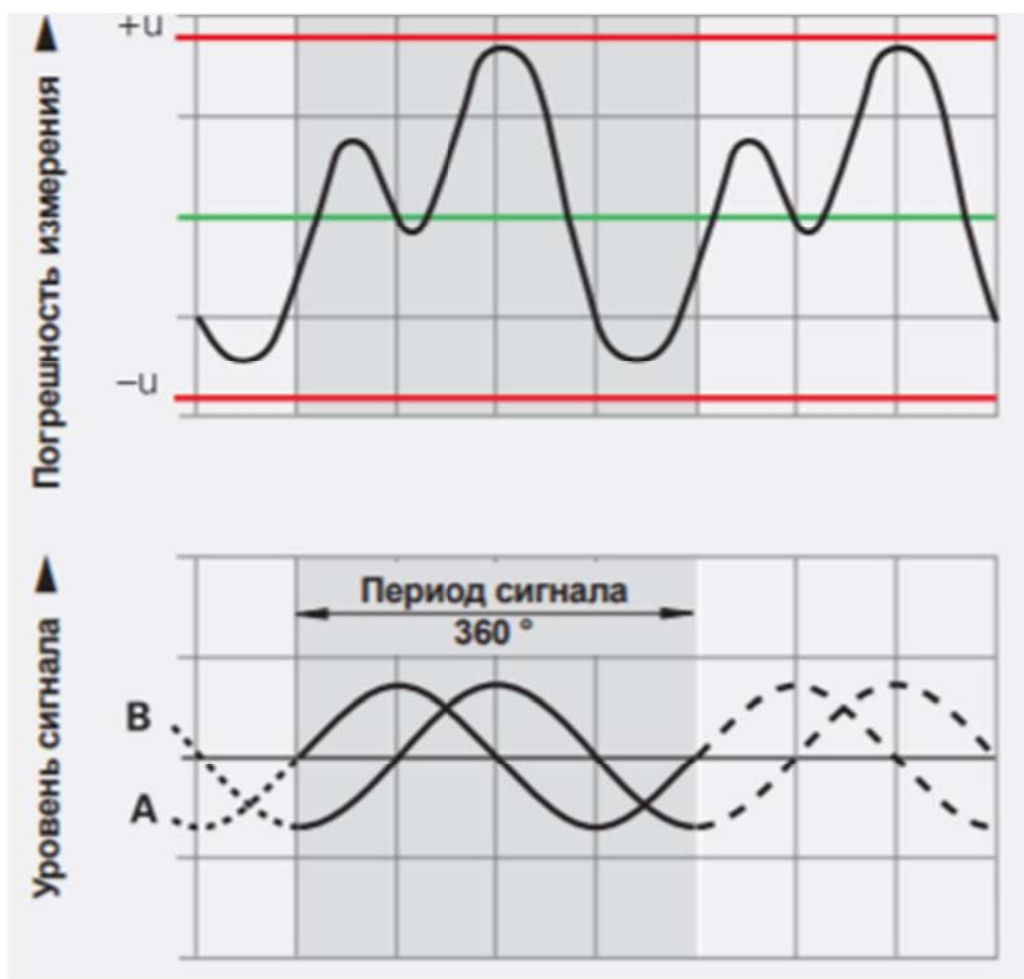


Рис. 4.12. Похибка вимірювання  $U$ , що віднесена до одного періоду сигналу

Похибка вимірювання зменшується зі зменшенням періоду сигналу шкали. Так для датчиків LF з періодом сигналу 4мкм вона складає  $\pm 0,08$ мкм, для датчиків LC та LS з періодом сигналу 20мкм –  $\pm 0,2$ мкм, а для датчиків LB з періодом сигналу 40мкм –  $\pm 0,8$ мкм.

Перевірка лінійних датчиків переміщення виконується при нормальній температурі  $\pm 20^\circ$ . Обов'язково вказується діапазон температур, при яких датчик зберігає точність вимірювання.

## 4.2. Датчики кутових переміщень

Датчики обертання (рис. 4.13) закріплюються на валу двигуна і дозволяють визначити його кутове положення.

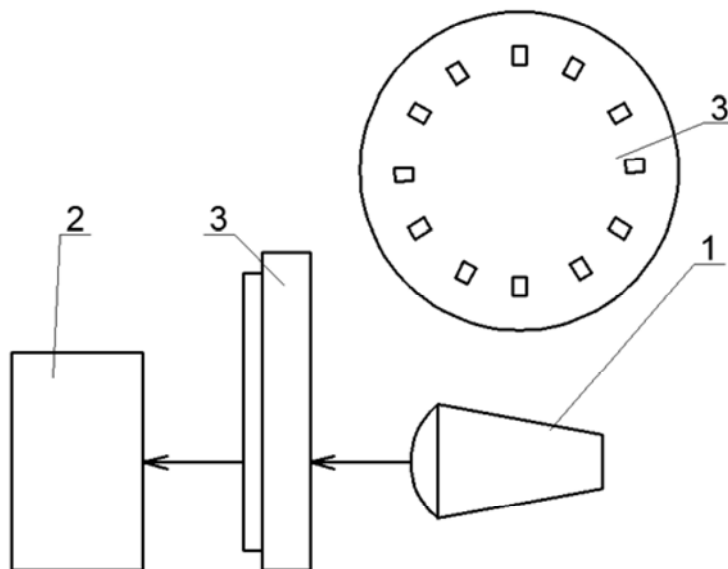


Рис.4.13. Датчик обертання

Цей датчик складається з джерела світла -1, оптичного датчика (приймача) - 2 та диску з радіальними прорізами (растрами) - 3.

Растровий диск закріплюється на валу. Джерело світла та приймач знаходяться по різні сторони диска.

При обертанні диска промені крізь прорізи попадають на оптичний датчик, що працює як перемикач і вмикається та вимикається при попаданні на нього світла.

Це дає можливість визначити відносне або абсолютне положення та напрямок обертання вала. Отримана інформація відправляється в підсистему керування.

Суттєвим недоліком цих датчиків є неможливість безпосереднього вимірювання лінійного положення виконавчого органу верстата. Вони дають тільки розрахункове положення, що базується на даних про крок ходового гвинта та кутове положення. Тому в високоточних верстатах не використовуються. Їх можна використовувати в шпindelних вузлах для визначення частоти обертання та кутового положення шпинделя.

Фірма HEIDENHAIN розробила і використовує датчики обертання (рис. 4.14), що використовуються для вимірювання швидкості обертання, а

при монтажі на ходовому гвинті або кульково-гвинтовій парі – для визначення лінійних переміщень.



Рис. 4.14. Датчики обертання з муфтою статора фірми HEIDENHAIN

Датчики обертання своїм пустотілим валом встановлюються на вал приводу та закріплюються з боку ротора двома гвинтами або трьома ексцентриковими затискачами. Датчики із наскрізним пустотілим валом можна також закріплювати і з боку кришки.

Точність датчиків обертання залежить від наступних параметрів:

- радіального зміщення шкали;
- зміщення центру шкали відносно вісі підшипників;
- радіального биття підшипників;
- похибки, що викликана використанням з'єднувальної муфти;
- інтерполяційних відхилень при обробці сигналів.

У вимірювальних датчиках використовують оптичний метод зчитування.

Як і в лінійних датчиках для вимірювання можуть використовуватися абсолютний або інкрементальний методи вимірювання.

При абсолютному методі вимірювання інформація про абсолютну координату зчитується по поділках шкали, що виконана у вигляді послідовно кодованої структури або у вигляді кількох паралельних доріжок.

Сигнал з окремої інкрементальної доріжки інтерполюється для визначення поточного значення координати і одночасно використовується для генерації додаткового інкрементального сигналу.

У однообертних датчиків інформація про координату абсолютної точки повторяється при кожному новому оберті, а багатообертні датчики дозволяють додатково розрізнити і кількість обертів.

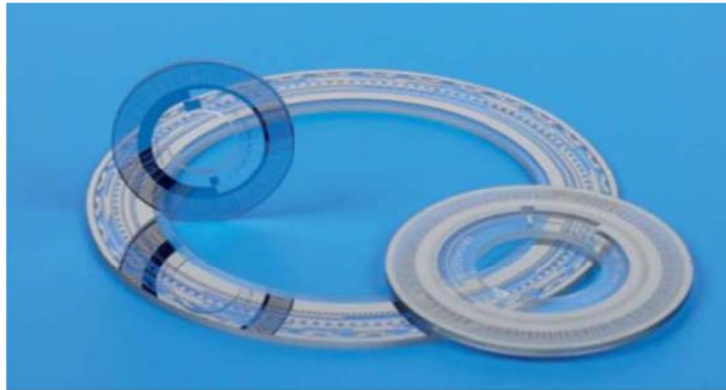


Рис. 4.15. Градуйовані диски датчиків обертання

При інкрементальному методі вимірювання шкала представляє собою послідовність штрихів з однаковим періодом. Координата положення визначається шляхом підрахунку окремих інкрементів (штрихів) від нульової точки, що задана в будь-якому місці шкали.

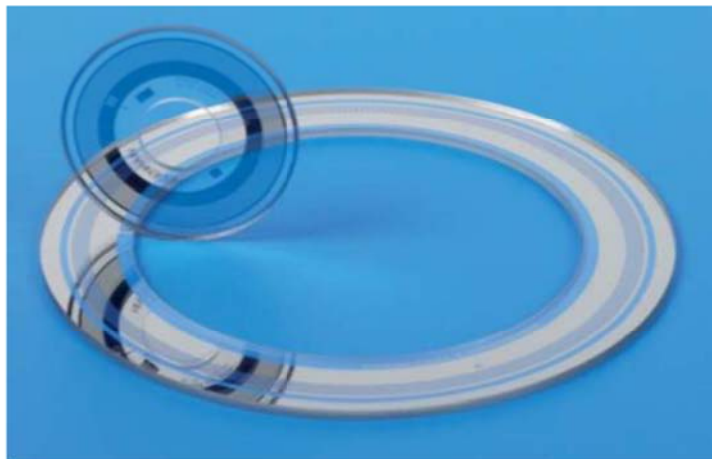


Рис. 4.16. Градуйовані диски інкрементальних датчиків обертання

Оскільки для визначення положення необхідна нульова точка підрахунку, то на градуйованих дисках передбачена окрема доріжка з референтною міткою. Період сигналу референтної мітки співпадає з періодом інкрементального сигналу.

Таким чином, перед тим як буде відновлена або встановлена заново нульова точка, повинна бути пройдена референтна точка.

Інкрементальні датчики обертання з числом штрихів від 6000 до 10000 забезпечують точність роботи системи в межах  $\pm 12$  кутових секунд.

У абсолютних датчиків обертання з додатковими інкрементальними сигналами точність залежить від кількості штрихів (для 512 штрихів точність  $\pm 60$  кутових секунд а для 2048 -  $\pm 20$  кутових секунд).

Також необхідно враховувати, що ці дані отримані при нормальній температурі в  $20^\circ$  і повільному обертанні.

Датчики кутових переміщень фірми HEIDENHAIN (рис. 4.17) мають високу точність (до кількох кутових секунд і вище). Зазвичай датчиками кутових переміщень називають датчики обертання з точністю  $\pm 5''$  і вище з кількістю штрихів більше 10000. Датчики кутових переміщень використовують в тих випадках, коли необхідно забезпечити точність вимірювання кута в межах декількох кутових секунд (поворотні столи, поворотні головки, ділильні головки, прецизійні установки для вимірювання кутів, антени, телескопи та інше).



Рис. 4.17. Датчики кутових переміщень фірми HEIDENHAIN

Точність датчиків кутових переміщень в межах від  $\pm 5''$  до  $\pm 0,4''$ , а кількість штрихів від 10000 до 180000. Для інкрементальних датчиків крок вимірювання  $0,018''$ , а для абсолютних датчиків біля 536млн позицій на оберт.

Датчики кутових переміщень реалізовані в декількох конструктивних варіантах.

Датчики з пустотілим валом, вбудованими підшипниками та муфтою статора RCN, RON та RPN мають хороші динамічні властивості. При кутових прискореннях муфта статора забезпечує компенсацію крутного моменту, що виникає внаслідок тертя в підшипниках. Ці датчики мають

невеликі габарити. Пустотілий вал може мати діаметр до 100мм, що дозволяє прокладати в них кабелі при монтажі.

Датчики з вбудованими підшипниками ROD без з'єднувальної муфти використовують для великих обертів або, якщо передбачені великі допуски, при монтажі. Використання прецизійних муфт підвищує осьові допуски спряження до  $\pm 1$ мм.

Датчики без підшипників ERP та ERA призначені для використання на елементах верстата. Вони мають великий діаметр пустотілого вала (варіант зі стрічкою до 10м) та допускають частоту обертання до 20000об/хв.

Точність вимірювання кутів залежить від декількох параметрів:

- якості штрихів;
- якості зчитування;
- ексцентриситету відносно підшипників;
- якості обробки сигналу;
- похибки радіального биття підшипників;
- гнучкості муфти статора для датчиків RCN, RON та RPN або муфти вала для датчиків ROD.

Точність вимірювання кута визначає точність позиціонування. При цьому гранично допустима похибка, що віднесена до середнього значення, не повинна виходити за межі точності системи  $\pm a$ .

Точність системи враховує похибку вимірювання як в межах одного оберту датчика (рис.4.18), що має місце при вимірюванні великих кутів, так і в межах одного періоду сигналу при вимірюванні невеликих кутів переміщення.

Таким чином, конструктивні особливості та якість виготовлення, а також ефективна система вимірювання гарантують високу точність датчиків та їх довговічність при експлуатації.

Системі ЧПК також необхідна інформація про швидкість, прискорення та уповільнення виконавчого органу. Розрахунок величини прискорення та уповільнення необхідний для точного позиціонування. Справа в тому, що коли робочий стіл переміщується в потрібну позицію, він при проході уповільнює швидкість переміщення, щоб не «проскочити» мимо потрібної координати.

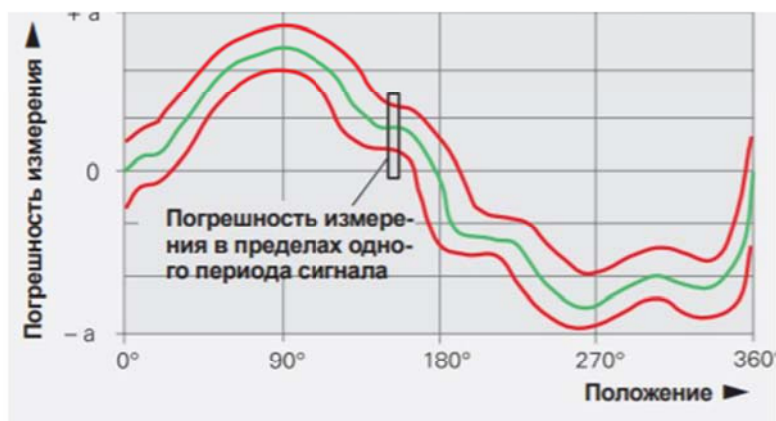


Рис. 4.18. Похибка вимірювання в межах одного обороту

Крім вищеперерахованих датчиків, використовуються і інші.

Температурні датчики (термопари) дозволяють визначити температуру виконавчих органів, розрахувати лінійне теплове розширення елементів верстата а також температуру масла та повітря.

Інфрачервоні датчики використовують в верстатних системах автоматичного вимірювання.

#### Запитання до розділу 4

1. Які задачі виконує система зворотного зв'язку?
2. Які датчики використовуються у верстатах з ЧПК?
3. Для чого використовуються лінійні датчики положення в сучасних верстатах з ЧПК?
4. Які спеціальні методи використовує фірма HEIDENHAIN для нанесення поділок в прецизійних шкалах?
5. Як компенсується в датчиках фірми HEIDENHAIN вплив підвищеної температури в закритому робочому просторі верстата та великі динамічні навантаження?
6. Охарактеризувати метод відображення для зчитування інформації з растрових шкал.
7. Охарактеризувати інтерферентний метод для зчитування інформації з растрових шкал.
8. Які особливості конструкції та переваги закритих лінійних датчиків фірми HEIDENHAIN?
9. Які похибки розрізняють при визначенні точності вимірювання лінійних датчиків закритого типу фірми HEIDENHAIN?
10. Які особливості конструкції та можливості датчиків обертання?
11. Охарактеризувати особливості конструкції та експлуатаційні можливості датчиків обертання фірми HEIDENHAIN.

## 5. Функціонування системи числового програмного керування

Система ЧПК працює наступним чином.

Програміст створює керуючу програму. Керуюча програма містить закодовану інформацію про траєкторії та швидкості переміщення виконавчих органів, частоту обертання шпинделя та інші дані, що необхідні для виконання обробки.

Підсистема керування читає цю програму, розшифровує її і розроблює профіль переміщення.

Профіль переміщення можна представити у вигляді графіка, що показує, в якій точці повинен знаходитись виконавчий орган верстата через конкретні проміжки часу.

У відповідності до профілю переміщення підсистема керування посилає на відповідний двигун строго визначену кількість електричних імпульсів. Двигун обертає ходовий гвинт і виконавчий орган переміщується у потрібну координату.

Датчики зворотного зв'язку посилають в підсистему керування інформацію про фактично досягнуте положення виконавчого органу.

Виконується порівняння фактичного і необхідного (теоретичного) положення. Якщо між ними є відмінність (похибка переміщення), то підсистема керування посилає скоректоване число електричних імпульсів на двигун.

Цей процес повторюється доти, поки виконавчий орган верстата не займе необхідну позицію з заданою достатньо високою точністю. Деяка похибка переміщення, звичайно, має місце, але вона повинна бути настільки малою, щоб її можна було не враховувати.

Це найпростіший варіант – переміщення вздовж однієї осі (рис. 5.1).



Рис.5.1. Переміщення робочого столу під кутом до осі



Якщо необхідно перемістити робочий стіл прямолінійно, але під кутом до осі, то для того щоб виконати таке переміщення система ЧПК змушена будувати між точками 1 та 2 множину опорних точок і рухати стіл по цих точках «ступінчасто». При цьому необхідно підтримувати таке співвідношення швидкостей переміщення по осях, щоб траєкторія руху відповідала заданій.

Роботу по розрахунку цих проміжних опорних точок виконує спеціальний пристрій, що входить до складу підсистеми керування – **інтерполятор**.

Інтерполятор безперервно, у відповідності до заданих переміщень, підтримує функціональний зв'язок між опорними точками та оцінює відхилення від заданої траєкторії, намагаючись звести їх до мінімуму.

Величина відхилення отриманої «ступінчастої» траєкторії рівна чи кратна ціні імпульсу системи ЧПК або імпульсу, що формується датчиком зворотного зв'язку.

Оскільки в сучасних верстатах розпізнавальна здатність системи ЧПК наближається до одного мікрметра, то отримане переміщення можна вважати плавним. Розглянута інтерполяція називається лінійною.

Якщо ж необхідно виконувати переміщення по дузі, то інтерполяція буде коловою (рис. 5.2). Виконується так звана лінійна апроксимація дуги, тобто заміна дуги малими прямолінійними відрізками. Виконавчий орган верстата також буде рухатись по «ступінчастій» траєкторії, яка візуально буде здаватись гладкою.



Рис.5.2. Схема кругової інтерполяції

Завдяки системі керування версти з ЧПК мають розширені технологічні можливості при збереженні високої надійності.

Для забезпечення ефективної роботи системи ЧПК використовують вісі координат та верстатну систему координат.

### 5.1. Система координат верстата з ЧПК

Система координат верстата з ЧПК є головною розрахунковою системою, що визначає переміщення виконавчих органів.

Вісі координат розташовують паралельно напрямним верстата, що дозволяє при створенні керуючої програми легко задавати напрямок та величини переміщень (рис. 5.3).

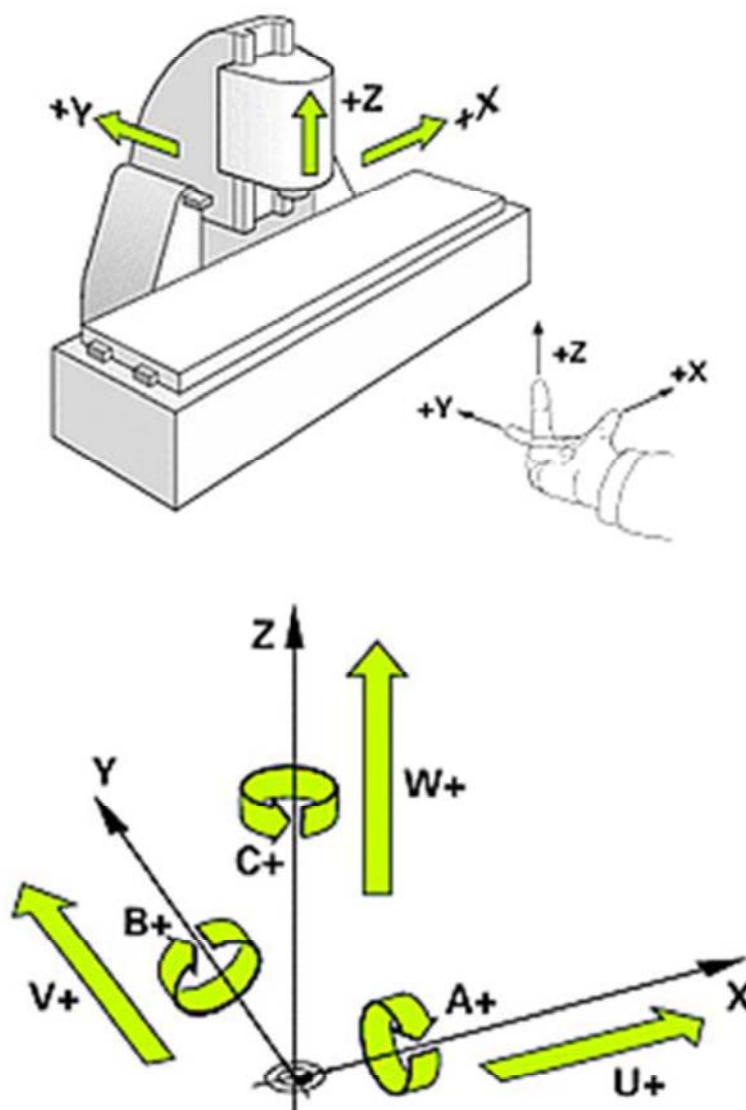


Рис. 5.3. Вісі координатної системи верстата

Стандартна система координат верстатів з ЧПК вибирається відповідно до рекомендацій Міжнародної організації зі стандартизації (ISO) і вона являє собою праву прямокутну систему координат, в якій позитивні напрямки осей координат визначаються правилом правої руки: великий палець вказує позитивний напрямок осі абсцис  $X$ , вказівний – осі ординат  $Y$  і середній – осі аплікат  $Z$  (рис. 5.4).

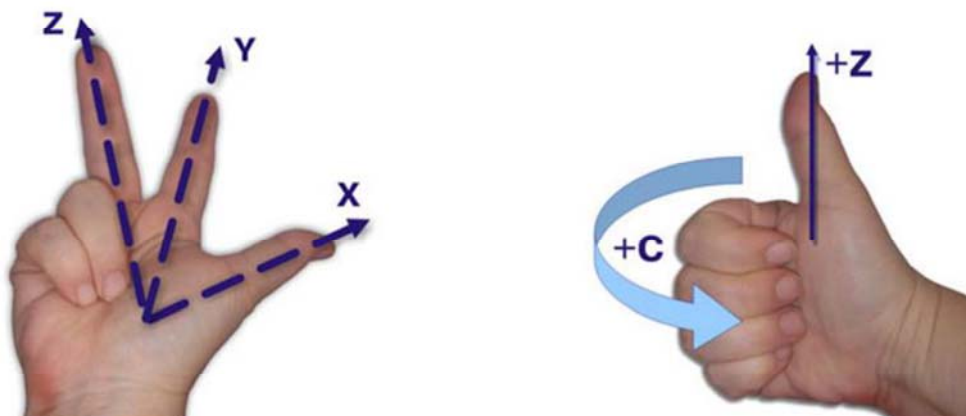


Рис.5.4. Правило правої руки

Особливість системи в тому, що вісь координати  $Z$  приймають завжди паралельно осі головного шпинделя верстата, незалежно від того, як він розташований – вертикально чи горизонтально.

Ця особливість дозволяє системі ЧПК використовувати у програмах позначення координат для найбільш поширеної плоскої обробки через  $X$  та  $Y$  незалежно від розташування шпинделя (рис. 5.5).

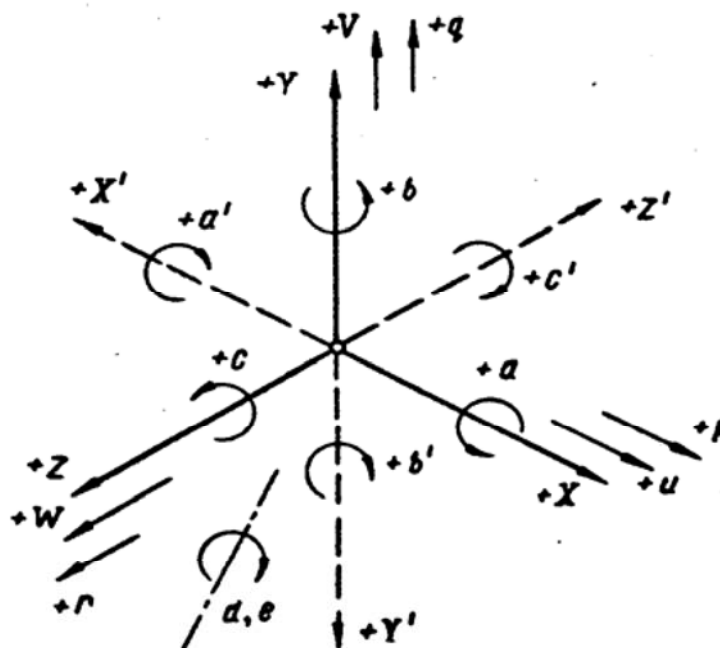


Рис.5.5. Права прямокутна система координат верстата

За позитивний напрямок осі  $Z$  приймають вертикальний напрямок виведення інструмента із заготовки (наприклад, свердла або фрези), тобто це напрямок від заготовки до інструменту. Таким чином вісь  $Z$  завжди зв'язана із шпинделем верстата.

Як правило, за  $X$  приймають вісь, вздовж якої можливе найбільше переміщення виконавчого органу.

Вісь  $X$  – завжди горизонтальна. При цьому вісь  $X$  перпендикулярна осі  $Z$  і паралельна площині робочого столу.

Визначивши осі  $Z$  та  $X$ , вісь  $Y$  визначають за правилом правої руки

Додаткові рухи, паралельні осям  $X$ ,  $Y$ ,  $Z$  позначаються відповідно  $U, V, W$  (вторинні) та  $P, Q, R$  (третинні).

Вісі  $X$ ,  $Y$ ,  $Z$  вказують позитивний напрямок переміщення інструменту відносно нерухомих частин верстата.

**При створенні керуючої програми програміст вважає, що інструмент переміщується відносно нерухомої заготовки, навіть якщо на верстаті відносно інструменту переміщується стіл із заготовкою.**

Система ЧПК сама визначить, в якому напрямку потрібно перемістити той, чи інший вузол верстата.

Крім лінійних переміщень конструкції деяких верстатів дозволяють виконувати і кругові переміщення.

Під круговим переміщенням розуміють, наприклад, поворот осі шпинделя фрезерного верстата. Однак саме робоче обертання шпинделя не входить в це поняття.

Кругові переміщення інструменту позначають латинськими буквами –  $A$  (навколо осі  $X$ ),  $B$  (навколо осі  $Y$ ),  $C$  (навколо осі  $Z$ ).

Позитивний напрямок обертання навколо цих осей визначають просто. Якщо розташувати великий палець в напрямку осі, то інші зігнуті пальці покажуть позитивний напрямок обертання.

Позитивні напрямки обертань  $A, B, C$  навколо координатних осей  $X, Y, Z$  показані на рисунку (рис. 15).

Для вторинних кутових переміщень навколо спеціальних осей використовуються букви  $D$  та  $E$ .

Початок стандартної системи координат верстата зазвичай поєднують з базовою точкою вузла, що несе заготовку і зафіксований в такому положенні, при якому всі переміщення робочих органів верстата описуються в стандартній системі позитивними координатами.

Положення виконавчих органів характеризують їхні базові точки, які вибираються з урахуванням конструкції верстата.

Наприклад, базовою точкою для шпинделя фрезерного верстата з ЧПК є точка перетину його торця з віссю обертання. Для робочого стола – точка перетину його діагоналей або один із кутів.

Системою координат токарного верстата є двохкоординатна система  $X, Z$  (рис. 5.6).

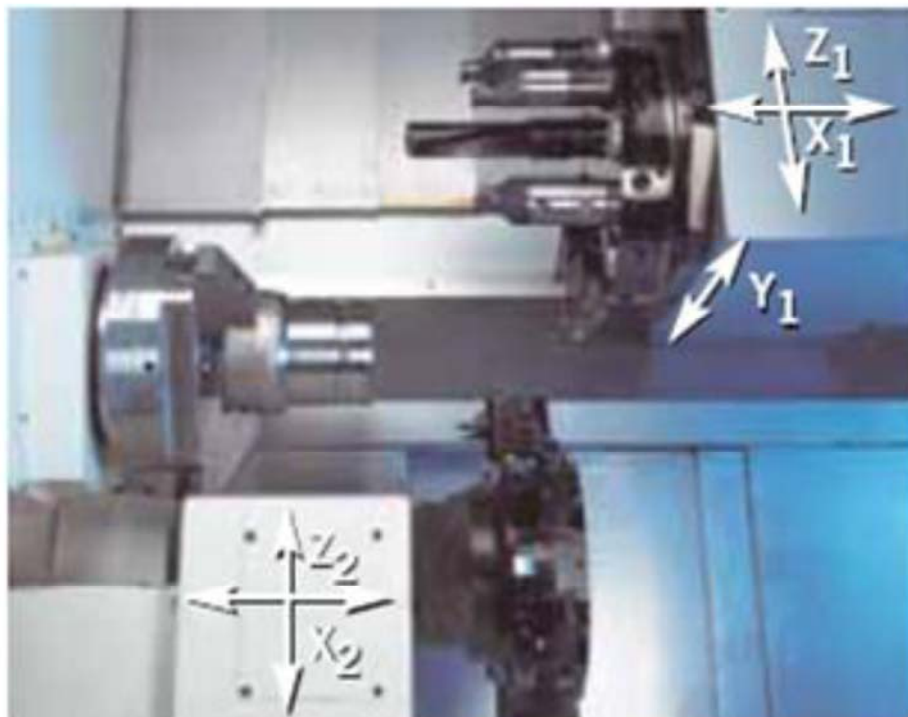


Рис.5.6. Робоча зона та системи координат токарного верстата з револьверною головкою

Початок цієї системи приймається в базовій точці шпиндельного вузла. Позитивні напрямки осей системи координат токарного верстата визначаються розташуванням основного робочого діапазону переміщень інструменту.

Для верстатів свердлильної, свердлильно-розточувальної та фрезерної груп застосовують трьохкоординатну систему  $X, Y, Z$  (рис. 5.7).

Початок цієї системи приймається в базовій точці столу, розташованого в одному з крайніх положень. Напрямки координатних осей цієї стандартної системи пов'язані з конструкцією верстата.

Система координатних осей робочих органів верстата являє собою сукупність окремих керованих за програмою координат, кожна з яких закріплена за конкретним робочим органом верстата і має індивідуальне позначення, напрямок і початок відліку.



Рис.5.7. Робоча зона та системи координат фрезерно-свердильного верстата

### Запитання до розділу 5

1. Як функціонує система ЧПК?
2. Які задачі виконує інтерполятор при переміщенні виконавчого органу під кутом до осі та по кривій?
3. Як розташовуються вісі координат по відношенню до верстата?
4. Як визначаються позитивні напрямки осей координат за правилом правої руки?
5. Як розташовується вісь Z-координати у верстаті?
6. Який напрямок осі Z приймають у верстаті за позитивний?
7. Як розташовується вісь X?
8. Який елемент системи верстата приймають за нерухомий при створенні керуючої програми – інструмент чи заготовку?
9. Якими буквами позначають кругові переміщення?
10. Як визначити позитивні напрямки обертань A, B, C навколо відповідних координатних осей за правилом правої руки?
11. Як вибираються базові точки шпинделів верстатів?
12. Які системи координат використовують в токарних та фрезерно-свердильних верстатах?

## **6. Нульова точка верстата та заготовки, напрямки переміщень**

### **6.1. Нульова точка верстата**

Нульова точка верстата – це фізична позиція, що встановлюється виробником верстата за допомогою відповідних датчиків.

Після включення верстата необхідно перемістити виконавчі органи в його нульову точку, щоб система числового програмного керування змогла визначити або «обнулити» їх машинну позицію. Іншими словами, необхідно синхронізувати систему числового програмного керування і верстат, тому, що в момент включення верстата система числового програмного керування ще не знає реального положення виконавчих органів і, якщо не виконати повернення в нульову точку, то верстат «відмовиться» працювати.

Процедура повернення в нуль верстата є стандартною і для її виконання кожний верстат має спеціальний режим та відповідні клавіші на панелі ЧПК.

Положення базової точки виконавчого органу відносно початку координат верстата з ЧПК (нульової точки верстата) називається позицією виконавчого органу в системі координат верстата або машинною позицією (machine - верстат).

В документації верстата граничні можливі переміщення робочих органів, як правило, вказують граничними зміщеннями базових точок, які і визначають максимально можливі габарити оброблюваної заготовки.

### **6.2. Нульова точка деталі та робоча система координат**

Для того, щоб обробити заготовку на верстаті необхідно відповідним чином установити робочу систему координат – нульову точку програми (деталі), оскільки при розробці керуючої програми це не виконується.

Задача оператора - визначити координати одного з кутів деталі і занести їх в реєстри робочих зміщень, наприклад, **G54**.

Після цього система числового програмного керування приймає цей кут за нульову точку для розрахунків усіх переміщень по програмі.

На відміну від фізичного нуля верстата, нульова точка програми(деталі) є логічною.

Як правило, робочу систему координат по осях **X** та **Y** встановлюють в один із кутів або центр деталі, а за нуль по осі **Z** приймають найвищу поверхню деталі.

Це полегшує програмісту виконання розрахунків, а оператору простіше «прив'язувати» систему координат та контролювати під час роботи переміщення інструменту.

Для знаходження машинної позиції нульової точки програми оператор визначає всі координати (**X**, **Y**, **Z**) в системі координат верстата і вводить ці координати в регістри робочих зміщень пам'яті системи числового програмного керування, використовуючи цифрові клавіші стійки ЧПК.

Під робочим зміщенням розуміють відстань від нуля верстата до нуля деталі вздовж якоїсь осі (рис. 6.1)

Сучасні системи числового програмного керування дозволяють запам'ятовувати багато зміщень. Завдяки декільком робочим системам координат програміст може виконувати одну і ту ж керуючу програму для обробки декількох заготовок, закріплених на столі.

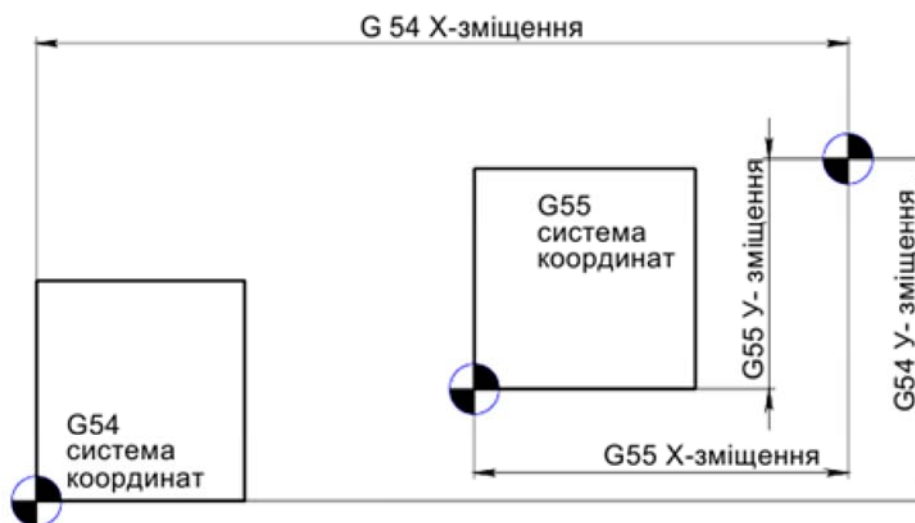


Рис. 6.1. Робочі зміщення деталей відносно нульової точки верстата

Після обробки однієї деталі система числового програмного керування зміщує координатну систему, тобто нульову точку програми до наступної деталі, що буде оброблюватись.

Для установки різних робочих систем координат використовуються відповідні **G** коди. Найчастіше, **G54** означає першу систему координат, **G55** – другу, **G56** – третю і т.д.



Запис **N20 G21 G54 G90** означає, що кадр **N20** активізує першу систему координат **G54**.

Нульова точка програми встановлюється для реалізації потрібної послідовності і повторюваності обробки.

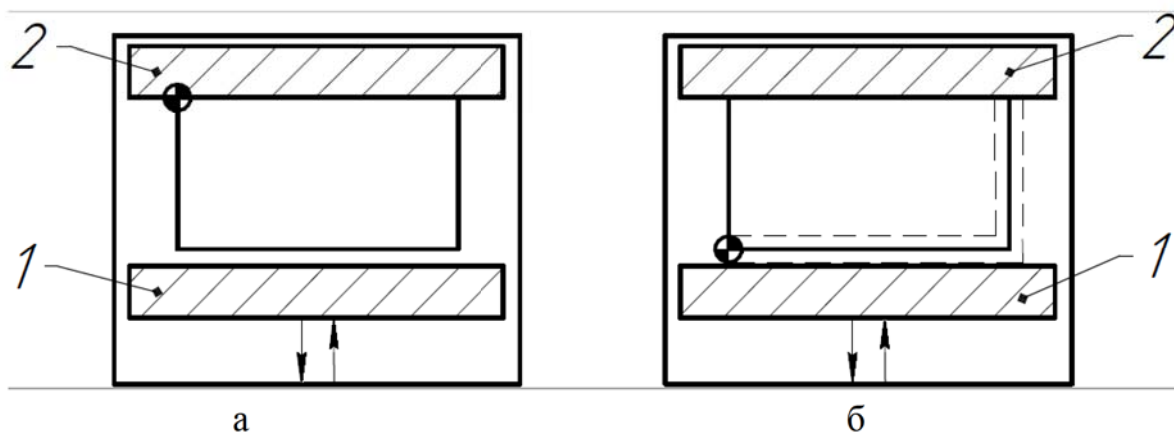


Рис.6.2. Два варіанти розташування нульової точки деталі в лещатах: а) нерухома губка - 2; б) рухома губка - 1, що виконує затиск заготовки.

Створення керуючої програми можна умовно розділити на два етапи.

**На першому етапі** технолог-програміст аналізує інформацію з конструкторської і технологічної документації та конструктивні і технічні можливості верстата з числовим програмним керуванням. На основі цієї інформації він остаточно визначає технологічні операції та маршрут обробки, вибирає різальний і допоміжний інструмент, визначає конструкторські та технологічні бази.

**На другому етапі** виконується остаточний розрахунок траєкторії інструменту по опорних точках та створення керуючої програми.

Виходячи з цього, при визначенні нульової точки програми використовують декілька правил.

**Перше**, але не основне – це зручність програмування.

Наприклад, якщо розташувати деталь в першому квадранті прямокутної системи координат, то це трохи спрощує процес розрахунку траєкторії, оскільки всі опорні **точки будуть описуватись позитивними координатами.**

**Друге** правило, більш важливе.

Нульова точка повинна співпадати з конструкторською базою.

Наприклад, якщо на кресленні розміри проставлені від лівого нижнього кута деталі, то краще, якщо в цьому куті буде знаходитись

нульова точка деталі, а якщо розміри вказуються від центрального отвору, то нуль розташовуємо в центрі отвору.

Бажано, щоб на верстат з ЧПК поступала заготовка з попередньо обробленими базовими поверхнями, що гарантує постійність координат нульової точки.

Бажано також зберігати постійність конструкторських і технологічних баз при установці заготовки на верстаті.

### 6.3. Компенсація довжини інструменту

При виконанні керуючої програми базова позиція шпинделя – це перетин торця та осі обертання, визначається запрограмованими координатами.

Проблема в тому, що обробка виконується кромкою різального інструменту, яка знаходиться на деякій відстані від базової точки шпинделя.

Для того, щоб в запрограмовану координату приходила різальна кромка, а не шпиндель, необхідно змістити базову точку на величину довжини інструменту по осі **Z**.

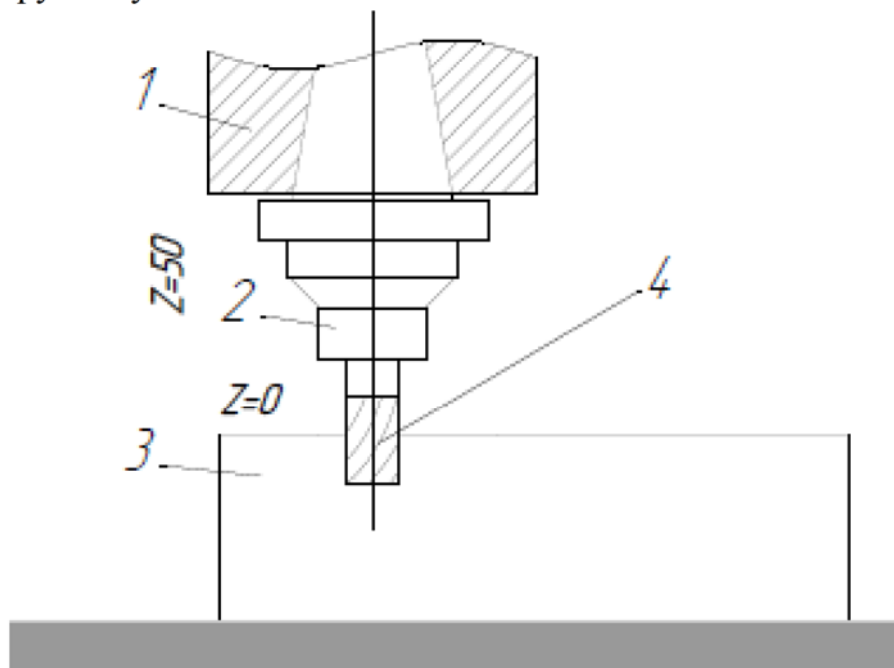


Рис. 6.3. Переміщення інструментального блоку 80мм в координату Z50 без компенсації довжини призведе до зіткнення інструменту із заготовкою: 1 – шпиндель верстата; 2 – затискний патрон; 3 – заготовка; 4 – різальний інструмент.

Перед початком обробки оператор повинен виміряти довжину кожного з інструментів і ввести числові значення довжин у відповідні **реєстри компенсації** довжини інструменту або в таблицю інструментів.

**Зміщення базової точки шпинделя на величину довжини інструмента і називається компенсацією довжини інструмента.**

Компенсація довжини інструменту на більшості сучасних верстатів активується командою **G43**, а відмінюється за допомогою **G49** або **H00**.

При створенні програми програміст використовує посилання на відповідний реєстр компенсації інструменту в пам'яті системи числового програмного керування.

Наприклад:

N025 G43 H02 Z50

H вказує на відповідний реєстр компенсації довжини

H02 – на реєстр інструмента №2

H03 – на реєстр інструмента №3 і т.д.

Це дає можливість оператору верстата змінюючи числові значення в реєстрі компенсації довжини добиватись потрібних розмірів деталі по осі Z.

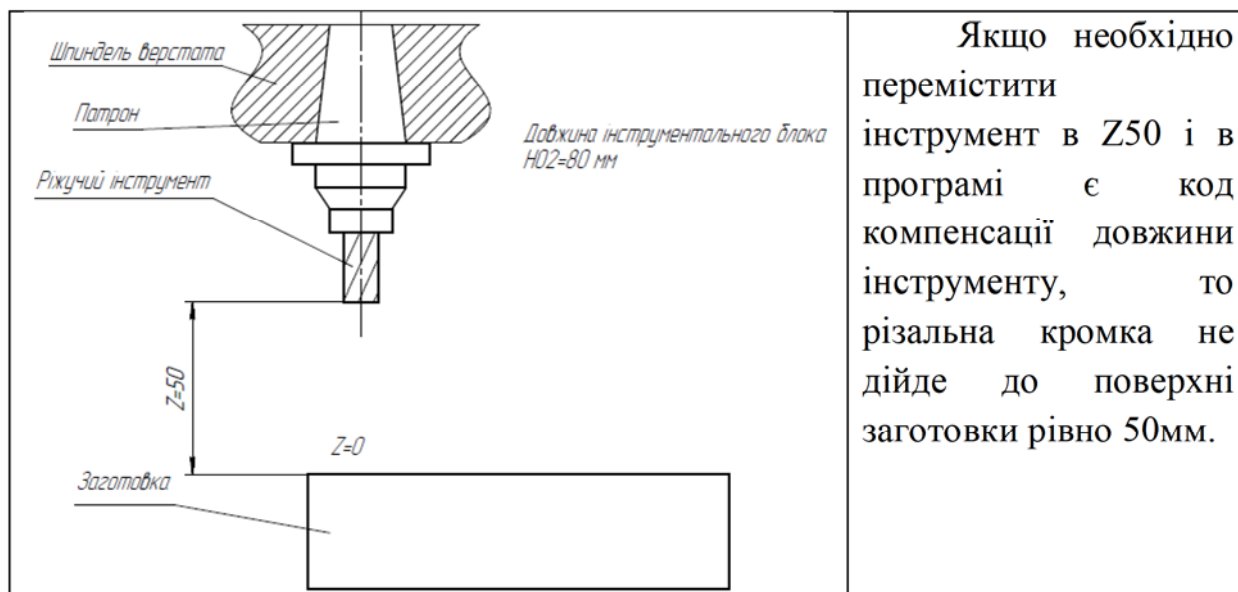


Рис. 6.4. Переміщення інструментального блоку 80мм в координату Z50 з компенсацією довжини

#### 6.4. Визначення робочої системи координат

Використовують декілька методів «прив'язування» координатної системи до деталі.

При **класичному методі** оператор використовує плоско паралельні кінцеві міри, калібри або індикатор, так званий центрошукач.

**Метод контакту** заснований на дотику різальним інструментом оброблюваної заготовки.

При **автоматичному методі** використовують спеціальний щуп та інфрачервоні датчики, які встановлюються в якості опції на сучасні оброблювальні центри.

Класичний метод є універсальним і може використовуватись на будь-якому верстаті з ЧПК. Оскільки «прив'язування» виконується в ручному режимі, то її точність невисока – в межах 0,02мм.

Метод достатньо простий. Шпинделем або калібром торкаються плоско-паралельної кінцевої міри довжини, що притиснута до поверхні деталі. Після нескладних підрахунків, поточні машинні координати заносяться вручну в реєстри робочих зміщень (**G54-G59**). Знаходження нульової точки складається з двох етапів: перший – по осі **Z** (рис.6.5), другий по осях **X** та **Y** (рис.6.6).

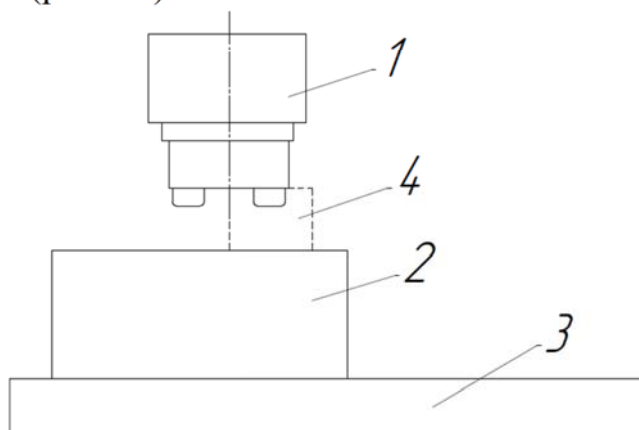


Рис.6.5. Визначення робочої системи координат по осі **Z**: 1 – шпиндель; 2 – заготовка; 3 – робочий стіл верстата; 4 – плоско паралельна кінцева міра

В сучасних верстатах з ЧПК найчастіше використовується автоматичний метод вимірювання радіусу та довжини інструменту з використанням відповідних щупів та датчиків. Конструкції щупів постійно удосконалюються з метою підвищення надійності, точності та швидкодії.

Фірма HEIDENHAIN пропонує два способи вимірювання інструменту: контактний за допомогою 3D-щупів серії TT та безконтактний з використанням лазерної системи TL.

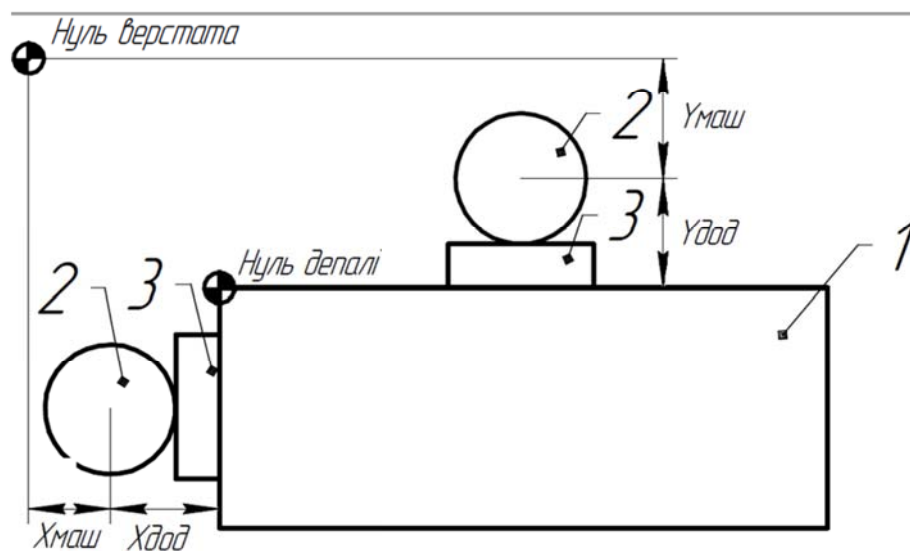


Рис.6.6. Визначення робочої системи координат по **X та Y**:  
1 – заготовка; 2 – циліндричний калібр; 3 - плоско паралельна кінцева міра

### 6.5. Контактний спосіб вимірювання інструменту

Для реалізації контактного способу розроблено вимірювальний щуп ТТ140, який дозволяє виконувати оцупування в трьох координатах.

Осьова сила торкання при цьому складає 8Н, а радіальна – 1Н. Вимірювальний стержень має головку у формі шайби, яка відхиляється при вимірюванні, генеруючи сигнал комутації, що передається в систему ЧПК і оброблюється там.

Щуп оснащений також двома світловими діодами для візуального контролю відхилення стержня з вимірювальною шайбою. Вимірювальна шайба має високу твердість, що дозволяє виконувати вимірювання при обертанні інструменту проти напрямку різання з частотою до 1000об/хв в залежності від діаметру інструмента, який контролюється.

В стержні передбачена точка надламування для захисту щупа від пошкодження при помилці оператора. Зламаний стержень швидко замінюється на новий без додаткового тестування щупа. Сигнал комутації створюється надійним безконтактним оптичним сенсором (чутливим елементом), що працює без зносу.

Світловий пучок, що створюється світловим діодом (LED), фокусується системою лінз в точку на диференціальному фотоелементі. При відхиленні вимірювального стержня диференціальний фотоелемент генерує комутаційний сигнал. Вимірювальний стержень з наконечником у

вигляді шайби жорстко з'єднується є перемикачем, який закріплений на корпусі в трьох точках і забезпечує надійне осьове орієнтування.

В поєднанні з циклами вимірювання системи ЧПК щуп ТТ140 (рис.6.7) дозволяє автоматичне вимірювання діаметру та довжини інструменту (рис. 6.8), результати якого записуються в таблицю інструментів. Це дозволяє при перевірці інструменту під час роботи вчасно реєструвати його знос або поломку. Система ЧПК блокує або автоматично замінює інструмент, параметри якого вийшли за допустимі межі.

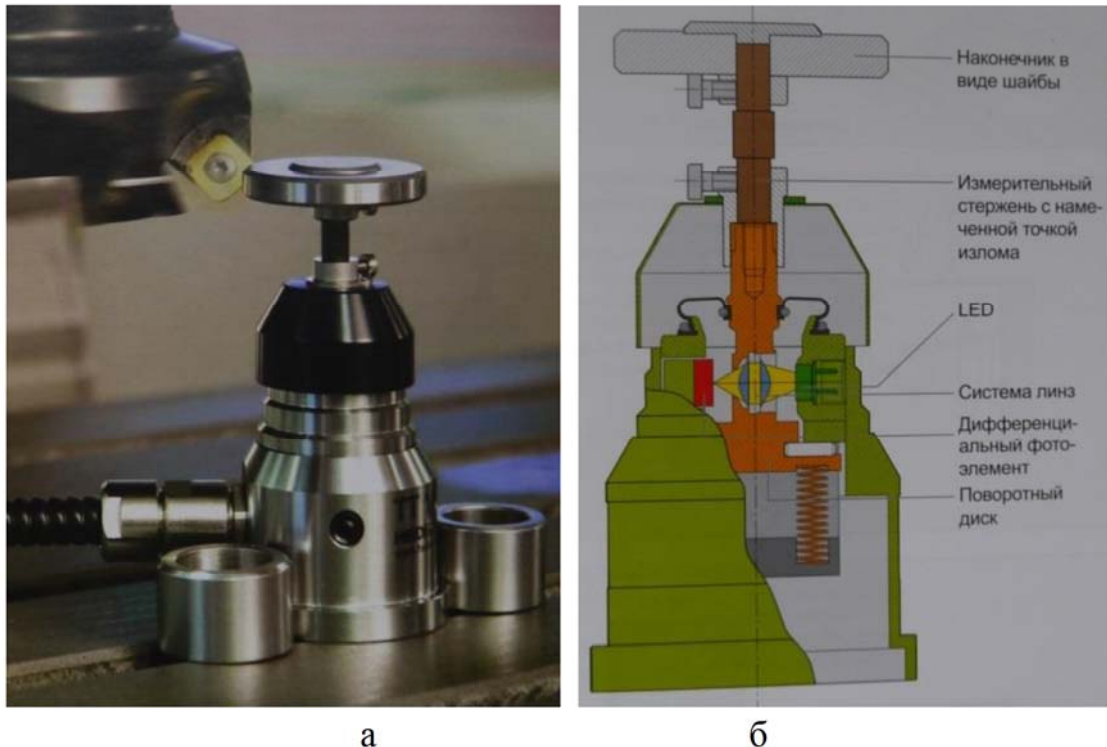


Рис. 6.7. Загальний вигляд (а) та принципова схема (б) щупа ТТ140

Для вимірювання заготовок безпосередньо на верстаті пропонуються 3D-щупи серії TS. В затискний патрон вони вставляються вручну або автоматично. Доторкнувшись до заготовки, вимірювальний стержень відхиляється в сторону і щуп створює комутативний сигнал, який через інфрачервоний передавач або по кабелю передається в систему ЧПК. Система ЧПК в цей момент зберігає фактичне положення вісі вимірювального приладу і оброблює отриманий сигнал. Сигнал комутації в щупах TS 2xx, TS 44x та TS 640 створюється оптичним сенсором (чутливим елементом), що працює без зносу і забезпечує таким чином високу надійність та стабільність їх роботи. Світловий пучок, що створюється світлодіодом LED, фокусується системою лінз в точку на диференціальному фотоелементі.

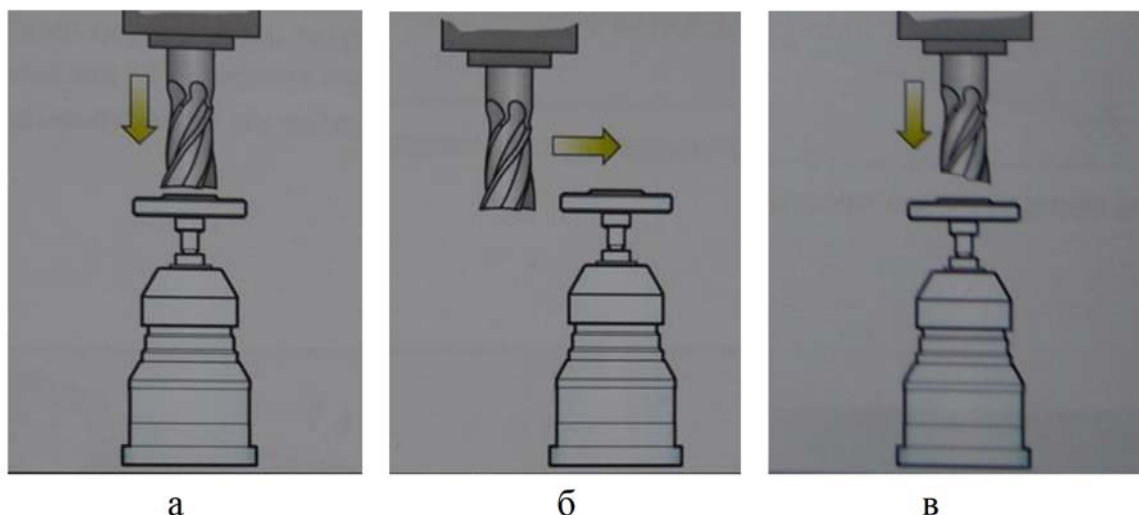


Рис. 6.8. Принципова схема вимірювання інструменту на верстаті: а- вимірювання довжини; б- вимірювання діаметру; в- контроль поломки

При відхиленні вимірювального стержня диференціальний фотоелемент генерує комутаційний сигнал. Вимірювальний стержень жорстко з'єднується з перемикачем, який закріплений на корпусі в трьох точках і забезпечує надійне орієнтування (рис. 6.9).

Щуп TS 740 використовує прецизійний датчик тиску (рис. 6.10). Комутаційний сигнал генерується шляхом аналізу діючої сили. Відхилення сили, що діє на датчики тиску при ощупуванні заготовки, оброблюються і генерується комутаційний сигнал. Цей спосіб дає можливість виконувати вимірювання в діапазоні 360°.

Відхилення вимірювального стержня щупа TS 740 визначається як середнє між показаннями кількох датчиків тиску, що розташовані між поворотним диском та корпусом датчика.

Завдяки невеликим силам, що діють при вимірюванні, можлива висока точність ( $\pm 5\text{мкм}$ ) і повторюваність результатів вимірювань (рис.6.11).

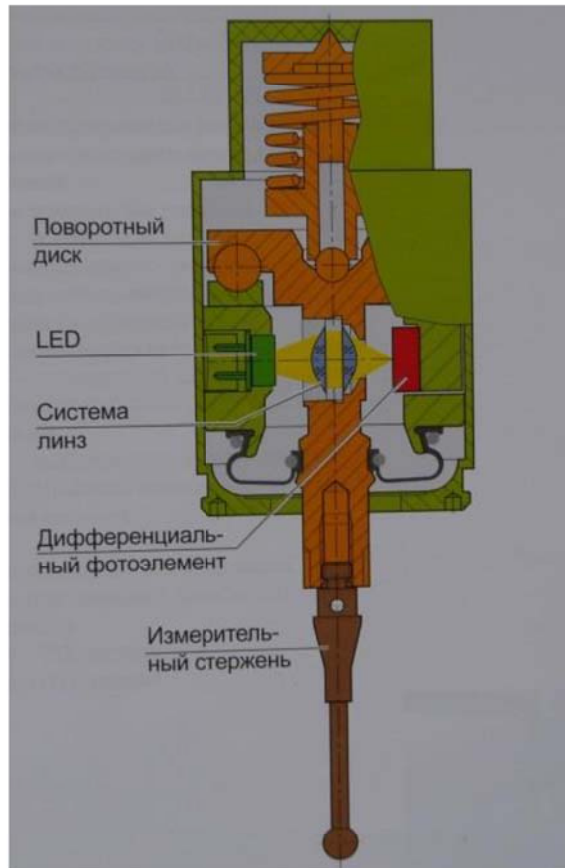


Рис. 6.9. Принципова схема безконтактного оптичного сенсора

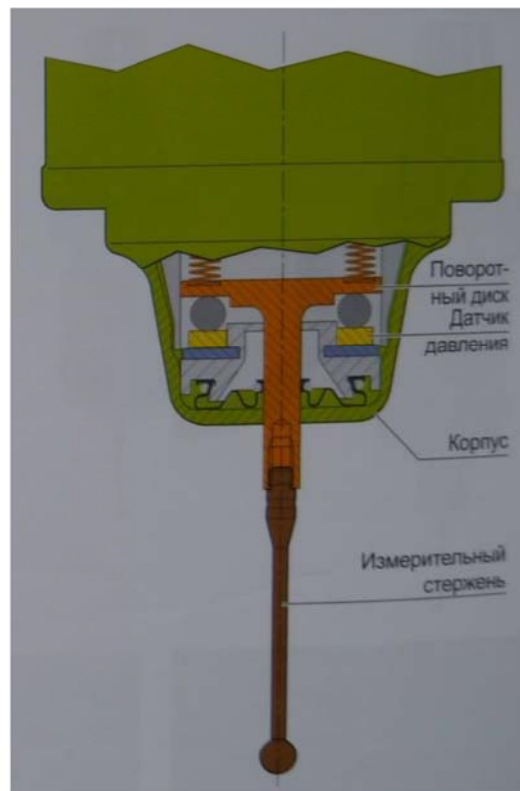


Рис. 6.10. Принципова схема датчика тиску в щупах



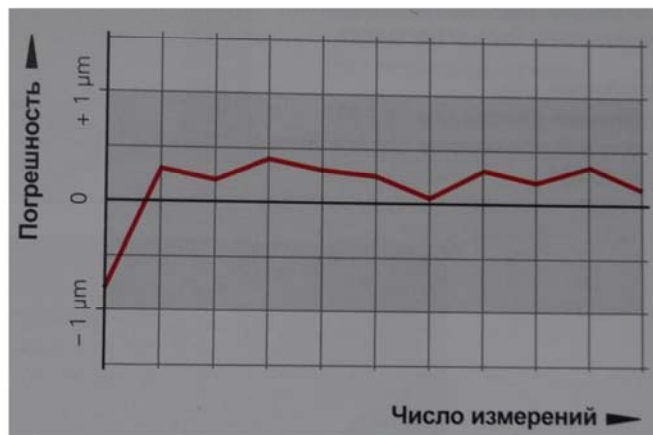


Рис. 6.11. Типова діаграма повторюваності результатів вимірювань 3D-щупа TS 2xx

Під повторюваністю розуміють похибку, що виникає при багаторазових вимірюваннях зразка в однакових умовах.

Визначення геометрії деталі або її положення за допомогою вимірювального щупа виконується шляхом механічного оцупування. При цьому щуп повинен бути чистим, щоб уникнути помилок при вимірюванні.

Тому вимірювальні щупи з інфрачервоним передавачем оснащуються системою обдування стисненим повітрям або змащувально-охолоджувальною рідиною через три отвори в нижній частині щупа, що дозволяє видаляти частки бруду з вимірюваної поверхні і елементи стружки.

Така система дозволяє повністю автоматизувати виробничий процес. На верстаті повинно бути передбачене підведення стисненого повітря або ЗОР.

Необхідно враховувати також, що надійність вимірювань залежить від швидкості передачі сигналу та величини максимального відхилення вимірювального стержня, Допустима швидкість оцупування задається в технічних характеристиках щупа а максимальне допустиме відхилення складає 5мм в кожному напрямку (рис.6.12).

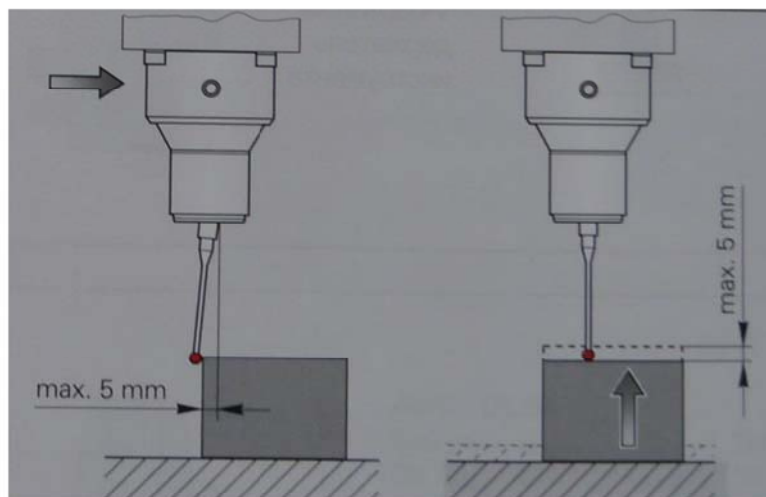


Рис. 6.12. Граничні відхилення вимірювального щупа

Точність вимірювання в значній мірі залежить і від конструкції вимірювальних стержнів. Для щупів TS фірмою розроблені вимірювальні стержні довжиною від 21мм до 40мм з рубіновими кульками діаметром від 1мм до 8мм (рис. 6.13). Усі вимірювальні стержні з'єднуються зі щупом різьбою, а стержні з діаметром кульки більше 4мм мають точку надламування для захисту щупа від пошкодження. Також передбачені подовжувачі для вимірювань глибоких порожнин.



Рис. 6.13. Вимірювальні стержні з рубіновими кульками

При використанні вимірювального щупа точне розташування заготовки вздовж осей верстата не обов'язкове. Він визначає відхилення при ошупуванні заготовки а система ЧПК компенсує це відхилення обертанням координатних осей або обертанням поворотного столу (рис.6.14).

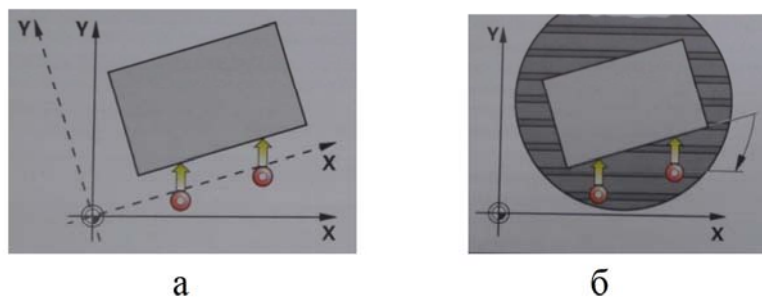


Рис. 6.14. Компенсація відхилення заготовки від напрямку координатних осей: а- обертанням координатних осей; б- обертанням поворотного столу

Програми оброблювання на верстатах з ЧПК прив'язуються до опорних точок, положення яких швидко і точно визначаються за допомогою вимірювального щупа (рис. 6.15), що суттєво економить час і підвищує точність оброблювання. Можливе і автоматичне визначення опорної точки в залежності від циклів вимірювання ЧПК.

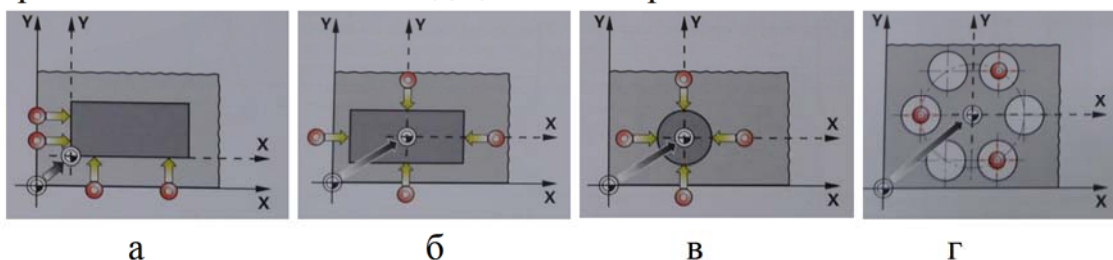


Рис. 6.15. Визначення опорних точок для за допомогою вимірювального щупа: а – зовнішній кут; б – центр прямокутної виконаних по колу

3D-щупи TS дозволяють виконувати вимірювання заготовки між двома циклами оброблювання в автоматизованому режимі, використовуючи результати вимірювань для компенсації зносу інструменту (рис.6.16).

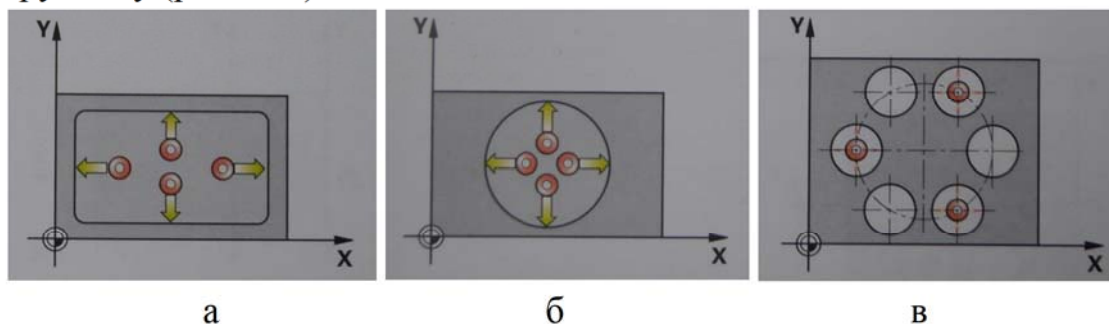


Рис. 6.16. Приклади вимірювання розмірів заготовок на верстаті: а –прямокутної кишені; б – круглої кишені; в – отворів по колу

## 6.6. Безконтактний спосіб вимірювання інструменту

Лазерні системи TLMicro та TLNano (рис. 6.17) дозволяють проводити безконтактне вимірювання інструменту при його обертанні з номінальною швидкістю в двох координатах:  $\pm X$ ,  $+Z$  та  $\pm Y$ ,  $+Z$ , що виключає можливість пошкодження зубів інструменту.

Не дивлячись на те, що вимірювання виконуються при номінальній швидкості, контролюється кожний зуб інструменту та геометрія спеціального інструменту. Це дозволяє вчасно виявляти пошкодження та максимальний знос і добиватись постійної якості оброблювання.

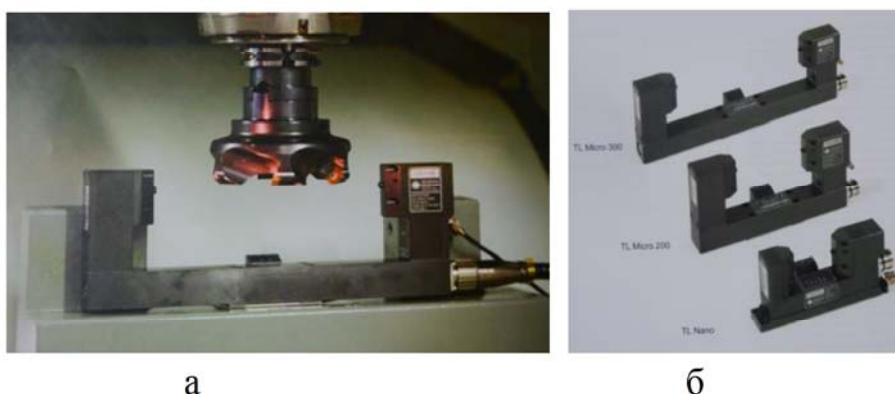


Рис. 6.17. Лазерні системи вимірювання інструментів TL: а – на верстаті під час вимірювання; б – конструкції для різних розмірів інструменту

Цикли вимірювання дозволяють вимірювати довжину та діаметр інструменту, контролювати форму окремих зубів, знос, знаходити тріщини і фіксувати пошкодження (рис.6.18).

Результати вимірювань система ЧПК оброблює, виконує необхідні розрахунки і записує в таблицю інструментів. Вимірювання виконуються достатньо швидко і просто за допомогою програми, яка позиціонує вимірювальний щуп і починає цикл вимірювання, який може бути проведений в любий момент – між двома переходами або після завершення оброблювання заготовки. Сфокусований промінь лазера вимірює інструмент діаметром від 0,03мм. Відхилення розмірів при повторних вимірюваннях знаходяться в межах  $\pm 1\mu\text{м}$ .

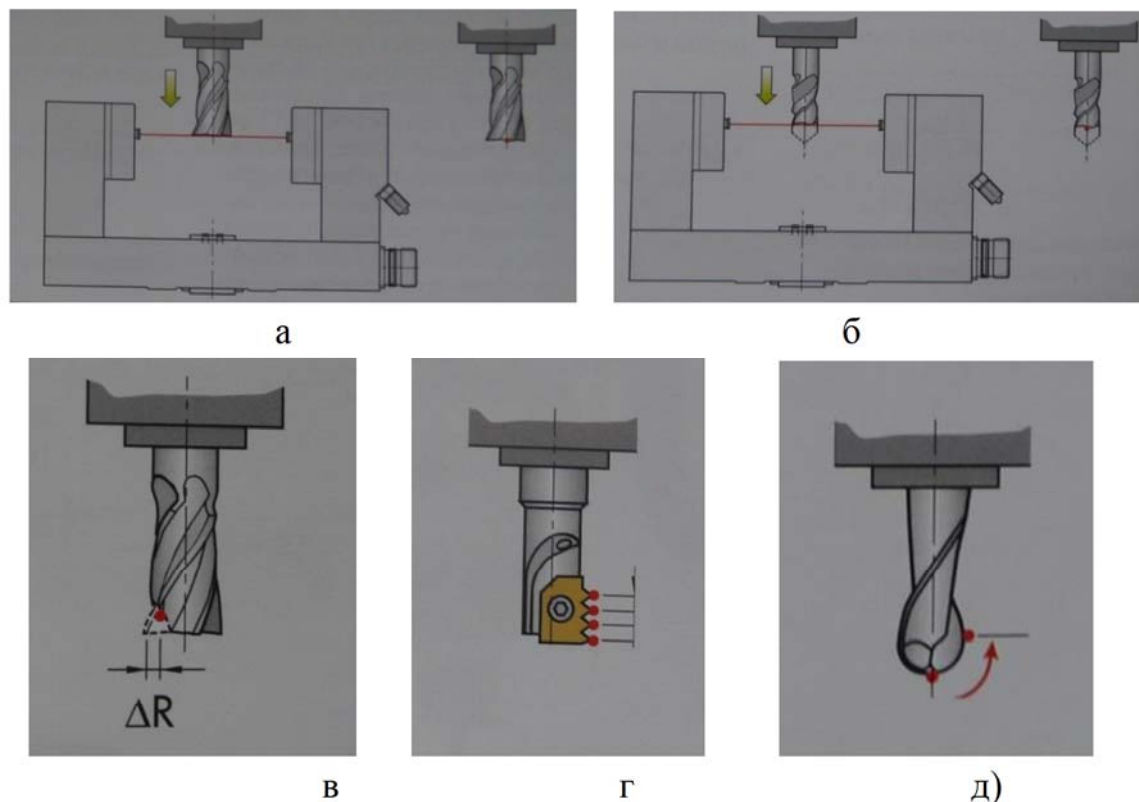


Рис. 6.18. Безконтактний метод вимірювання лазерною системою TL: а – вимірювання довжини інструменту; б – розпізнавання полочки; в - вимірювання радіусу інструменту та розпізнавання пошкоджених зубів; г- контроль окремих зубів; д - контроль форми зубів

Для досягнення найкращої повторюваності результатів вимірювань лазерна система повинна встановлюватись строго паралельно до двох осей. Інструмент перед вимірюваннями необхідно очистити обдуванням стиснутим повітрям або змащувально-охолоджувальною рідиною, що гарантує високу надійність вимірювань. Для цього усі лазерні системи оснащені вбудованою системою обдування. Оптика системи захищена від попадання ЗОР та стружки діафрагмою з вбудованою системою закривання. Діафрагма відкривається стисненим повітрям тільки під час вимірювань.

Передавач та приймач лазерного променя обдувається спеціально очищеним повітрям, що захищає від появи конденсату. На приймачі додатково розміщені світлові діоди, для забезпечення візуального контролю стану лазерної системи вимірювань. Це дозволяє оператору миттєво визначати її робочий стан, в якому режимі вона знаходиться та чи передає в даний момент динамічний сигнал.

Ощупування інструменту (6.19) виконується в наступній послідовності. Від джерела випромінювання лазерний промінь попадає на приймач. При кожній зміні стану системи під час вимірювання інструменту, наприклад, відсутності випромінювання на приймачі або при його появі генерується динамічний комутаційний сигнал DYN тривалістю 20с, що оброблюється системою ЧПК верстата. У випадку поломки (рис. 6.18,б), тобто відсутності зуба або виходу за межі допуску, динамічний вихідний сигнал DYN приймає низький рівень до 100с. Додатково для визначення часу відсутності сигналу на приймачі лазерна система видає статичний сигнал STA.

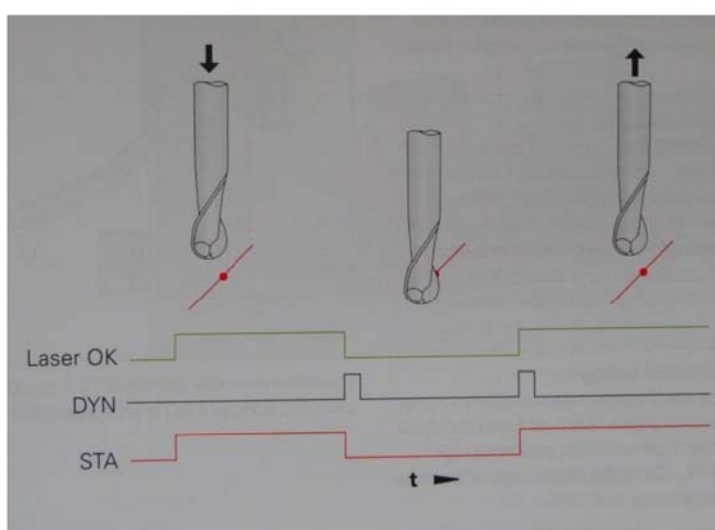


Рис. 6.19. Схема ощупування інструменту лазерною системою

Важливе значення має напрямок вимірювань, який суттєво впливає на надійність та точність. Розрізняють два види вимірювань: при підведенні до та при відведенні інструменту від лазерного променя. Вимірювання при відведенні інструменту забезпечує надійний захист від залишків ЗОР та інших забруднень. Вимірювання при підведенні інструменту більше підходить для інструментів малого діаметру.

Перед початком вимірювань обов'язково необхідно калібрувати лазерну систему, тобто визначити точні координати точок вимірювання в системі координат верстата за допомогою еталону у вигляді циліндра зі скошеним краєм для проведення вимірювань в обох напрямках осі Z. Циліндр вставляється в затискний патрон шпинделя і вимірюється його довжина, діаметр та висота над столом в системі координат верстата.

## **Запитання до розділу 6**

1. Що таке нульова точка верстата?
2. Що таке позиція виконавчого органу в системі координат верстата?
3. Що таке нульова точка деталі?
4. Що розуміють під робочим зміщенням нуля деталі?
5. Які G-коди використовуються для установки різних систем координат верстата?
6. Які основні два етапи створення керуючої програми програмістом?
7. Які правила визначення нульової точки деталі?
8. Для чого необхідна компенсація довжини інструменту?
9. Яка послідовність виконання процедури компенсації на довжину інструмента?
10. В чому різниця використання абсолютних та відносних координат?
11. Які особливості контактного способу вимірювання інструменту за допомогою 3D-щупів серії TT фірми HEIDENHAIN?
12. Які особливості процесу вимірювання заготовок безпосередньо на верстаті за допомогою 3D-щупів серії TS фірми HEIDENHAIN?
13. Які конструктивні особливості вимірювальних стержнів для щупів серії TS фірми HEIDENHAIN?
14. Як проводиться безконтактне вимірювання інструменту при його обертанні з номінальною швидкістю в двох координатах за допомогою лазерних систем TLMicro та TLNano?

## 7. Основи програмування обробки на верстатах з числовим програмним керуванням

Математичною базою для програмування механічної обробки є прямокутна система координат, так звана Декартова система координат, що придумана французьким математиком більш ніж 300 років тому.

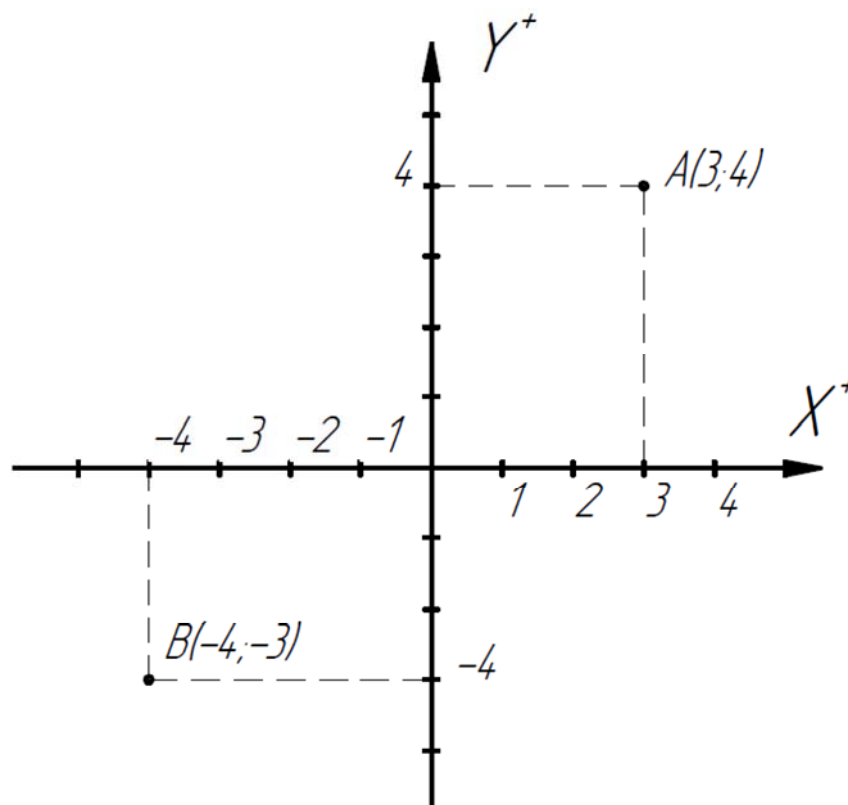


Рис.7.1. Прямокутна система координат на площині

Координатна система з двома осями  $X$  та  $Y$  дозволяє описати положення точки на площині (рис. 7.1).

Відстань від початку координат до точки  $A$  вздовж осі  $X$  є  $(x)$  координатою, а вздовж осі  $Y$  –  $(y)$  координатою цієї точки.

Координати точки прийнято показувати в дужках, спочатку по осі  $X$ , а потім по осі  $Y$ .

Таким чином, на рисунку знаходиться точка  $A(3;4)$ .

Кожна вісь має позитивний та від'ємний напрямки.

Координата має від'ємне значення, коли точка лежить лівіше початку координат або нижче  $B(-4;-3)$ . Якщо точка лежить на осі, то одна із координат дорівнює нулю.



Перетин трьох взаємно перпендикулярних площин створює трьохмірну систему координат (рис. 7.2), яка використовується для описування положення точки в просторі.

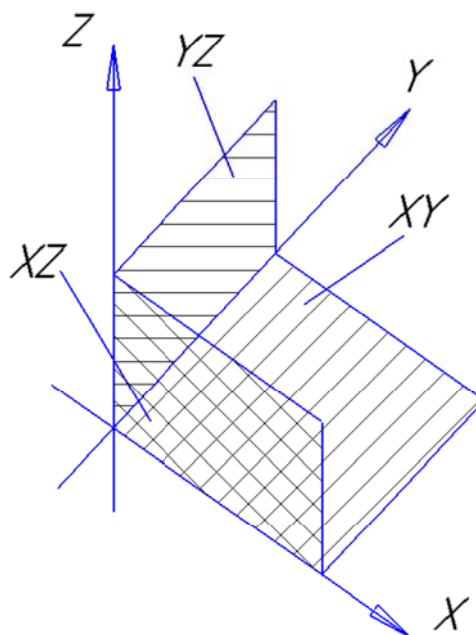


Рис.7.2. Прямокутна просторова система координат

Координати точки позначаються в дужках в алфавітному порядку  $(x,y,z)$ .

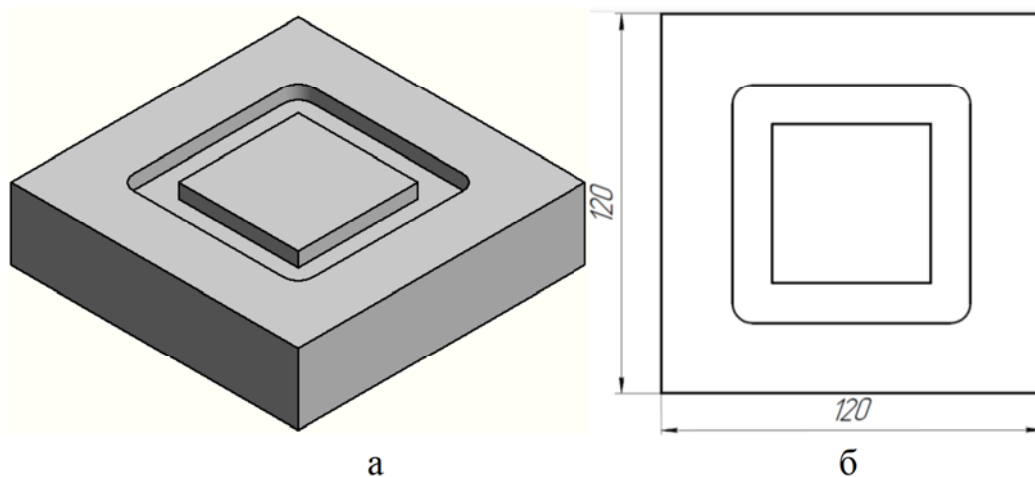
Деталі, що оброблюються на верстаті з ЧПК можна розглядати як геометричні об'єкти.

Керуюча програма описує рух центру інструмента при його відносному переміщенні до заготовки.

Траєкторія інструмента складається з окремих взаємопов'язаних ділянок. Цими ділянками можуть бути прямі лінії, дуги кола, криві другого та вищих порядків.

Точки перетину цих ділянок називаються опорними або вузловими. Як правило, в керуючій програмі і знаходяться координати опорних точок, тому для створення програми обробки необхідно визначити координати усіх опорних точок. Знаючи координати опорних точок, написати програму відносно просто.

Наприклад, необхідно створити програму для обробки паза (рис. 7.3).



а б  
 Рис.7.3. Ескіз деталі з пазом глибиною 3мм

Деталь розташовують в прямокутній системі координат та знаходять координати опорних точок (рис. 7.4).

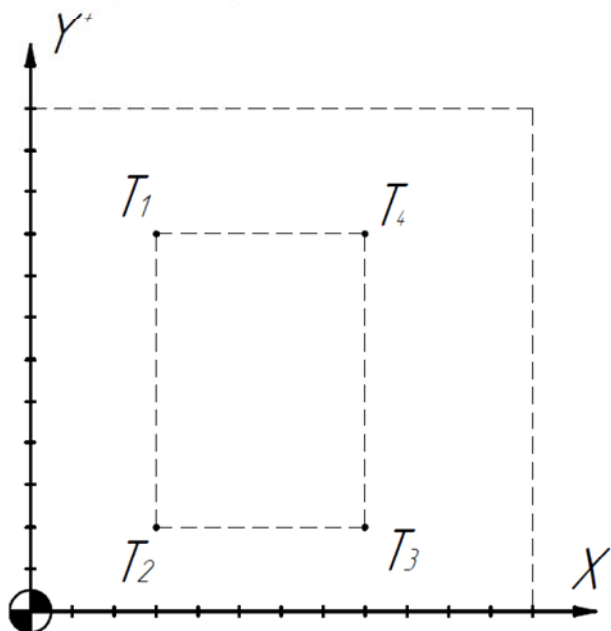


Рис.7.4. Координати чотирьох опорних точок

Для зручності координати опорних точок заносять в таблицю 7.1.

Таблиця 7.1

Координати опорних точок

Точка	Координата по осі X	Координата по осі Y
T1	3	9
T2	3	2
T3	8	2
T4	8	9

Найбільшу увагу приділяють написанню рядків(кадрів) керуючої програми, що безпосередньо відповідають за переміщення через опорні точки паза.

Для обробки паза необхідно перемістити фрезу в точку Т1, опустити її на відповідну глибину і перемістити послідовно через усі опорні точки, а потім підняти ввєрх.

Підводимо різальний інструмент до першої опорної точки.

N50 G00 X3 Y9

Наступні два кадри заставляють інструмент опуститись на потрібну глибину.

N60 G00 Z0.5

N70 G01 Z-3F25

Для переміщення інструменту через усі опорні точки використовуємо наступні три кадри:

N80 G01 X3 Y2

N90 G01 X8 Y2

N100 G01 X8 Y9

N110 G01 X3Y9

Тепер необхідно вивести інструмент із матеріалу заготовки на невелику висоту

N110 G01 Z5

Якщо всі основні кадри зібрати разом та додати допоміжні кадри, то отримаємо остаточний варіант керуючої програми для обробки паза (табл.7.2).

Таблиця 7.2

Керуюча програма для обробки паза

Кадри керуючої програми	Пояснення
%	Символ початку програми
O0001 (PAZ)	Номер програми(0001) та її назва(PAZ)
N10 G021 G40 G49 G54 G80 G90	Рядок безпеки
N20 M06 N01 (FREZA D1)	Виклик інструменту №1
N30 G43 H01	Компенсація довжини інструменту №1
N40 M03 S1000	Включення обертів шпинделя (1000 об/хв)
N50 G00 X3 Y9	Прискорене переміщення в опорну точку Т1
N60 G00 Z0.5	Прискорене переміщення в Z0.5
N70 G01 Z-3 F25	Переміщення на глибину 3мм при подачі 25мм/хв
N80 G01 X3 Y2	Переміщення в точку Т2 з подачею 25мм/хв

Кадри керуючої програми	Пояснення
N90 G01 X8 Y2	Переміщення в точку T3 з подачею 25мм/хв
N100 G01 X8 Y9	Переміщення в точку T4 з подачею 25мм/хв
N110 G01 X3Y9	Переміщення в точку T1 з подачею 25мм/хв
N120 G01 Z5	Піднімання інструменту ввєрх в Z5 з подачею 25мм/хв
N130 M05	Вимикання обєртїв шпинделя
N140 M30	Завєршення програми
%	Символ кінця програми

### Запитання до розділу 7

1. Яка система координат є математичною базою для програмування механічної обробки?
2. Як визначається позитивний та від'ємний напрямок вісі координат?
3. Для чого використовується трьохмірна система координат?
4. З чого складається траєкторія інструменту?
5. Які точки називаються опорними або вузловими?
6. Як використовуються координати опорних точок для написання програми оброблювання?

## 8. Етапи створення керуючої програми

### 8.1. Абсолютні та відносні координати

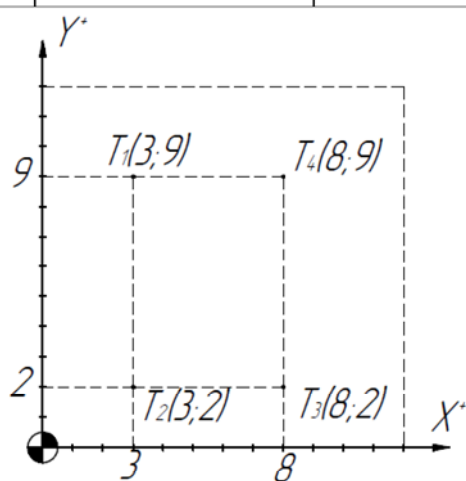
При **абсолютному** способі програмування координати точок відраховуються від постійного початку координат.

При **відносному (інкрементному)** способі підрахунку за нульове положення кожний раз приймається положення виконавчого органу, яке він займав перед початком переміщення до наступної опорної точки.

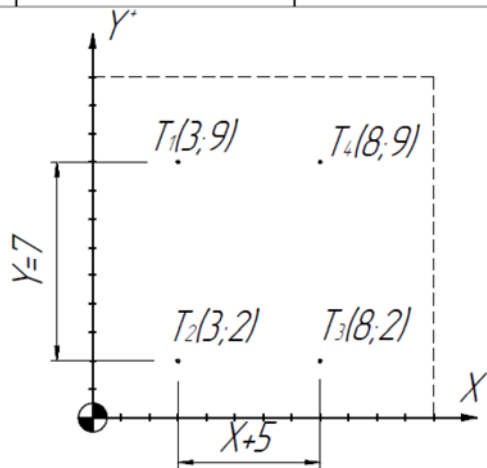
Таблиця 8.1

Опорні точки паза при абсолютному та відносному програмуванні

Точка	Абсолютні координати		Відносні координати	
	Координати по осі X	Координати по осі Y	Координати по осі X	Координати по осі Y
T1	3	9	3	9
T2	3	2	0	-7
T3	8	2	5	0
T4	8	9	0	7
T1	3	9	5	0



а



б

Рис.8.1. Абсолютний (а) та відносний (б) методи програмування

Приклад програм, де видно як змінюються координати опорних точок паза і програма обробки при відносному способі програмування, представлено в табл. 8.2.

Таблиця 8.2

Абсолютне та відносне програмування

Абсолютне програмування <b>G90</b>	Відносне програмування <b>G91</b>
%	%
O0001(PAZ ABS)	O0001(PAZ INCR)
N10 G21 G40 G49 G54 G80 G90	N10 G21 G40 G49 G54 G80 G91
N20 M06 T01 (FREZA D1)	N20 M06 T01 (FREZA D1)
N30 G43 H01	N30 G43 H01
N40 M03 S1000	N40 M03 S1000
N50 G00 X3 Y9	N50 G00 X3 Y9
N60 G00 Z0.5	N60 G00 Z0.5
N70 G01 Z-3 F25	N70 G01 Z-3 F25
N80 G01 X3 Y2	N80 G01 X0 Y-7
N90 G01 X8 Y2	N90 G01 X5 Y0
N100 G01 X8 Y9	N100 G01X0 Y7
N 110 G01 X3 Y9	N110 G01 X5 Y0
N120 G01 Z0.5	N120 G01 Z0.5
N130 G90 G28 X0 Y0 Z0	N130 G91 G28 X0 Y0 Z0
N140 M05	N140 M05
N150 M30	N150 M30
%	%

Програмування у відносних координатах було обов'язковою умовою при роботі на верстатах з ЧПК старих моделей.

В сучасних верстатах системи числового програмного керування дозволяють вільно працювати з двома способами програмування.

Для переключення із одного режиму в інший використовують команди:

**G90** – абсолютне програмування.

**G91** – відносне програмування.

**Необхідно пам'ятати**, що у випадку появи однієї координатної помилки при відносному способі програмування усі наступні переміщення будуть неправильними.

## 8.2. Створення керуючої програми на ПК

Запис (набирання) керуючих програм виконують за допомогою стійки ЧПК верстата (цехове програмування) або використовуючи персональний комп'ютер з наступною передачею програми в стійку ЧПК.

Код керуючої програми можна набирати в будь-якому текстовому редакторі і зберігати його у відповідному форматі.

Створені і спеціальні текстові редактори для роботи з кодом керуючої програми. Вони мають додаткові функції, які дуже корисні при створенні та доопрацюванні програми обробки.

Сучасні редактори керуючих програм мають інструментарій для графічної перевірки коду і трансляції його на верстат.

Верстати з числовим програмним керуванням працюють у форматі **Gта M-кодів** у відповідності до стандарту **ISO**.

**Система ЧПК вимагає, щоб в кінці кожного кадру керуючої програми стояв знак кінця кадру, наприклад, (;) або (\*).**

Тому, якщо програма пишеться на комп'ютері, то вона виглядає так:

```
...  
N50 G00 X150  
N60 G01 Y200  
N70 G01 Z100  
...
```

Якщо ж ця програма вводиться прямо зі стійки ЧПК, то прийдеться додати символ кінця кадру в кожний рядок керуючої програми.

```
...  
N50 G00 X150;  
N60 G01 Y200;  
N70 G01 Z100;  
...
```

Якщо програма створюється на комп'ютері а потім передається на верстат, то знак кінця кадру поміщається в кінець кожного кадру керуючої програми при передачі у більшості випадків автоматично.

**Необхідно пам'ятати**, що помилка в програмі обробки може викликати масу проблем – в кращому випадку поломка інструменту та пошкодження заготовки а в гіршому – пошкодження верстата або травма оператора.

Тому безпечніше, простіше і дешевше перевірити програму на ПК.

### 8.3. Методи перевірки керуючої програми

Основним методом перевірки керуючої програми є графічне відтворення обробки у вигляді промальовування траєкторії центру інструмента або повної імітації механічної обробки на верстаті з демонстрацією процесу видалення матеріалу.

В першому випадку програміст може визначити помилку в керуючій програмі, просто слідкуючи за траєкторією переміщення центра інструменту на моніторі комп'ютера.

Така перевірка називається **бекплот**(Backplot).

Бекплот дозволяє легко побачити помилку, яку важко помітити при звичайному перегляді коду керуючої програми. Приклад правильного та помилкового варіанту програми представлено в табл. 8.3.

Таблиця 8.3

Правильний та помилковий варіанти програми

Правильний варіант	Помилковий варіант
...	...
N80 G01 X3. Y2.	N80 G01 X3. Y2.
N90 G01 <b>X8.</b> Y2.	N90 G01 <b>X.8</b> Y2.
...	...
	

В другому варіанті в кадрі **N90** десяткова точка стоїть не на своєму місці, а програма обробки може мати достатньо великий об'єм і таку малу помилку розпізнати в коді дуже важко.

Бекплот мментально покаже, що траєкторія переміщення інструмента якісно не відповідає тому, що було задумано програмістом.

**Твердотільна верифікація** (verification - перевірка) на відміну від бекплота демонструє процес видалення матеріалу заготовки і дозволяє побачити повний результат роботи керуючої програми у вигляді готової деталі.

Отриману віртуальну деталь можна розглянути з різних сторін і зрозуміти, чи всі елементи виконані правильно, чи немає зарізів та зіткнень інструменту із заготовкою.



Бекплот використовують в процесі написання та відпрацювання програми, а твердотільну верифікацію на заключному етапі перевірки.

**Передача керуючої програми на верстат** після її створення та відпрацювання з комп'ютера на систему ЧПК верстата виконується з використанням спеціального комунікаційного забезпечення.

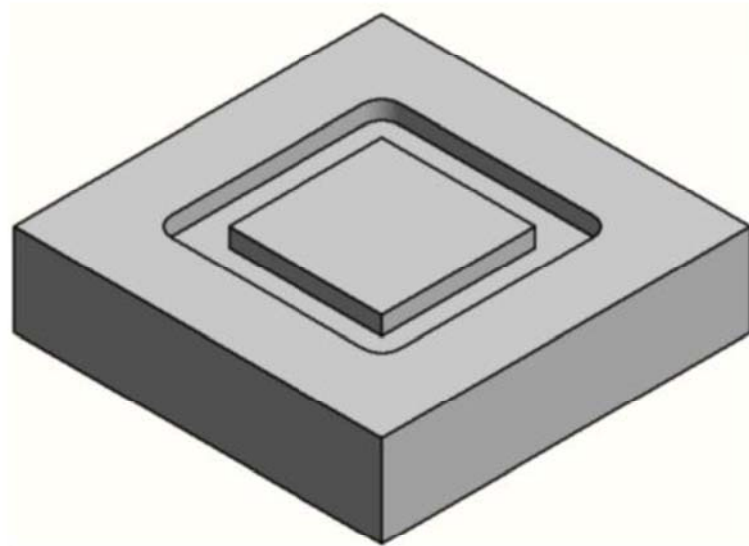


Рис.8.2. Твердотільна верифікація

**Для передачі даних необхідно, щоб система ЧПК верстата та комунікаційна програма були синхронізовані, що досягається настроюванням параметрів системи ЧПК та комунікаційної програми.**

Як правило, комунікаційна програма та кабель поставляються разом з верстатом, а інформацію про настроювання параметрів системи ЧПК для поєднання верстата з комп'ютером надає документація верстата.

**Для того, щоб виключити будь-яку помилку, перед виготовленням реальної деталі, прямо на верстаті виконують пробний прогін або тестову перевірку.**

Основною тестовою функцією є відпрацювання керуючої програми на холостих ходах (**Dryrun**).

При запуску керуючої програми в цьому режимі верстат блокує переміщення виконавчих органів по осі **Z**, по осях **X** та **Y**, або по усіх осях одночасно.

Наприклад, якщо верстат блокує вісь **Z**, то будуть виконуватись переміщення по осях **X** та **Y**, при цьому шпиндель буде обертатись. Це

дозволяє спокійно подивитись, як працює керуюча програма без побоювання «зарізатись» в матеріал заготовки.

Режим покадрового відпрацювання (**Singleblock**) призначений для виконання програми обробки по окремих кадрах.

Система числового програмного керування призупиняє роботу в кінці кожного кадру і чекає, доки оператор натисне кнопку «Старт циклу» для виконання наступного кадру.

Користуючись цим режимом можна легко визначити чи прийшов інструмент в правильну позицію.

Як правило, режим покадрового відпрацювання керуючої програми використовується разом з режимом обробки на холостих ходах чи при «піднятій» нульовій точці деталі.

Контролювати правильність переміщення в програмі обробки оператору допомагає також екранний режим **«Distance to GO -Відстань, що залишилась»**.

В цьому режимі оператор бачить, скільки міліметрів залишилось інструменту переміститись в поточному кадрі керуючої програми, що дозволяє йому візуально порівняти фактичну і задану відстані переміщення і уникнути серйозної помилки.

Для перевірки керуючої програми можуть використовуватись методи без тестових режимів верстата з ЧПК.

**Наприклад, після установки робочої системи координат – так званого «прив'язування» до деталі, можна підняти нульову точку на безпечну висоту над поверхнею заготовки.**

Ця висота повинна бути значно вищою ніж «найглибше» переміщення інструменту в даній керуючій програмі.

Виконання програми відрізняється лише тим, що інструмент «ріже» повітря а не реальну заготовку (рис. 8.3).

Найбільш реальним способом перевірки керуючої програми є обробка заготовки із пластика, наприклад, або спеціального різання не побоюючись зламати інструмент або пошкодити заготовку.

Для початкового програміста рекомендується виконувати повну перевірку керуючої програми.

Послідовність повної перевірки керуючої програми:

Виконати графічну перевірку коду програми на комп'ютері методами **бекплота та твердотільної верифікації**.

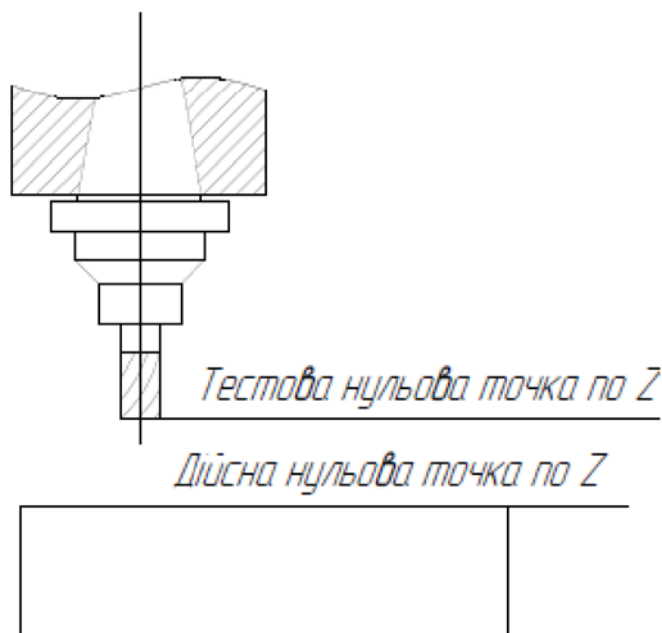


Рис.8.3. Підняття нульової точки по осі **Z** для відпрацювання керуючої програми в повітрі

Виконати наступні додаткові перевірки програми і налагодження верстата:

- усі прискорені переміщення виконуються над поверхнею заготовки на безпечній відстані;
- в керуючій програмі номери інструментів і номери коректорів повинні співпадати;
- інструмент, що встановлений в інструментальному магазині, повинен відповідати інструменту, що описаний в керуючій програмі;
- інструмент повинен бути надійно закріплений в патроні;
- в стійці ЧПК повинні знаходитись правильні величини компенсації довжини та радіусів інструментів;
- режими різання повинні відповідати умовам обробки;
- шпиндель повинен обертатись в потрібному напрямку;
- в керуючій програмі повинна бути команда на вмикання змащувально-охолоджувальної рідини (при необхідності);
- операції обробки повинні виконуватись в потрібній послідовності;
- чорнові операції повинні виконуватись перед чистовими;
- заготовка повинна бути надійно закріплена в пристосуванні;
- інструмент повинен переміщуватись від одного оброблюваного елемента до іншого на безпечній висоті по осі **Z**.

Виконати графічну перевірку програми на стійці ЧПК, якщо це можливо.

Відпрацювати керуючу програму на холостих ходах в наступній послідовності:

- включити режим покадрової обробки керуючої програми;
- зменшити величину робочої подачі;
- змістити нульову точку на безпечну відстань над поверхнею заготовки і прогнати програму «по повітрю».
- повернути нульову точку в нормальне положення, відмінити режим покадрової обробки керуючої програми і виконати обробку заготовки на понижених режимах;
- перевірити розміри виготовленої деталі і, при необхідності, внести поправки в програму.

### **Запитання до розділу 8**

1. Яка математична база використовується у верстатах з ЧПК для програмування механічної обробки?
2. Які точки траєкторії інструменту називають опорними?
3. Які особливості створення керуючої програми на ПК?
4. Які основні методи перевірки керуючої програми на персональному комп'ютері?
5. Які особливості використання методів блекпоту та твердотільної верифікації при перевірці керуючої програми на комп'ютері?
6. Які методи перевірки керуючої програми на верстаті?
7. Які особливості перевірки керуючої програми на холостих ходах?
8. В чому особливості методу покадрового відпрацювання керуючої програми на верстаті?
9. Які методи перевірки керуючої програми можна використовувати без тестових режимів верстата?
10. Яка послідовність повної перевірки керуючої програми?

## 9. Структура керуючої програми

**Керуюча програма є упорядкованим набором команд, за допомогою яких визначаються переміщення виконавчих органів верстата. Вона включає також і різноманітні допоміжні функції.**

Кожна програма обробки складається із деякої кількості рядків, які називаються **кадрами** керуючої програми.

**Кадр керуючої програми – це складова частина керуючої програми, що вводиться і відпрацьовується як єдине ціле і в своєму складі має не менше однієї команди.**

Система числового програмного керування зчитує і виконує програму кадр за кадром.

Дуже часто програміст призначає кожному кадру свій номер, який розташовується на початку кадру і позначається буквою **N**.

Більшість верстатів з ЧПК дозволяють спокійно працювати і без номерів кадрів, які використовуються виключно для зручності візуального сприйняття програми та для пошуку потрібної інформації.

Програмісту рекомендується розташовувати номери кадрів з інтервалом в 5 або 10 номерів, щоб при необхідності можна було вставити в програму додаткові кадри.

### 9.1. Підготовчі та допоміжні коди

Для написання керуючих програм використовують найчастіше **G** та **M-коди** (табл. 9.1).

Коди з адресою **G** називаються **підготовчими** і визначають настроювання системи числового програмного керування на виконання конкретної роботи.

Коди з адресою **M** називаються **допоміжними** і вони призначені для керування режимами роботи верстата.

Для керування численними функціями верстата з ЧПК використовується достатньо велике число різних кодів. Але вивчивши набір основних **G** та **M-кодів** можна без особливих труднощів створити керуючу програму.

Таблиця 9.1

Основні G та M-коди

Код (функція)	Призначення та приклад кадру з кодом
	<b>Підготовчі коди</b>
	<b>Коди осьового переміщення</b>
G00	<b>Прискорене переміщення</b> – переміщення на дуже великій швидкості в задану точку <b>G00 X10. Y20. Z25</b>
G01	<b>Лінійна інтерполяція</b> – переміщення по прямій лінії із вказаною величиною подачі <b>G01 X10. Y20. F100</b>
G02	<b>Кругова інтерполяція</b> – переміщення по дузі за годинниковою стрілкою із вказаною подачею <b>G02 X10. Y20. R10. F100</b>
G03	<b>Кругова інтерполяція</b> – переміщення по дузі проти годинникової стрілки із вказаною подачею <b>G03 X10. Y20. R10. F100</b>
	<b>Коди настроювання</b>
G20	<b>Введення дюймових даних</b> <b>G20 G00 X10. Y20.</b>
G21	<b>Введення метричних даних</b> <b>G21 G00 X10. Y20.</b>
G90	<b>Абсолютне позиціонування</b> – всі координати відраховуються від постійної нульової точки <b>G90 G00 X10. Y20.</b>
G91	<b>Відносне позиціонування</b> – всі координати відраховуються від попередньої позиції <b>G91 G00 X10. Y20.</b>
	<b>Постійні цикли для обробки отворів</b>
G81	<b>Цикл свердління</b> <b>G81 X10. Y20. Z-5. F30.</b>
G82	<b>Цикл свердління з затримкою на дні отвору</b> <b>G82 X10. Y20. Z-5. Q0.25 R1. F30.</b>
G85	<b>Цикл розточування отвору</b> <b>G85 X10. Y20. Z-5. F30.</b>
	<b>Допоміжні коди (функції)</b>
M00	<b>Запрограмована зупинка</b> – виконання програми тимчасово припиняється
M01	<b>Запрограмована зупинка за вибором</b> – виконання програми тимчасово припиняється, якщо активовано режим зупинки за вибором
M03	<b>Пряме обертання шпинделя</b> – шпиндель обертається за годинниковою стрілкою
M04	<b>Обернене обертання шпинделя</b> – шпиндель обертається проти годинникової стрілки
M05	<b>Зупинка шпинделя</b>
M06	<b>Автоматична зміна інструменту</b> <b>M06 T02</b>
M08	<b>Вмикання подачі охолоджувальної рідини</b>

Код (функція)	Призначення та приклад кадру з кодом
M09	<b>Вимикання подачі охолоджувальної рідини</b>
M30	<b>Кінець програми, переведення курсору в початок програми</b>

Для знайомства із структурою керуючої програми повернемося до програми обробки паза (рис.7.3).

%

O0001 (PAZ)

N10 G21 G40 G49 G54 G80 G90

N20 M06 T01 (FREZA D1)

N30 G43 H01

N40 M03 S1000

N50 G00 X3 Y9

N60 G00 Z0.5

N70 G01 Z-3 F25

N80 G01 X3 Y2

N90 G01 X8 Y2

N100 G01 X8Y9

N110G01 X3Y9

N120 G01 Z0.5

N130 M05

N140 M30

%

На самому початку програми обов'язково повинен знаходитись код початку програми % і номер програми (наприклад, **O0001**).

Два цих перші кадри не впливають на процес обробки, але вони необхідні, щоб система числового програмного керування могла розрізнити в пам'яті окремі програми.

**Призначати номери для таких кадрів не допускається.**

Кадр **N10** настроює систему числового програмного керування на необхідний режим роботи з наступними кадрами керуючої програми.

Наприклад, **G21** означає, що верстат буде працювати в метричній системі.

Інколи такі рядки називають **рядками безпеки**, оскільки вони дозволяють перейти системі в деякий стандартний режим роботи або відмінити непотрібні функції.

**N10 G21 G40 G49 G54 G80 G90**

Наступні кадри необхідні для підготовки верстата до обробки

Також при написанні програми можна використовувати символи коментарів. Система ЧПК ігнорує любий текст, що в круглих дужках. Тому можна вказувати в кадрі також діаметр і найменування фрези.

**N20 M06 T01 (FREZA D1)** – Доставка інструменту із магазину інструментів в шпindelь

**N30 G43 H01** – Активація компенсації довжини інструменту

**N40 M03 S1000** – Обертання шпинделя в потрібному напрямку із вказаною частотою.

Кадри з номерами від **№50** до **№110** безпосередньо відповідають за обробку деталі. В цій частині керуючої програми знаходяться кадри, що відповідають за переміщення інструменту в задані координати.

Наприклад, кадр **№80** переміщує інструмент в точку з координатами **X3, Y3** з подачею **25мм/хв.**

Заключні кадри призначені для зупинки шпинделя (**кадр №130**) та завершення програми (**кадр №140**).

**N130 M05**

**N140 M30**

%

Схематично кожну керуючу програму можна представити у наступному вигляді (рис. 9.1):

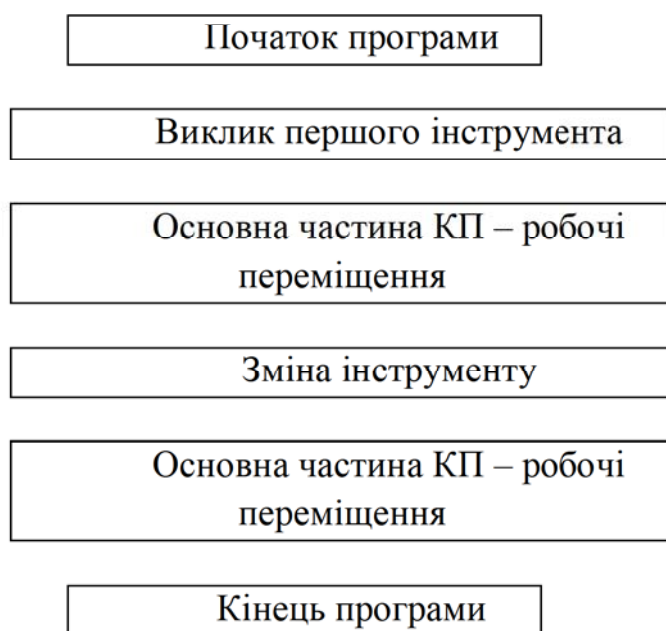


Рис. 9.1. Схематичний вигляд керуючої програми



## 9.2. Слово даних, адреса та число

Керуюча програма складається з великої множини різних **кадрів**. В свою чергу, кожний кадр керуючої програми складається із слів даних.

**Слово даних** складається із адреси (літери) та числа, що до нього відноситься.

### **G01 X3 Y2 - кадр керуючої програми**

G – адреса

01 - число

G01 – слово даних (G – код)

X - адреса

3 - число

X3 - слово даних (G – код)

Y - адреса

2- число

Y2 - слово даних (G – код)

Необов'язково, щоб число, яке відноситься до **G** або **M**-коду, включало **ведучі нулі** – це нулі перед числом, **G01, G02, G03....** Записи **G1, G2, G3** рівнозначні.

Все ж програмісти в силу звички віддають перевагу варіанту з ведучими нулями.

В залежності від здатності зберігатись в пам'яті системи ЧПК усі верстатні коди поділяють на два класи:

- **немодальні;**

- **модальні.**

**Немодальні** коди діють лише в тому кадрі, в якому вони знаходяться.

**Модальні** коди, навпаки, можуть діяти нескінченно довго, доки їх не відмінять іншим кодом.

Виділяють декілька груп модальних кодів, в залежності від функції, яку вони виконують.

Таблиця 9.2

Групи модальних кодів	
Функціональна група	Коди
Переміщення	G00, G01, G02, G03
Тип координатної системи	G90, G91
Одиниці введення даних	G20, G21
Постійні цикли	G80, G81, G82, G83, G84...

Функціональна група	Коди
Робоча система координат	G54, G55, G56
Компенсація довжини інструменту	G43, G44, G45
Корекція на радіус інструменту	G40, G41, G42
Повернення в постійних циклах	G98, G99
Активна площа обробки	G17, G18, G19

**Два модальні коди із однієї групи не можуть бути активними в один і той же час.**

Наприклад, **G02** та **G03** знаходяться в одній групі кодів осьових переміщень і їх неможливо використати одночасно.

Один з цих кадрів обов'язково відмінить дію іншого.

Однак можна використовувати одночасно коди з різних функціональних груп. Наприклад, можна в одному кадрі записати **G02** та **G90**.

Особливістю модальних кодів є те, що не потрібно вводити активний код в наступні кадри. Наприклад, код **G01** використовується для переміщення інструменту по прямій лінії. Якщо нам необхідно виконати кілька прямолінійних переміщень, то не обов'язково в кожному наступному кадрі писати **G01**.

Для відміни **G01** необхідно використати один із кодів тієї ж групи (**G00, G02** або **G03**).

Більшість із **G** кадрів є модальними. Програміст повинен знати, до якої групи і до якого класу належить той чи інший код.

**M**-коди не розділяють зазвичай на модальні та немодальні, хоча можна виділити групу **M**-кодів, що відповідають за подачу охолоджувальної рідини (**M07, M08, M09**) або за обертання шпинделя (**M03, M04, M05**). Все ж, більшість **M**-кодів потрібно розглядати як немодальні.

Деякі стійки ЧПК допускають програмування тільки одного **M**-коду в кадрі.

### 9.3. Рядок безпеки

**Рядком безпеки називається кадр, в якому знаходяться G коди, які переводять систему ЧПК в потрібний стандартний режим, відмінюють непотрібні функції та забезпечують безпечну роботу з керуючою програмою.**

В нашій програмі це кадр №10.

**N10 G21 G40 G49 G54 G80 G90**

Вище було вказано, що модальні коди залишаються в пам'яті системи ЧПК до того часу, доки їх не відмінюють. Можливі випадки, коли вже непотрібний модальний G код не був відмінений. Наприклад, якщо програма була перервана з якихось причин посередині.

Рядок безпеки, який знаходиться на початку керуючої програми, або після кадру зміни інструменту, дозволяє «відновити» забуті G коди і вийти в звичайний режим роботи.

Коди, що знаходяться в типовому рядку безпеки, наступні:

**G21** – метрична система або **G20** – дюймова;

**G40** – відмінює автоматичну корекцію на радіус інструменту, яка потрібна для автоматичного зміщення інструменту від запрограмованої траєкторії. Корекція може бути активною, якщо в кінці попередньої програми її забули відмінити (вимкнути), що може призвести до неправильної траєкторії переміщення інструменту;

**G49** – відмінює компенсацію довжини інструменту;

**G54** – дозволяє активізувати одну із декількох робочих систем координат. Цей код також модальний і зберігається активним в системі ЧПК доки його не вимкнуть. Щоб не виникла помилка в рядок безпеки включають код потрібної робочої системи координат (**G54 – G59**);

**G80** – відмінює усі постійні цикли (наприклад, цикл свердління) та їхні параметри;

**G90** – активізує роботу з абсолютними координатами. Якщо необхідно працювати у відносних координатах, то використовують код **G91**.

## 9.4. Формат програми

Одна і та ж керуюча програма може виглядати по різному (табл. 9.3).

Таблиця 9.3

Варіанти керуючої програми

...	...
N70 G01 Z-3 F25	N70 G01 Z-3 F25
N80 G01 X3 Y2	N80 X3 Y2
N90 G01 X8 Y2	N90 X8 Y2
N100 G01 X8 Y9	N100 X8 Y9
N110 G01 X3 Y9	N110 X3 Y9
N120 G01 Z0.5	N120 Z0.5
...	...

Оскільки **G01** є модальним кодом, то зовсім не обов'язково записувати цей код в кожному кадрі лінійних переміщень. В той же час адреси **X** та **Y** також є модальними, тобто зберігаються в пам'яті, доки система ЧПК не замінить їх іншими значеннями.

Таким чином даний фрагмент керуючої програми можна переписати ще раз:

```
...
N70 G01 Z-3 F25
N80 X3 Y2
N90 X8
N100 Y9
    N110 X3
N120 Z0.5
```

Система ЧПК читає програму обробки кадр за кадром. При цьому в буфер пам'яті системи попадає один або декілька кадрів. В сучасних системах ЧПК не принципово, в якому місці кадру знаходиться той чи інший код – слово даних. Лише в старих системах ЧПК порядок слів має значення. Для сучасної стійки мають однаковий ефект усі приведені три записи:

```
N01 G54 G01 X25.5 Y4.6 M08
N02 M08 Y4.6 G54 X25.5 G01
N03 G01 X35.5 Y4.6G54M08
```

Все ж для того, щоб програмісту було легше створювати і читати керуючу програму рекомендується наступний порядок розташування слів даних та знаків програмування в кадрі.

1. Код пропуску кадра - /
2. Номер кадру – N
3. Підготовчі функції – G-коди
4. Адреси осьових переміщень (X, Y, Z, I, J, K, A, B, C)
5. Команда подачі F
6. Команда частоти обертання S
7. Допоміжні функції (M-коди).

Після номеру кадру **N** найчастіше використовується

**G**-код. Це як дієслово в реченні. **G**-коди говорять, яку функцію несе кожний кадр. Потім адреси і позиції осьових переміщень. **M**-коди найчастіше ставлять в кінець кадру.

Якщо ж в кадрі відсутній **G**-код, то програмісти ставлять **M**-код на початок кадру.

**N40 M03 S1000**

В керуючій програмі не допускаються пробіли між адресою (літерою) та числом або всередині **G** та **M**-кодів.

**N50 G00 X3 Y9**

Більшість сучасних стійок прекрасно працюють і без пробілів між словами даних. Видалення пробілів дозволяє скоротити розмір керуючої програми.

Однак людині, на відміну від комп'ютера, незвично читати керуючу програму в такому вигляді

**1. N50 G00 X3 Y9**

**2. N50G00X3Y9**

Перший варіант читати легше, а це означає, що менше помилок при перевірці програми.

Номери кадрів для більшості сучасних систем ЧПК не обов'язкові. Вони використовуються для полегшення пошуку потрібної інформації в керуючій програмі та для створення переходів до потрібного кадру в деяких особливих випадках.

Тому фрагмент однієї і тієї ж програми може бути записаний по-різному (табл. 9.4).

Таблиця 9.4

Фрагменти однієї і тієї ж програми

...	...
N70 G01 Z-3 F25	G01 Z-3 F25
N80 X3 Y2	X3 Y2
N90 X8	X8
N100 Y9	Y9
N110 X3	X3
N120 Z0.5	Z0.5
...	...

Необхідно приділити також особливу увагу **числовому формату**, з яким може працювати стійка ЧПК.

Взагалі система ЧПК працює з десятковим форматом і дозволяє використовувати декілька знаків до десяткової крапки та декілька знаків після неї (наприклад, **999.9999**).

Можливі різні варіанти використання **ведучих** (перед десятковою крапкою) і **наступних** (після десяткової крапки) нулів:

**0.1**

**.1**

**0.100**

В деяких випадках наявність десяткової крапки в окремих словах даних обов'язкове. А в інших випадках недопустиме.

При роботі з додатними числами не потрібно вводити знак «+», оскільки система ЧПК сприймає всі числа як додатні, якщо не введений ніякий знак. Але якщо вводиться від'ємне число, знак «-» обов'язково повинен бути запрограмований.

Тепер можна порівняти програму обробки паза з новою, так званою економічною (табл. 9.5).

Не дивлячись на те, що другий варіант КП має менший розмір, а це економія програмної пам'яті системи ЧПК, його важче читати, що збільшує ймовірність виникнення помилок.

**Таким чином, форматування програми забезпечує читабельність програми завдяки чіткій структурі, коментарям, номерам кадрів та пробілам між словами даних.**

Форматування також забезпечує сумісність, що дозволяє кожному програмісту розібратись в програмі свого колеги, знайти помилку.

Таблиця 9.5

Звичайна та стиснута керуючі програми

Звичайна КП	Стиснута КП
%	%
O0001 (PAZ)	O0001(PAZ)
N10 G21 G40 G49 G54 G80 G90	G21G40G49G54G80G90
N20 M06 T01 (FREZA D1)	M6T1
N30 G43 H01	G43H1
N40 M03 S1000	M3 S1000
N50 G00 X3 Y9	G0X3Y9
N60 G00 Z0.5	Z.5
N70 G01 Z-1 F25	G1Z-3F25
N80 G01 X3 Y2	X3Y2
N90 G01 X8 Y2	X8
N100 G01 X8 Y9	Y9
N110 G01 X3 Y9	X3
N120 G01 Z0.5	Z.5
N130 M05	M5
N140 M30	M30
%	%

Ще важливішою причиною необхідності форматування КП є специфіка багатоінструментальної обробки на сучасних верстатах з ЧПК, що вимагає частої зміни інструментів та багаторазового використання одного і того ж інструменту.

У оператора може виникнути необхідність перезапуску програми з потрібного номера інструменту або операції.

Тому в керуючу програму включають набір додаткових команд, що дозволяють відпрацьовувати програму із потрібного кадру. Цими командами можуть бути не тільки команди вмикання заданих обертів шпинделя **S** та **M03**, але і рядки безпеки, команди на виконання компенсації довжини інструменту та корекції на його радіус.

Це означає, що одна керуюча програма може складатись з багатьох «міні програм».

```
%  
O0002  
(PROGRAMNAME – OTVIR)  
N10 G21  
N20 G00 G17 G40 G49 G80 G90  
(1 OPERATION- SVERDLINNJA)  
N30 T1 M6  
N40 G00 G90 G54 X52.0 Y34.0 M3 S120  
N50 G43 H1 Z120  
...  
(2 OPERATION- ZENKERUVANNJA)  
N100 T2 M6  
N110 G00 G90 G54 X52.0 Y34.0 M3 S240  
N120 G43 H2 Z120  
...  
(3 OPERATION - ROZVERTANNJA CHORNOVE)  
N200 T3 M6  
N210 G00 G90 G54 X52.0 Y34.0 M3 90  
N220 G43 H3 Z120  
...  
(4 OPERATION - ROZVERTANNJA CHISTOVE)  
N300 T4 M6  
N310 G00 G90 G54 X52.0 Y34.0 M3 70  
N320 G43 H4 Z120  
...  
N400 M08  
N410 G28 X0 Y0 Z0  
N420 M30  
%
```

При роботі з керуючими програмами практично неможливо запам'ятати технологічні особливості та нюанси роботи з тією чи іншою програмою, а оператор повинен мати відповідну інформацію для настроювання верстата. Наприклад, де знаходиться нульова точка програми, які різальні інструменти використовуються.

На сьогоднішній день ця інформація зберігається двома способами: за допомогою коментарів в програмі та карти наладки.

**Коментарі** – це звичайні речення, за допомогою яких програміст передає оператору верстата технологічну інформацію.



Як правило, в коментарях знаходяться наступні дані:

- Дата та час створення керуючої програми.
- Номер креслення.
- Матеріал заготовки.
- Дані про робочу систему координат.
- Розміри інструменту.
- Назви технологічних операцій.

Для того, щоб ввести коментарі в керуючу програму необхідно використовувати спеціальні символи – знаки програмування.

Для більшості верстатів з ЧПК використовують круглі дужки або крапку з комою.

Перед символами коментарів не ставлять номери кадрів, якщо коментарі займають кадр повністю.

...

(TOOLT3)

Більшість систем числового програмного керування працюють з латинськими літерами, тому коментарі також треба писати латинськими літерами, інакше вони будуть нечитабельні.

**Карта наладки** – документ, в якому є вся необхідна технологічна інформація для налагоджування верстата, включаючи ескіз деталі та пристосування, опис різального інструменту, положення нульової точки програми, режимів різання та часу обробки.

Така карта наладки дозволяє швидко відновити в пам'яті технологічні особливості виготовлення деталі навіть через тривалий термін.

Коментарі в керуючій програмі та карта наладки не замінюють один другого, а використовуються разом.

Карту наладки можна створити вручну, схематично зобразивши деталь, нульову точку, схему базування та записавши порядок операцій і дані інструментів.

Системи CAD/CAM вміють самостійно генерувати карту наладки та інструментальну карту, в якій зберігаються детальні відомості про різальний інструмент, що використовується в даній керуючій програмі.

## **Запитання до розділу 9**

1. Що таке кадр керуючої програми?
2. Які коди найчастіше використовують для написання керуючої програми?
3. В чому різниця підготовчих G-кодів та допоміжних M-кодів?
4. Для чого потрібний рядок безпеки в керуючій програмі?
5. Як схематично можна представити керуючу програму?
6. Що розуміють під словом даних, адресою та числом?
7. Яка різниця між модалними та немодальними кодами?
8. Який порядок розташування слів рекомендують використовувати в кадрі керуючої програми?
9. Для чого використовують номери кадрів і чи вони обов'язкові в програмі?
10. Для чого необхідне форматування програми?
11. Які дані знаходяться в коментарях програми?
12. Як вводяться коментарі в керуючу програму?
13. Для чого необхідна карта наладки та яку технологічну інформацію вона містить?

## 10. Особливості використання базових кодів та постійних циклів

### 10.1. G-коди

Сучасні верстати з ЧПК розуміють більше сотні різних команд, але в повсякденній роботі технолог-програміст використовує все ж достатньо обмежений набір **G** та **M**-кодів.

Справа в тому, що керуюча програма вирішує основну задачу переміщення інструменту по заданих координатах.

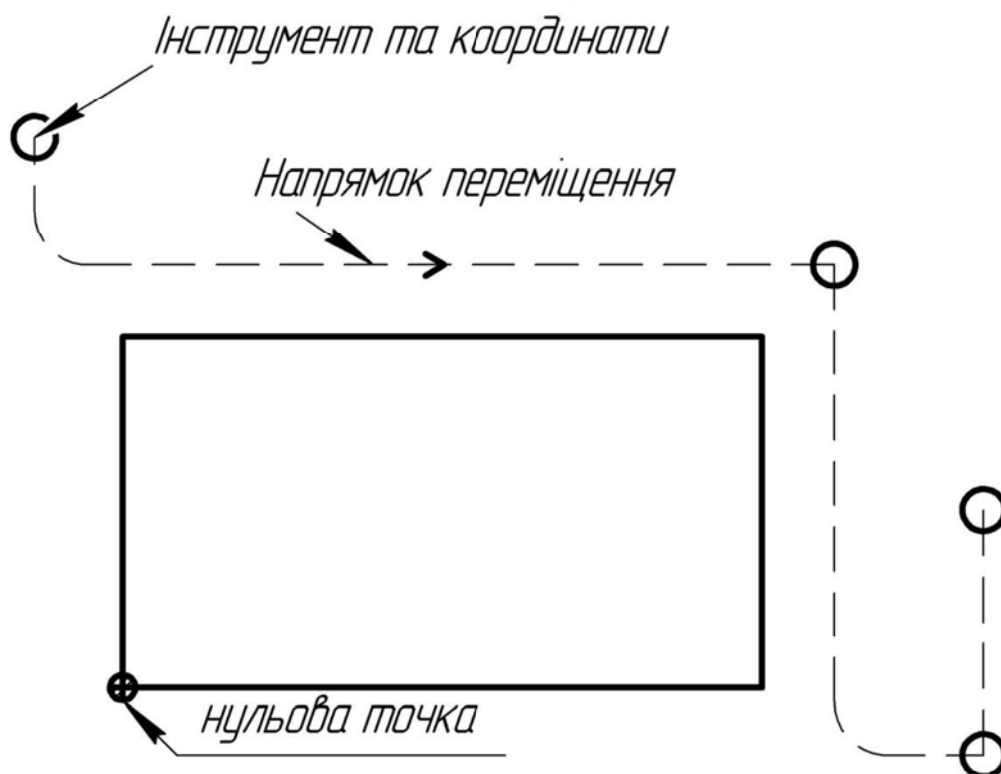


Рис.10.1. Схема переміщення інструменту відносно заготовки

Для реалізації таких переміщень необхідно використовувати лише декілька кодів, які прийнято називати базовим.

#### **G00 – прискорене переміщення**

Прискорене переміщення необхідне для швидкого транспортування інструменту в позицію обробки або безпечну позицію.

**Прискорене переміщення ніколи не використовується для виконання обробки, оскільки швидкість руху інструменту занадто висока і непостійна.**

Використання коду **G00** дозволяє значно знизити час обробки.

Для виконання прискореного переміщення достатньо вказати в кадрі код **G00** та координати потрібної позиції. Умовно кадр прискореного переміщення виглядає так:

```
...  
N10G00 X10.0 Y20.0  
N20 X40.0  
N30Y0.0  
...
```

Кадр **N10** забезпечує прискорене переміщення інструменту в точку з координатами **(10;20)**. Наступний кадр **N20** виконає позиціонування в точку **(40;20)**. Кадр **N30** переміщує інструмент в точку **(40;0)**. Оскільки код **G00** є модальним кодом, то немає ніякої необхідності вказувати його ще раз в кадрах **N20** та **N30**.

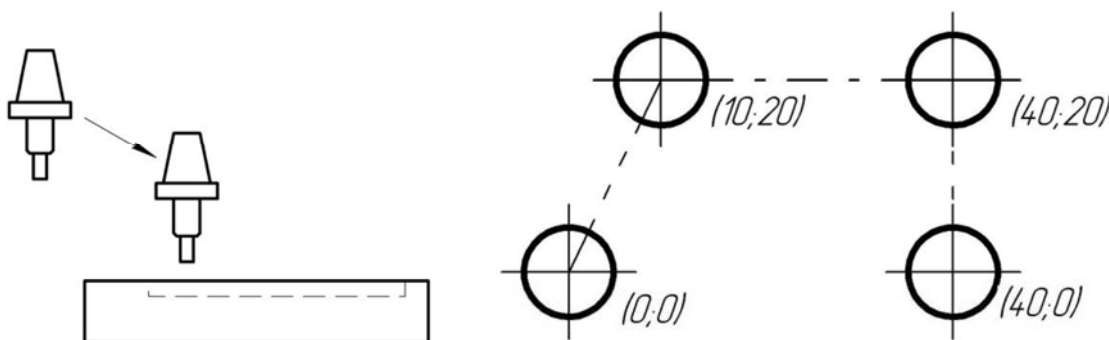


Рис.10.2. Прискорене переміщення інструменту

При використанні коду **G00** завжди потрібно залишати невелику відстань, в межах **0.5...5мм**, між поверхнею заготовки та інструментом.

При прискореному **підведенні** інструменту по трьох осях спочатку краще виконати позиціонування по осях **X Y**, а вже потім по осі **Z**. При **відведенні** інструменту – навпаки.

Тому рекомендується розділяти прискорене переміщення інструменту на два кадри – на позиціонування по осях **XY** в одному кадрі та по осі **Z** в іншому. Справа в тому, що при позиціонуванні з кодом **G00** по трьох осях одночасно, траєкторія руху інструменту може не бути прямою лінією. Система ЧПК віддає команду двигунам на переміщення колони або столу верстата на максимальній швидкості у вказану координату. Як тільки досягається потрібна координата по одній із осей, то прискорене переміщення по цій осі зупиняється, хоча по інших осях може ще продовжуватись через різні величини переміщень. Внаслідок цього

траєкторія може виглядати як ламана лінія, результатом чого може стати неочікуване зіткнення інструменту з кріпильними елементами та пристосуванням.

При порівнянні двох фрагментів однієї і тієї ж керуючої програми

...

```
N10 G00 X50.0 Y40.0 Z0.5    N10 G00 X50.0 Y40.0  
                            N20 Z0.5
```

бачимо, що в другому фрагменті прискорене переміщення розбито на два кадри: спочатку інструмент швидко переміщується в точку **(50, 40)**, а вже потім опускається по осі **Z0.5**.

Цей варіант більш безпечний, оскільки основне позиціонування виконується по осях **X** та **Y**, далеко від заготовки та пристосування, тоді як в першому випадку є ймовірність зіткнення інструменту з деталлю.

**При роботі з прискореними переміщеннями необхідно проявляти підвищену увагу, оскільки швидкість може досягати 30 і більше метрів за хвилину.**

Тому програмісти ніколи не допускають прискорені переміщення нижче площини заготовки і завжди перевіряють кадри, в яких є код **G00**.

#### **G01 - лінійна інтерполяція**

Цей код використовується для виконання лінійної інтерполяції, тобто переміщення інструменту по прямій лінії.

...

```
N10 G01 X10.0 Y30.0 F90  
N20 X20.0 Y50.0 Z30.0
```

...

В першому кадрі з'явилося слово **F90** – це величина подачі, з якою може виконуватись обробка. При цьому система ЧПК підтримує прямолінійне переміщення навіть по трьох осях одночасно.

В кадрі **N10** інструмент переміщується в точку **(10;30)** зі швидкістю **90мм/хв**. Наступний кадр виконує лінійне переміщення в точку **(20;50;30)**. Оскільки кадр **G01** модальний, то його непотрібно вказувати в кадрі **N20**. Те ж саме відноситься і до подачі **F90**, яка буде незмінною і в інших кадрах, доки не буде запрограмовано нове значення.

**Лінійна інтерполяція використовується не тільки для обробки в площині X – Y, але і для вертикального врізання в матеріал заготовки.**

#### **G02 та G03 - кругова інтерполяція**

Код **G02** використовується для переміщення за годинниковою стрілкою а **G03** – проти годинникової стрілки.

Напрямок визначається, якщо на інструмент дивитись з боку шпинделя – у від'ємному напрямку осі  $Z$  (рис.10.3).

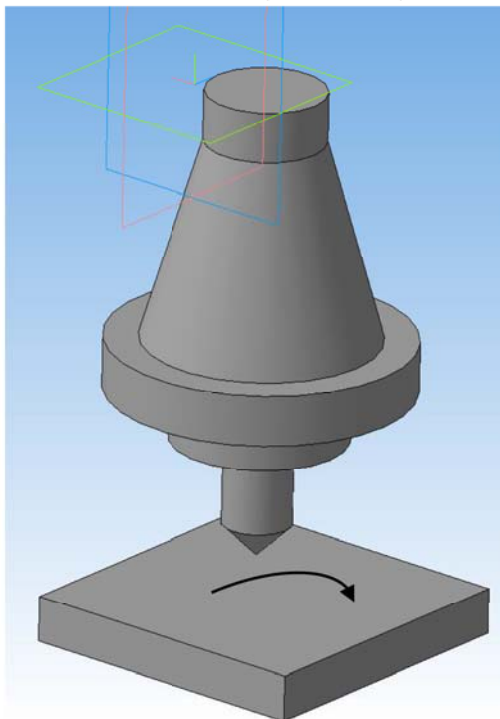


Рис.10.3. Напрямок переміщення по дузі

Як і при виконанні лінійної інтерполяції, в кадрі необхідно вказати величину робочої подачі  $F$ .

Можливі два способи для форматування кадру кругової інтерполяції.

В першому варіанті вказують: координати кінцевої точки дуги  $X, Y, Z$ ; відносні (інкрементальні) відстані від початкової точки дуги до її центру  $I, J, K$  та величину подачі  $F$ , а в другому – замість  $I, J, K$  вказують величину радіуса  $R$ .

Більшість сучасних верстатів з ЧПК підтримують обидва варіанти запису.

Якщо в кадрі крім  $X, Y$  знаходиться  $Z$  слово даних, то це означає, що виконується **гвинтова інтерполяція**, яка дозволяє виконувати фрезерування різьби та забезпечує плавне гвинтове врізання інструменту в матеріал заготовки.

### Дуга з $I, J, K$

В цьому випадку (рис. 10.4) вказують відносні (інкрементальні) відстані від початкової точки дуги до її центру  $I, J$  та  $K$ . Слово даних з  $I$  відноситься до осі  $X$ , слово даних з  $J$  відноситься до осі  $Y$ , а слово даних з

**K** відноситься до осі **Z**. При цьому в залежності від розташування дуги, значення можуть бути додатними або від'ємними.

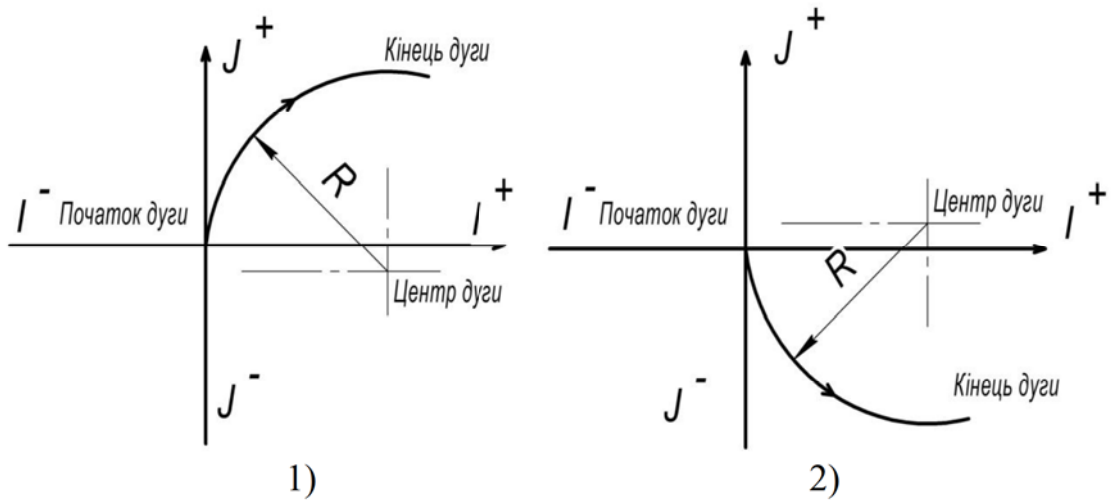


Рис.10.4. Опис дуги

Для опису дуги №1 необхідно вказати позитивне значення для **I** та від'ємне для **J**, а для опису дуги №2 необхідно вказати позитивне значення як для **I** так і для **J**.)

Приклад:

**N10 G02 X3.0 Y3.0 I3.0 J0.0 F90**

Приведений фрагмент керуючої програми переміщує інструмент по дузі з радіусом **3мм** з точки **A(0;0)** в точку **B(3;3)** з робочою подачею **90мм/хв** (рис. 10.5).

Оскільки центр дуги знаходиться на відстані **3мм** по осі **X** та **0мм** по осі **Y** відносно початкової точки **A**, то координата **I** буде **3.0** а координата **J** - **0**.

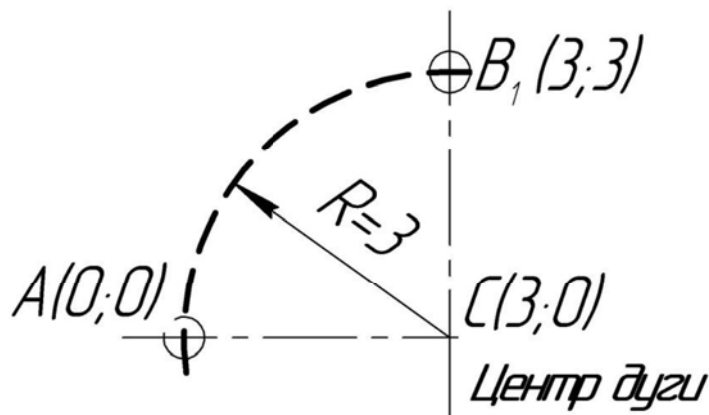


Рис.10.5. Переміщення інструменту по дузі на чверть кола з точки **A** в точку **B**

Отримана дуга складає лише чверть від повного кола.

Попробуємо виконати все коло.

Щоб перемістити інструмент з точки **B**(**B1**) в точку **B2**. в наступному кадрі немає необхідності повторно вказувати слово даних **F90**, оскільки подача не змінюється

**N20 G02 X6.0 Y0.0 I0.0 J-3.0**

Центр дуги знаходиться на відстані **0мм** по осі **X** та **3мм** по осі **Y** відносно точки **B**, тому **I** буде дорівнювати **0**, а **J** - **3**.

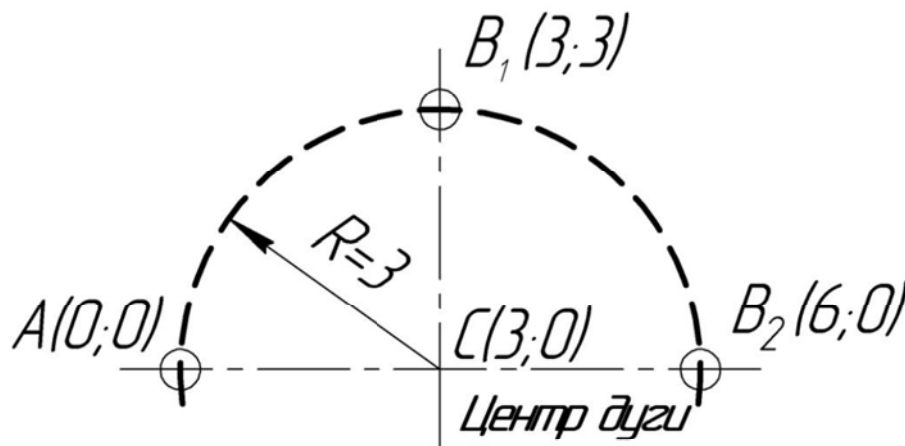


Рис.10.6. Переміщення інструменту по дузі на чверть кола з точки **B1** в точку **B2**

Таким чином за допомогою двох кадрів вдалось створити переміщення по дузі з точки **A** в точку **B2**.

Приклад не випадковий. Справа в тому, що багато верстатів вимагають такої розбивки кола. Тобто для описування повного кола необхідно чотири кадри.

На сьогоднішній день більшість систем **ЧПК** дозволяють виконувати операцію по описуванню повного кола за два або за один кадр.

Тому для переміщення з точки **A** в точку **B2** можна використати наступний запис

**N05 G02 X6.0 Y0.0 I3.0 J0.0**

Для повного кола з радіусом **3мм** та центром в точці з координатами **(0;0)** справедливий такий кадр (рис. 10.7)

**N15 G02 X-3.0 Y0.0 I3.0 J0.0**



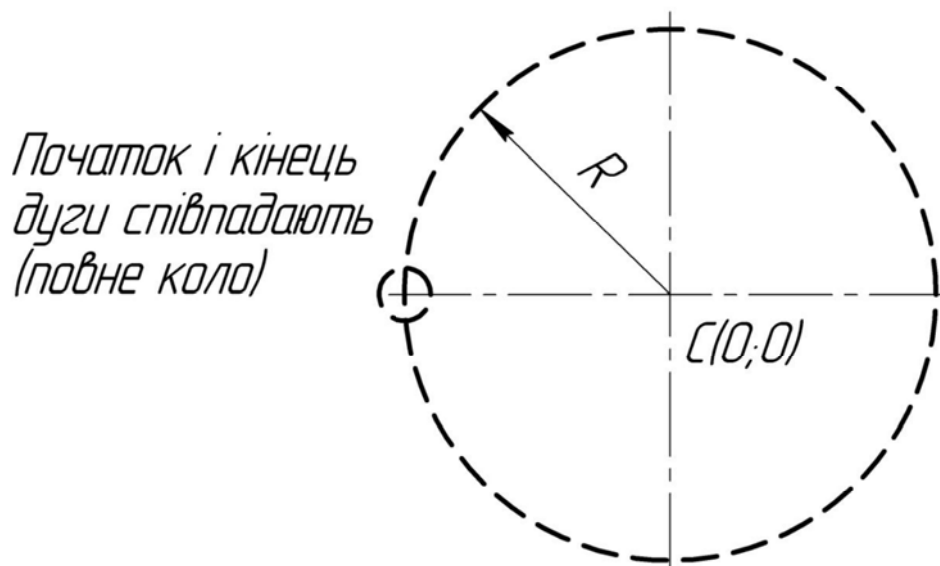


Рис.10.7. Побудова повного кола

Якщо початкова і кінцева точки створюють складний кут або знаходяться в різних квадрантах (рис. 10.8), то необхідні відповідні тригонометричні розрахунки для визначення **I, J, K**. Розрахунки повинні бути достатньо точними бо система ЧПК може видати повідомлення про неможливість побудови дуги.

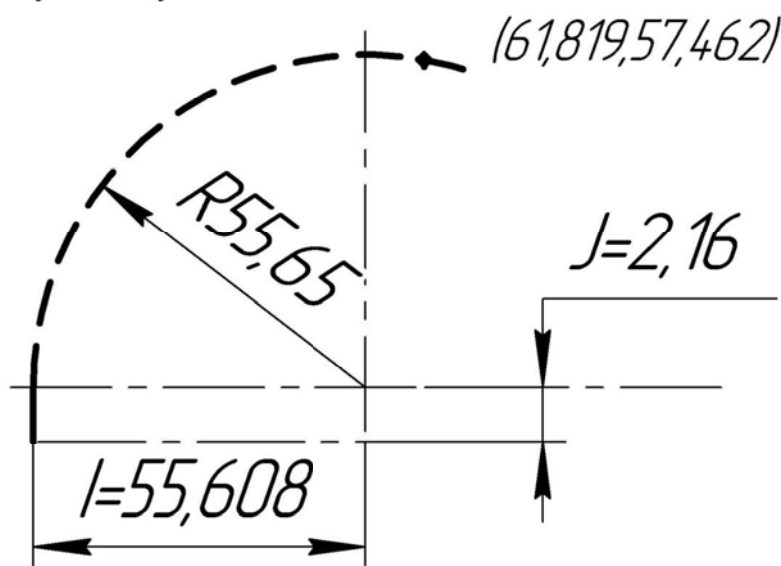


Рис.10.8. Побудова кривої, у якої початкова і кінцева точки створюють складний кут ( $>90^\circ$ )

### Дуга з R

При використанні адреси **R**(радіуса) спрощується визначення центру дуги, оскільки система ЧПК самостійно розраховує необхідні координати для визначення центру дуги.

Для однозначного визначення форми дуги необхідно вказувати відповідний знак перед числовим значенням радіуса  $R$ .

Для дуги більшої за  $180^\circ$  значення  $R$  буде від'ємним, а для дуги меншої від  $180^\circ$  значення  $R$  буде позитивним.

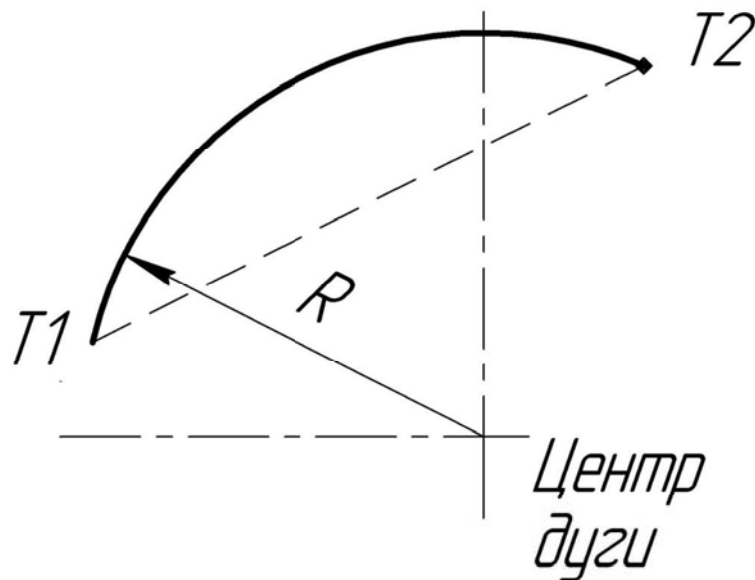


Рис.10.9. Дуга менша від  $180^\circ$  (центр розташований зовні хорди) -  $R$  позитивний

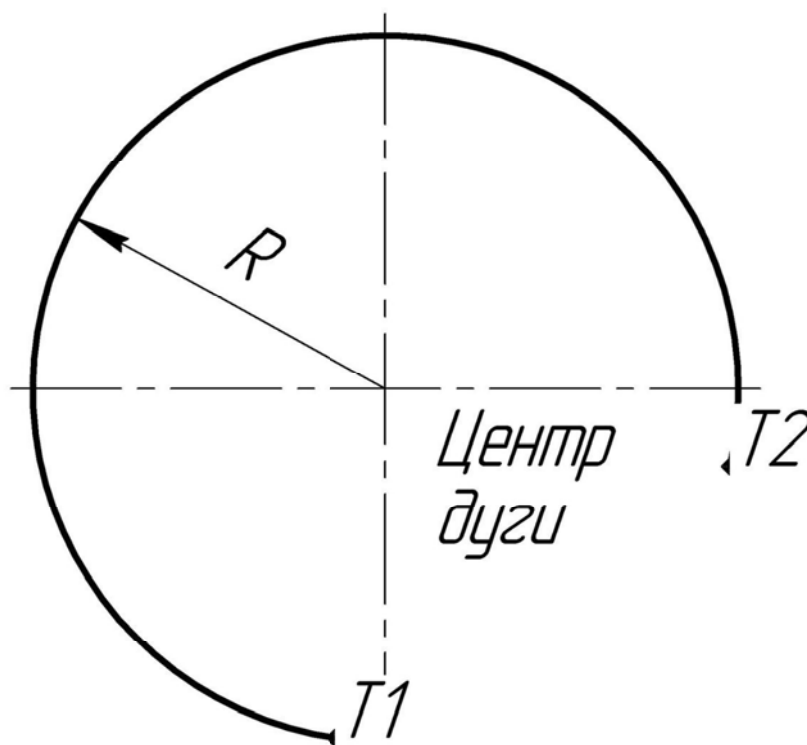


Рис.10.10. Дуга більша за  $180^\circ$  (центр розташований всередині хорди) -  $R$  від'ємний

Приклад:

Необхідно описати дугу за допомогою кадрів кругової інтерполяції з **R** словом даних.

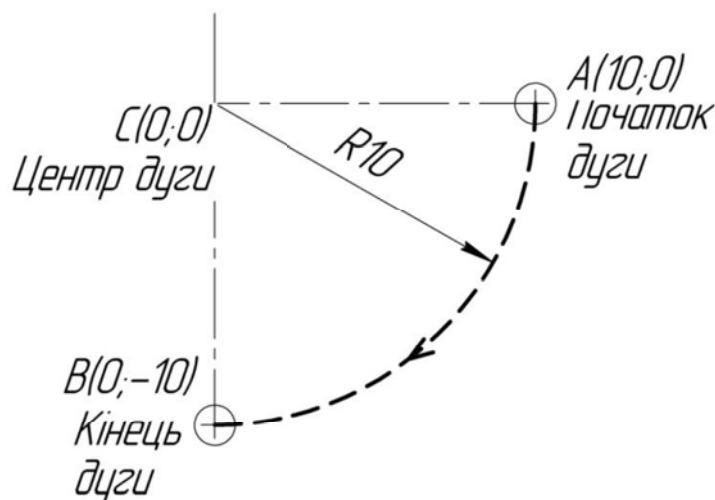


Рис.10.11. Виконання дуги за годинниковою та проти годинникової стрілки з допомогою R

Якщо інструмент переміщується по дузі за годинниковою стрілкою з точки **A** в точку **B**, то в керуючій програмі повинен бути кадр **G02 X0.0 Y-10.0 R10**

Якщо ж інструмент переміщується по дузі проти годинникової стрілки з точки **B** точку **A**, то в керуючій програмі повинен бути кадр **G03 X10.0 Y0.0 R10**.

## 10.2. Допоміжні коди – M-коди

Для керування режимами верстата використовують **M(Miscellaneous)** коди, які називаються **допоміжними**

**M-код** може стояти як окремо, так і знаходитись в кадрі з **G-кодами**.

Деякі **M-коди** працюють разом з іншими адресами. Наприклад, **M-код**, що відповідає за напрямок обертання шпинделя вказується з адресою **S**, яка необхідна для встановлення частоти обертання **N10 M03 S1000**.

Якщо **M-код** знаходиться в адресі з **G-кодом**, то порядок виконання залежить від моделі ЧПК.

Наприклад:

**N10 G01 X110.0 Y130.0 Z50.0 F80.0 M08**

Цей кадр виконує лінійне переміщення і включає подачу охолоджувальної рідини.

Одні верстати включають подачу ЗОР зразу ж , інші – лише після переміщення у вказану позицію. Тому для надійності стараються вказувати код **M08** перед виконанням переміщень.

**N05 M08**

**N10 G01 X110.0 Y130.0 Z50.0 F80.0**

**M00 та M01 – зупинка виконання керуючої програми**

Коли система ЧПК читає код **M00**, то всі осьові переміщення вимикаються, а шпиндель ще продовжує обертатись та і інші функції залишаються активними.

Така тимчасова зупинка програми необхідна, наприклад, для видалення стружки, перевірки розмірів деталі або перестановки кріпильних елементів.

Продовження програми з кадру, що є наступним після **M00**, можливе лише після того, як оператор натисне кнопку «Старт циклу».

...

N110 G01 X110

N120 G00 Z50

N130 **M00**

N140 G00 Z0.5

N150 G01 Z-3F80

...

Все ж при необхідності видалення стружки із зони різання або із свердла для гарантування безпеки необхідно зупинити і шпиндель. Для цього перед командою **M00** вказують **M05** - вимикання шпинделя. Але потім треба обов'язково знову ввімкнути шпиндель щоб інструмент переміщувався з обертанням, що виключає його поломку.

...

N200 G01 X110

N210 G00 Z50

N220 **M05**

N230 **M00**

N240 **M03 S1000**

N250 G00 Z0.5

N260 G01 Z-3 F80

...

### Код **M01** призначений для зупинки за вибором.

Він діє аналогічно коду **M00**, але дає можливість оператору вибирати самому чи переривати чи не переривати виконання керуючої програми. Для цього на панелі системи ЧПК є клавіша «**M01**». Якщо клавіша натиснута, то при зчитуванні кадру **M01** виконується зупинка. Якщо ж клавіша не натиснута, то команда **M01** пропускається і виконання керуючої програми не переривається.

...

N200 G01 X110

N210 G00 Z50

N220 **M01**

N230 G00 Z0.5

N240 G01 Z-3 F80

...

### **M03, M04, M05** – керування обертанням шпинделя

Код **M03** відповідає за **пряме** (за годинниковою стрілкою) обертання шпинделя, а **M04** – за **обернене** (проти годинникової стрілки).

Напрямок обертання визначається якщо дивитись з боку шпинделя в сторону заготовки, тобто у від'ємному напрямку осі **Z**.

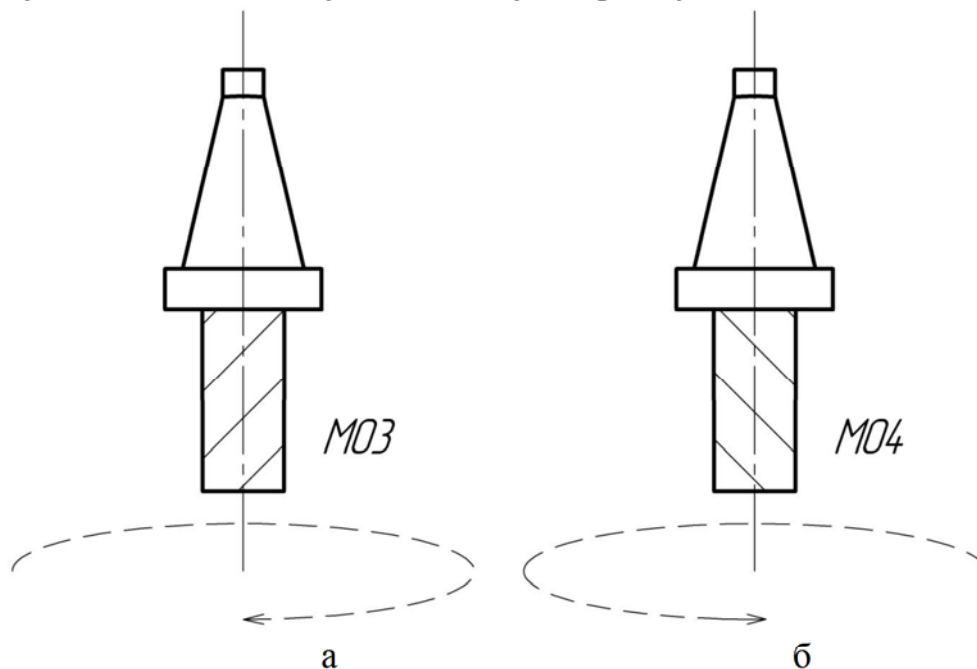


Рис.10.12. Пряме (а) та обернене (б) обертання шпинделя

При фрезеруванні інструменти повинні мати пряме обертання **M03**. При виведенні мітчика з отвору, при нарізуванні лівої різьби, в

циклах автоматичного вимірювання діаметру інструмента може використовуватись обернене обертання шпинделя (**M04**).

В кінці програми обробки та перед заміною інструменту необхідно зупинити обертання шпинделя за допомогою команди **M05**.

...

N40 Z5.0

N45 G00 Z50.0

N50 **M05**

N55 M30

%

Для встановлення частоти обертання шпинделя використовують Садресу. За буквою **S** вказується числове значення частоти обертання шпинделя, найчастіше в цілих числових значеннях обертів за хвилину, що вимагає більшість систем ЧПК, які сприймають тільки цілі значення **S**. Коди **M03** та **S** зазвичай знаходяться в одному кадрі.

Наприклад, в приведеній програмі кадр **N20**заставляє обертатись шпиндель в прямому напрямку з частотою **1000** обертів за хвилину.

%

O0002

N10G21 G40 G49 G54 G80 G90 G98 G00

N20 T1 M06

N30 G43 H1 Z50.0

N40 **M03 S1000**

N50G00 X110.0 Y130.0 Z0.5

N60 G01 Z-3

N70 X180.0 Y190.0

N80 Z0.5

N90 G00 Z50.0

N100 M05

N110 M30

%

Сучасні верстати з ЧПК оснащуються системою автоматичної подачі ЗОР і для керування цією системою використовуються коди **M07**, **M08** та **M09**.

**Код M08** використовується для вмикання подачі охолоджувальної рідини, а код **M09** – для вимикання.

В деяких верстатах код **M08** подає в зону обробки охолоджувальну рідину у вигляді струменю, а код **M07** – в розпиленому вигляді.

Подачу ЗОР обов'язково вимикають перед зміною інструменту та в кінці програми обробки. Сучасні верстати в основному виконують це автоматично при читанні коду **M06**(зміна інструменту), кодів **M30** та **M02**(кінець програми).

Окрім програмного керування має місце і ручне керування, що дозволяє оператору за допомогою клавіш на панелі пристрою ЧПК вмикати або вимикати подачу ЗОР у випадку необхідності.

#### **M06 – автоматична заміна інструменту**

Більшість сучасних верстатів з ЧПК мають інструментальні магазини. Кожне гніздо магазину має власний номер. Спеціальні датчики та пристрій зворотного зв'язку допомагають системі ЧПК визначити положення магазину інструментів та наявність інструменту в гнізді.

Для виконання автоматичної заміни інструменту програміст напряму вказує номер інструменту, який потрібно взяти. В більшості випадків номери інструменту і гнізда співпадають. Такий спосіб заміни інструменту називається **абсолютним** на відміну від **відносного**, коли номер інструменту відраховується від номеру поточного, що не так зручно.

Найбільш поширені дві основні конфігурації пристроїв автоматичної заміни інструментів:

- магазин інструментів переміщується при заміні інструменту – тип «парасолька» (рис. 10.13);
- магазин інструментів не переміщується при заміні інструменту – тип «рука».



Рис. 10.13. Заміна інструменту методом «парасолька»

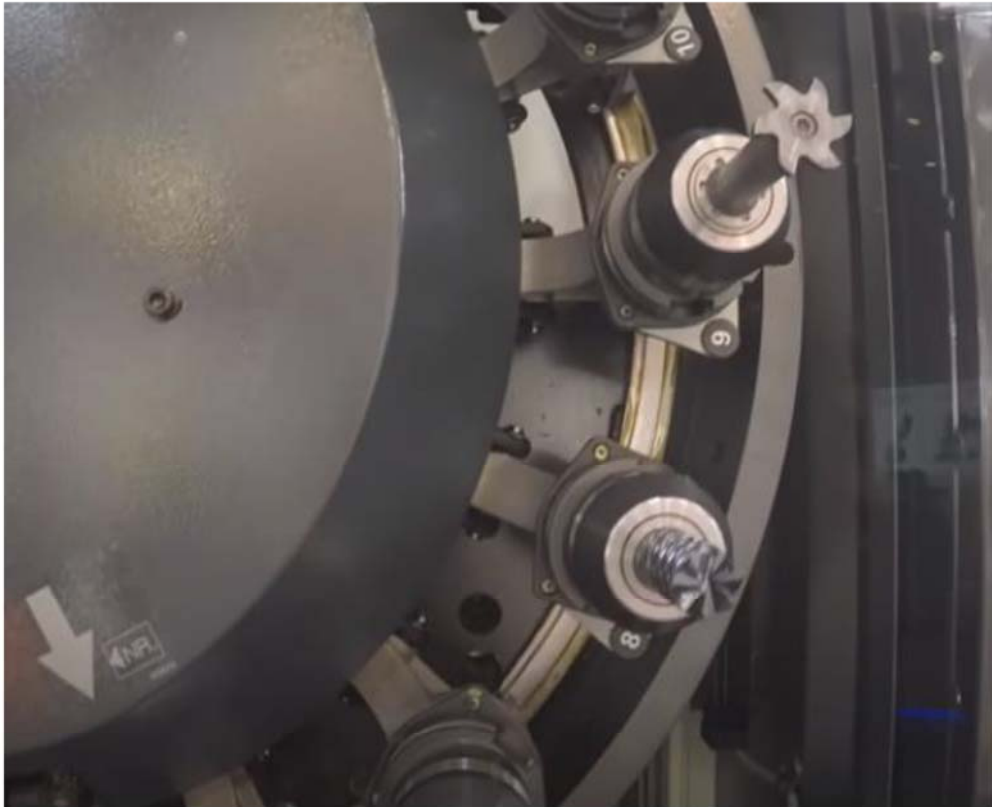


Рис. 10.14. Інструментальний магазин



Рис. 10.15. Заміна інструменту маніпулятором



В першому випадку коли керуюча програма доходить до кадру зміни інструменту, шпиндель переміщується у визначену точку, що знаходиться рядом з магазином інструментів. Магазин переміщується в ту ж точку до «зчеплення» інструменту з пустим гніздом. Шпиндель піднімається на якусь величину вверх, звільняючи відпрацьований інструмент. Магазин повертається таким чином, щоб вибраний інструмент розташувався під шпинделем. Шпиндель опускається, затискає новий інструмент і відводиться вверх. Магазин інструментів відводиться на своє місце.

В другому випадку (рис. 10.14, 10.15) )спочатку шпиндель переміщується у визначену точку коло магазину інструментів. Потім магазин інструментів повертається таким чином, щоб вибраний інструмент знаходився навпроти шпинделя. Механічний захват - «рука» захватує відпрацьований інструмент в шпинделі та новий в магазині. Захват опускається вниз, звільняє інструменти і, повернувшись навколо своєї осі на  $180^{\circ}$ , міняє їх місцями,потім знову піднімається вверх. При цьому новий інструмент затискається в шпинделі, а відпрацьований поміщається в магазин.

Крім коду **M06** в команді на заміну інструменту вказують також і адресу **T** з номером інструменту, що викликається. **M06 T05**

Більшість систем ЧПК допускають любий порядок слів даних в кадрі заміни інструменту. Тобто інструмент можна викликати і таким чином:

**T05 M06**

Деякі верстати вимагають щоб адреса **T** та команда **M06** знаходились в різних кадрах, інакше автоматична заміна інструменту може бути виконана неправильно.

**N10 T1**

**N20 M06**

**Одразу після заміни інструменту необхідно виконати компенсацію довжини нового інструменту.**

Компенсація довжини нового інструменту виконується за допомогою коду **G43** та наступного за ним **H** слова даних. Для зручності номер коректора на довжину співпадає з номером інструменту. Таким чином для виконання компенсації довжини нового інструменту **№5** в керуючій програмі необхідно вказати: **G43 H5**

Деякі верстати старих моделей вимагали вказувати **напрямок компенсації** довжини інструменту. При цьому код **G43** означав позитивний напрям, а код **G44** – від'ємний.

Приклад програми з кадрами заміни інструменту та активації виконання компенсації довжини нового інструменту в керуючій програмі:

```
%  
O0002  
N05 G21 G40 G49 G54 G80 G90 G98 G00  
N10 T1 M06  
N15 G43 H1 Z50.0  
N20 M03 S1000  
N25 G00 X110.0 Y130.0  
N30 Z2.0  
N35 G01 Z-3  
N40 X180.0 Y190.0  
N45 Z2.0  
N50 G00 Z50.0  
N55 M05  
M60 M30  
%
```

В кадрі **N10** виконується заміна інструменту (виклик інструменту №1), а в кадрі **N15** виконується компенсація довжини інструменту №1 та інструмент переміщується в точку **Z50.0**.

Перед викликом нового інструменту прийнято відмінити компенсацію довжини активного інструменту. Ця дія проводиться за допомогою коду **G49**, хоча багато сучасних систем ЧПК відмінюють компенсацію довжини активного інструменту автоматично при вказуванні команди **M06**.

**Якщо заміна інструменту виконана, а компенсація його довжини не виконана, то можливе зіткнення інструменту з заготовкою або частинами верстата.**

Тому для забезпечення безпеки перед зміною інструменту виконують повернення у вихідну позицію по осі **Z**.

```
...  
G91 G28 Z0  
T3 M06  
G43 H3
```

### **M30 та M02 – завершення програми**

В кінці кожної керуючої програми повинен знаходитись код її завершення – **M30** або **M02**. При виконанні любого з цих кодів верстат зупиняється незалежно від того, яку функцію він виконував.

Різниця між ними лише в тому, що код **M30** крім завершення програми переводить курсор поточного положення в початок програми, а код **M02** цього не робить.

```
...  
N45Z2.0  
N50G00 Z50.0  
N55M05  
M60M30  
%
```

Після завершення програми обробки деталі виконується переміщення робочого столу або інструменту в позицію, яка полегшує оператору зняття готової деталі з верстату, за допомогою **коду повернення у вихідну позицію G28**.

```
N120 G91G28 X0 Y0 Z0  
N130 M05  
N140 M30  
%
```

### **Запитання до розділу 10**

1. Які особливості використання коду G00 – прискорене переміщення?
2. Який код використовують для лінійної інтерполяції та особливості його використання?
3. Які коди використовують для реалізації кругової інтерполяції та особливості цих кодів?
4. Які два способи використовують для форматування кадру кругової інтерполяції?
5. Які особливості побудови дуги з координатами центру дуги?
6. Які особливості побудови дуги через величину радіуса дуги R?

7. Для чого потрібно вказувати знак перед значенням радіусу дуги R?
8. Для чого використовують M-коди і як вони розташовуються в кадрі програми?
9. В чому різниця використання кодів зупинки виконання керуючої програми M00 та M01.
10. Які особливості використання кодів керування обертанням шпинделя M03; M04 та M05?
11. Які особливості використання кодів керування системою автоматичної подачі змащувально-охолоджувальної рідини M07; M08 та M09?
12. Як виконується автоматична заміна інструменту з використанням коду M06?
13. Який код використовують для компенсації довжини інструменту та чому необхідно обов'язково виконувати цю команду?
14. В чому різниця використання кодів завершення програми M30 та M02?

## 11. Постійні цикли верстата з ЧПК

**Постійними циклами** називають спеціальні мікропрограми, що закладені в пристрій ЧПК для виконання стандартних операцій механічної обробки.

Практично усі верстати з ЧПК мають набір циклів для обробки отворів – цикли свердління, розточування та нарізання різьби. Ці цикли спрощують процес написання керуючих програм та економлять час, оскільки дозволяють за допомогою одного кадру виконувати багато переміщень.

### 11.1. Цикли свердління

Якщо необхідно просвердлити кілька отворів, то щоб просвердлити один отвір, необхідно на робочій подачі опустити свердло на задану глибину, потім вивести його вверх на прискореній подачі і перемістити до наступного отвору.

Програма без використання постійних циклів матиме наступний вигляд:

%	
O0003	Початок програми
N100 G21	
N110 G00 G17 G40 G49 G80 G90	Рядок безпеки
N120 T1 M06	Виклик інструменту
N130 G00 G90 G54 X8.0 Y8.0 S1000 M03	Переміщення до отв. №1
N140 G43 H1 Z50.0	Корекція на довжину інструмента
N150 Z2.0	
N160 G01 Z-14.0 F80	Свердління отвору №1
N170 G00 Z2.0	Виведення свердла на прискореній подачі
N180 X20.0	Переміщення до отвору №2
N190 G01 Z-14.0 F80	Свердління отвору №2
N200 G00 Z2.0	Виведення свердла на прискореній подачі
N210 X-8.0	Переміщення до отвору №3
N220 G01 Z-14.0 F80	Свердління отвору №3
N230 G00 Z2.0	Виведення свердла на прискореній подачі
N240 X-20.0	Переміщення до отвору №4
N250 G01 Z-14.0 F80	Свердління отвору №4
N260 G00 Z2.0	Виведення свердла на прискореній подачі
N270 X-20.0 Y-8.0	Переміщення до отвору №5
N280 G01 Z-14.0 F80	Свердління отвору №5

N290 G00 Z2.0 Виведення свердла на прискореній подачі  
N300 X-8.0 Переміщення до отвору №6  
N310 G01 Z-14.0 F80 Свердління отвору №6  
N320 G00 Z2.0 Виведення свердла на прискореній подачі  
N330 X8.0 Переміщення до отвору №7  
N340 G01 Z-14.0 F80 Свердління отвору №7  
N350 G00 Z2.0 Виведення свердла на прискореній подачі  
N360 X20.0 Переміщення до отвору №8  
N370 G01 Z-14.0 F80 Свердління отвору №8  
N380 G00 Z2.0 Виведення свердла на прискореній подачі  
N390 Z50.0  
N400 M05  
N410 M30 Кінець програми  
%

Програма з використанням постійного циклу свердління значно спрощується, її легко читати.

%  
O0003 Початок програми  
N100 G21  
N110 G00 G17 G40 G49 G80 G90 Рядок безпеки  
N120 T1 M06 Виклик інструменту  
N130 G00 G90 G54 X8.0 Y8.0 S1000 M03 Переміщення до отвору №1  
N140 G43 H1 Z50.0 Корекція на довжину інструмента  
N150 Z2.0  
N160 G99 G81 Z-14.0 R2.0 F80 **Виклик циклу свердління**  
N170 X20.0 Координати отвору №2  
N180 X-8.0 Координати отвору №3  
N190 X-20.0 Координати отвору №4  
N200 X-20.0 Y-8.0 Координати отвору №5  
N210 X-8.0 Координати отвору №6  
N220 X8.0 Координати отвору №7  
N230 X20.0 Координати отвору №8  
N240 **Відміна циклу свердління**  
N250 Z50.0  
N260 M05  
N270 M30 Кінець програми  
%

Зрозуміло, що нова програма має менший розмір.

В кадрі **N160** знаходиться код **G81** для виклику циклу свердління. В цьому ж кадрі знаходяться адреси, що відповідають за настроювання параметрів циклу. Адреса **Z** означає глибину свердління, а **R** визначає висоту відведення свердла з отвору відносно нульової площини.

**В наступних кадрах знаходяться координати оброблюваних отворів. В них не потрібно ставити коди виклику циклу свердління, оскільки G81 буде активним, доки його не відмінять за допомогою коду G80.**

Верстати з ЧПК можуть мати і більш складні цикли для обробки контурів та кишень.

Деякі цикли стандартизовані, хоча більшість з них розроблюються виробниками верстатів та систем ЧПК самостійно, виходячи із свого бачення потреб користувачів. Тому на різних верстатах однакові цикли можуть записуватись по-різному, що утруднює програмування.

Розглянемо цикли для обробки отворів, що використовуються майже на усіх сучасних верстатах з ЧПК.

Таблиця 11.1

Цикли для обробки отворів

<b>G-код</b>	<b>Опис</b>
<b>G80</b>	Відміна постійного циклу
<b>G81</b>	Стандартний цикл свердління
<b>G82</b>	Свердління з витримкою
<b>G83</b>	Цикл переривчастого свердління
<b>G73</b>	Високошвидкісний цикл переривчастого свердління
<b>G84</b>	Цикл нарізання різьби
<b>G74</b>	Цикл нарізання лівої різьби
<b>G85</b>	Стандартний цикл розточування

## **G81 та G82 – Стандартний та цикл свердління з витримкою**

Код **G81** призначений для виклику стандартного циклу свердління.

### **G81 X8.0 Y16 Z-10.0 R2F80**

Адреси **X** та **Y** визначають координати оброблюваних отворів, а **Z** – кінцеву глибину свердління. **R** використовується для встановлення **площини відведення** – це координат та по осі **Z**, з якої починається свердління на робочій подачі і в яку повертається інструмент після того, як він досягнув дна оброблюваного отвору.

**Площина відведення** встановлюється дещо вище поверхні деталі, тому значення при **R** завжди позитивне.

**Площина відведення** не повинна бути дуже високо щоб не втрачати продуктивність при переміщенні свердла на робочій подачі. Робоча подача для циклу встановлюється за допомогою **F** слова даних.

### **Постійні цикли і їхні параметри є модальними.**

Викликавши цикл за допомогою відповідного **G**-коду, в наступних кадрах вказуються лише координати отворів, які необхідно обробити. Після кадру, в якому вказані координати останнього отвору, треба запрограмувати **G80** – код відміни (закінчення) постійного циклу.

Якщо цього не зробити, то всі наступні координати переміщень будуть сприйматись як координати оброблюваних отворів.

Цикл свердління з витримкою (**G82**) аналогічний стандартному з єдиною різницею, що свердло затримується на дні отвору. Цей цикл часто використовується для свердління глухих отворів щоб краще видалялась стружка з дна отвору.

Адреса **P** встановлює час очікування. Як правило, в тисячних долях секунди без десятинної крапки.

### **G82 X8.0 Y16 Z-10.0 P8500 R2 F80**

В цьому кадрі виконується цикл свердління з витримкою на дні отвору **8,5с**.

При використанні постійних циклів крім поняття **площини відведення** користуються також поняттям **вихідна площина** – це координата (рівень) по осі **Z**, в якій розташовується інструмент перед викликом постійного циклу.



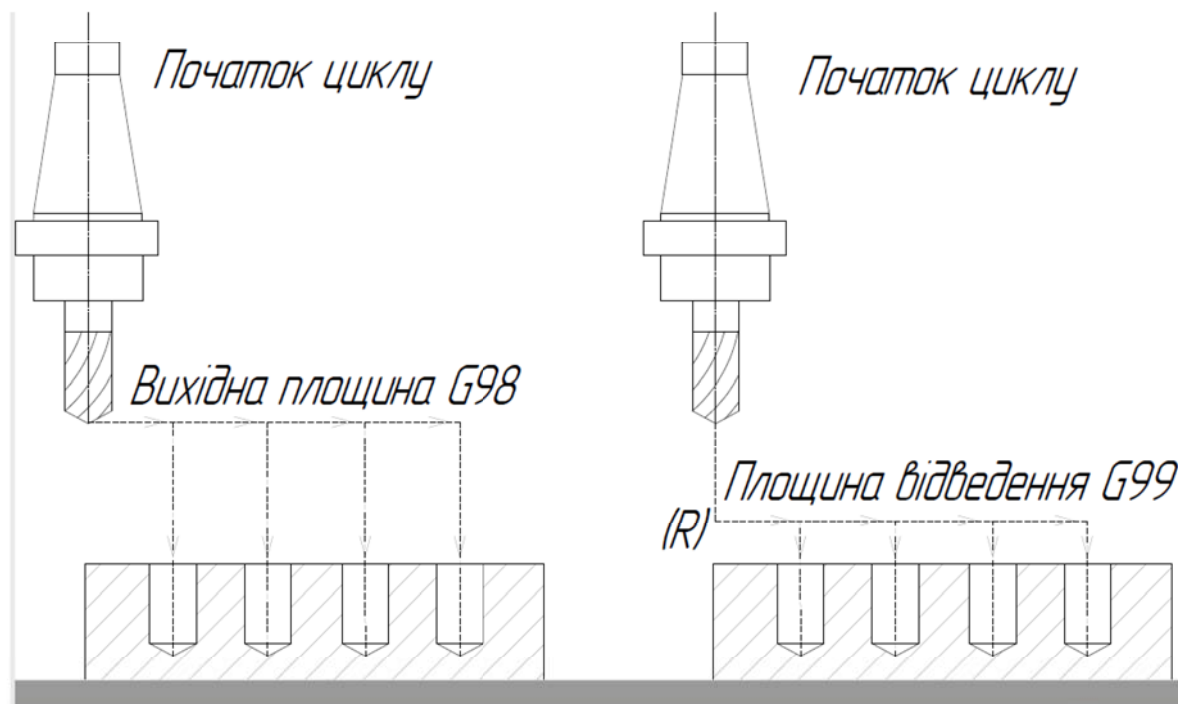


Рис. 11.1. Вихідна площина та площина відведення при використанні кодів **G98** та **G99**

Код **G98** використовується для роботи з вихідною площиною, а код **G99** - з площиною відведення.

**G98 X8.0 Y16 Z-10.0 F80**

або

**G99 X8.0 Y16 Z-10.0 R2 F80**

При використанні коду **G98** в постійному циклі інструмент кожний раз повертається у вихідну площину, а при використанні коду **G99** – в площину відведення з координатою **Z**, що встановлена **R**адресою.

Код **G98** використовується, коли потрібно виключити можливість зіткнення інструменту з деталлю. Необхідно пам'ятати, що при роботі з **G98** кодом після зміни інструменту вихідна площина буде встановлена дуже високо і це призведе до втрати часу.

Коли небезпеки зіткнення інструменту з деталлю немає, то використовують код **G99**, що дозволяє скоротити час при обробці кількох отворів.

Якщо площина відведення встановлена неправильно, то може бути зіткнення інструменту з деталлю (рис. 11.2).

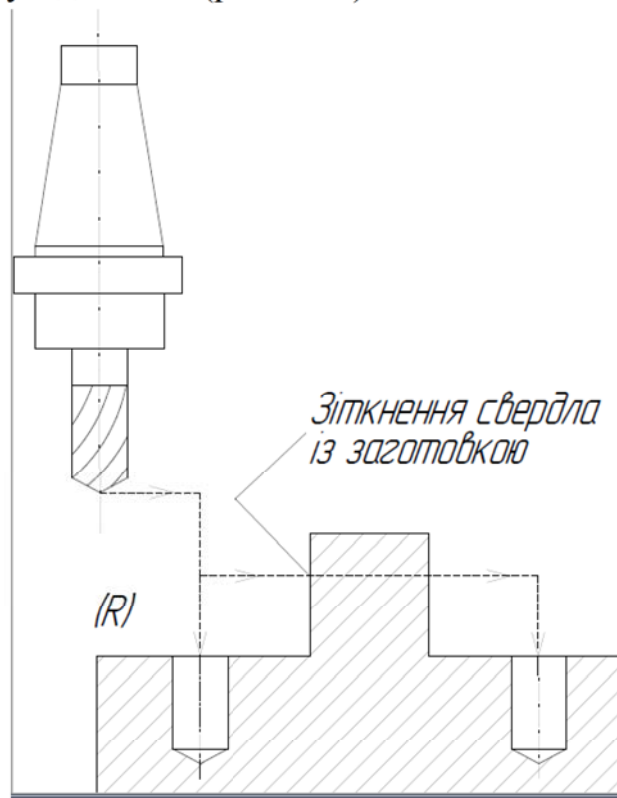


Рис. 11.2. Помилкове визначення площини відведення

Як правило, системи ЧПК дозволяють переключатись між **G98** та **G99** прямо всередині постійного циклу між оброблюваними отворами.

...

```
G99 G81 X8.0 Y16.0 Z-10.0 R2.0 F80  
X20.0 Y40.0  
G98 X40.0 Y30.0
```

...

### **Відносні координати в постійному циклі**

При необхідності використання відносних координат (**G91**) необхідно пам'ятати наступне:

Площина відведення встановлюється відносно вихідної площини.

Глибина свердління по осі *Z* встановлюється відносно площини відведення.

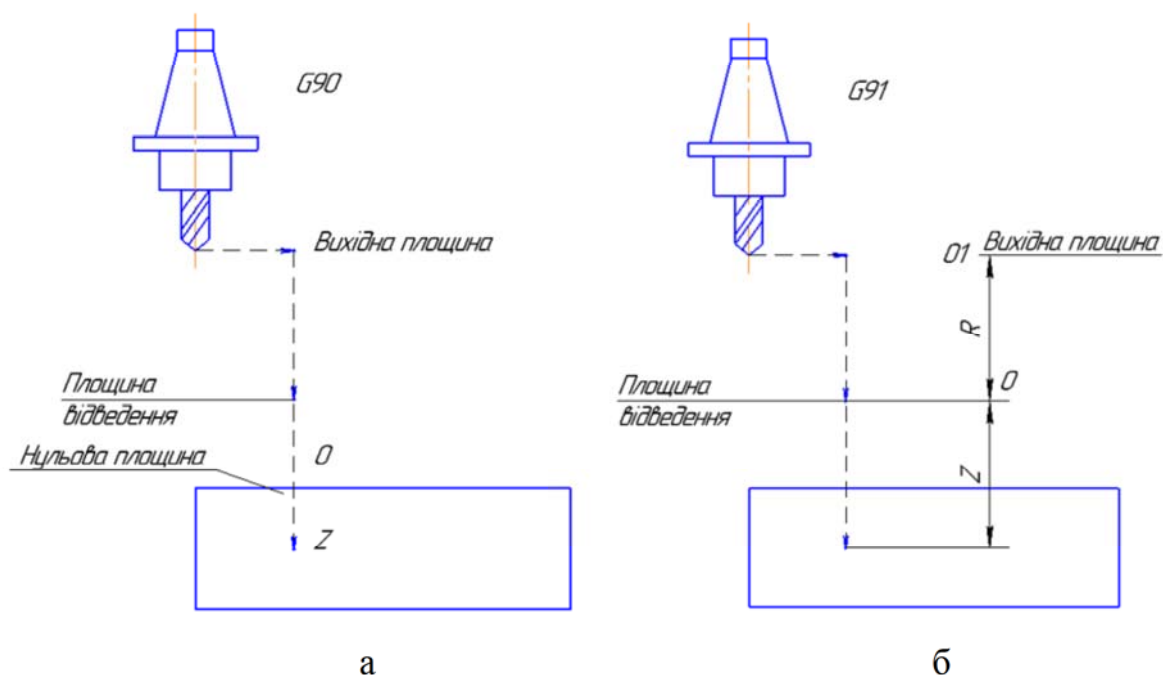


Рис. 11.3. Різниця між G90 (а) та G91 (б) в постійних циклах свердління

Приклад використання абсолютних і відносних координат в постійному циклі представлено в таблиці 11.2.

Таблиця 11.2

Абсолютні та відносні координати в постійному циклі

Абсолютні координати	Відносні координати
%	%
O0001	O0002
N100 G21	N100 G21
N105G00 G17 G40 G49 G80 <b>G90</b>	N105G00 G17 G40 G49 G80 <b>G91</b>
N110 T3M6	N110 T3M6
N115 G00 G90 G54 X-14 Y8 S1000 M3	N115 G00 G90 G54 X-14 Y8 S1000 M3
	N120 G91
N120 G43 H3 Z50	N125 G43 H3 Z50
<b>N125 G98 G81 Z-12 R2 F80</b>	<b>N130 G98 G81 Z-14 R-48 F80</b>
N130 X16	N135 X30
N135 G80	N140 G80
N140 M5	N145 M5
N145 G91 G00 G28 Z0	N150 G00 G28 Z0
N150 G28 X0 Y0	N155 G28 X0 Y0
N155 M30	N160 M30
%	%

У випадку відносного програмування глибина свердління (**12мм** нижче нульової площини) встановлюється відносно площини відведення, тому в програмі вказано **Z-14**.

Площина відведення (**2мм** нижче нульової площини) задана відносно координати **Z50.0**, тому в циклі свердління вказано **R-48**.

### **G83 та G73 - Цикли переривчастого свердління**

Переривчасте свердління (рис. 11.4) часто використовується при обробці глибоких отворів, коли через встановлені проміжки часу інструмент піднімається вгору для видалення стружки, оскільки є небезпека, що стружка самостійно не встигне вийти з отвору і інструмент може заклинитись і поламатись. Оптимальність цикл встановлює технолог-програміст виходячи з конкретних умов. Цикл переривчастого свердління викликає код G83.

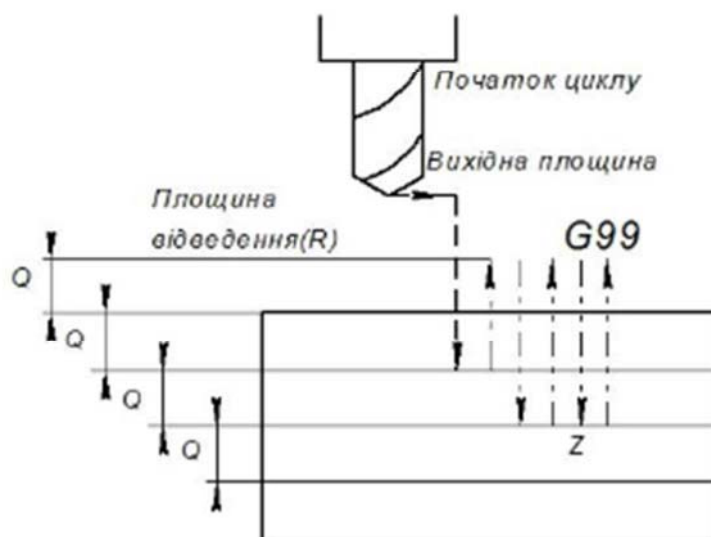


Рис. 11.4. Схема циклу переривчастого свердління

Формат кадру для циклу переривчастого свердління схожий на формат звичайного циклу свердління.

...

**G83 X15.0 Y15.0 Z-20.0 Q3.0 R1.0 F60**

...

**Q**-адреса визначає відносну глибину кожного робочого ходу свердла. В даному випадку свердління проходить по такому алгоритму:

1. Свердло від вихідної площини переміщується до площини відведення (**R1.0**) на прискореній подачі.

2. Від площини відведення (**R**) свердло подається на глибину 3мм (**Q3.0**) з подачею (**F60**).
3. Свердло прискорено переміщується до площини відведення (**R1.0**).
4. Свердло прискорено переміщується до раніше досягнутої позиції по глибині (або трохи не доходить до неї для виключення зіткнення свердла з матеріалом заготовки).
5. Свердло переміщується на глибину **6мм (3+3)** зі швидкістю подачі (**F60**).
6. Кроки 3,4 та 5 повторюються доти, доки свердло не досягне координати **Z-20.0**. Після цього свердло виводиться з отвору до площини відведення (**G99**) або вихідної площини (**G98**).

При написанні керуючої програми для свердління необхідно враховувати наступне:

На кресленнях довжину отвору вказують по прямій частині, але різальна кромка заточена під кутом **2φ** (рис. 11.5).

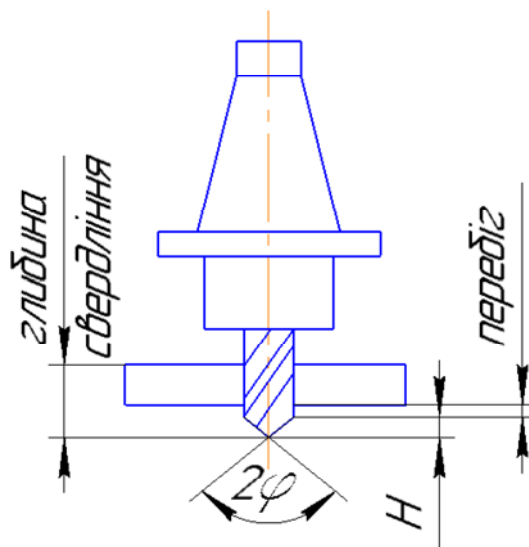


Рис. 11.5. Схема визначення координати Z при свердлінні

В програмі вказується координата **Z** для поперечної кромки свердла тобто вершини, тому інструменту необхідно пройти додаткову відстань

$$H = R / \tan \varphi$$

Для свердла діаметром 12мм при куті  $\varphi = 120^\circ$  додаткова висота  $H = 6 / \tan 60^\circ = 3.46$ мм. Таким чином, в керуючій програмі треба вказати глибину свердління **Z = 23.46** замість 20.0.

При свердлінні наскрізних отворів свердлу необхідно задати невеликий перебіг (**0.5 – 1мм**) для прямої частини свердла.

В цьому випадку в керуючій програмі глибина свердління  $Z = 24.46$  замість 20.0.

Високошвидкісний цикл переривчастого свердління **G73** працює аналогічно циклу **G83**. Єдина відмінність в тому, що при високошвидкісному циклі свердло виводиться з отвору не повністю, що дозволяє зменшити машинний час.

Формат кадру

...

**G73 X15.0 Y15.0 Z-20.0 Q3.0 R1 F60**

...

### 11.2. Цикл нарізання різьби - G84

При нарізанні різьби кожній подачі по осі  $Z$  на величину кроку нарізуваної різьби шпиндель обертається на один оберт. Коли мітчик досягне дна отвору, шпиндель, обертаючись в іншу сторону, виводить мітчик з отвору. Пристрій ЧПК самостійно синхронізує подачу та швидкість обертання шпинделя, що виключає можливість пошкодження різьби та поломки інструменту.

Для виклику циклу **нарізання різьби** використовується код **G84**.

Формат кадру для циклу нарізання різьби має наступний вигляд:

...

**G98 G84 X10.0 Y10.0 Z-6.0 R10.0 F10**

...

Код **G74** викликає цикл **нарізання лівої різьби мітчиком**.

Формат цього циклу аналогічний формату для **G84**. Єдина різниця лише в напрямку обертання шпинделя.

...

**G98 G74 X15.0 Y15.0 Z-18.0 R2.0 F10**

...

Деякі системи ЧПК дозволяють програмувати цикли нарізання різьби аналогічно циклу переривчастого свердління.

Використовуючи постійні цикли нарізання різьби потрібно бути особливо уважними при призначенні режимів різання та глибини обробки.

### 11.3. Цикли розточування

Код **G85** викликає **стандартний цикл розточування**. Інструментом слугує розточувальний патрон з різцем, що настроєний на потрібний розмір. Інструмент вводиться в отвір на робочій подачі. Після досягнення заданої координати виводиться з отвору також на робочій подачі.

Формат схожий на формат циклу свердління

...

**G98 G85 X15.0 Y15.0 Z-18.0 R2.0 F10**

...

Використовують багато різновидностей циклу розточування, які відрізняються в основному поведінкою при виведенні інструменту з розточеного отвору (табл. 11.3).

Таблиця 11.3

Розточувальні цикли

Цикл розточування	Опис циклу
<b>G76</b>	Після досягнення заданої координати різець орієнтується відповідним чином, зміщується від обробленої поверхні та виводиться на прискореній подачі. Різець необхідно правильно орієнтувати при настроюванні та установці щоб не поламати та не пошкодити деталь.
<b>G86</b>	Після досягнення заданої координати шпиндель перестає обертатись та виводиться на прискореній подачі. На обробленій поверхні може залишитись вертикальна риска.
<b>G87</b>	Може реалізовуватись по-різному. В одних верстатах цей цикл виконує розточування за декілька робочих ходів, аналогічно переривчастому свердлінню. В інших шпиндель зупиняється на дні отвору та виводиться з нього вручну. На більшості сучасних оброблюваних центрів є циклом зворотного розточування.
<b>G88</b>	Аналогічний <b>G87</b> . На дні отвору можна задати час витримки.
<b>G89</b>	Аналогічний <b>G85</b> . На дні отвору можна задати час витримки.

## 11.4. Приклад програми на свердління з допомогою постійних циклів

Просвердли 4 отвори (1-4) діаметром 8мм на глибину 20мм та діаметром 12мм (5-8) на глибину 52мм з попереднім центруванням (рис. 11.6, табл. 11.4).

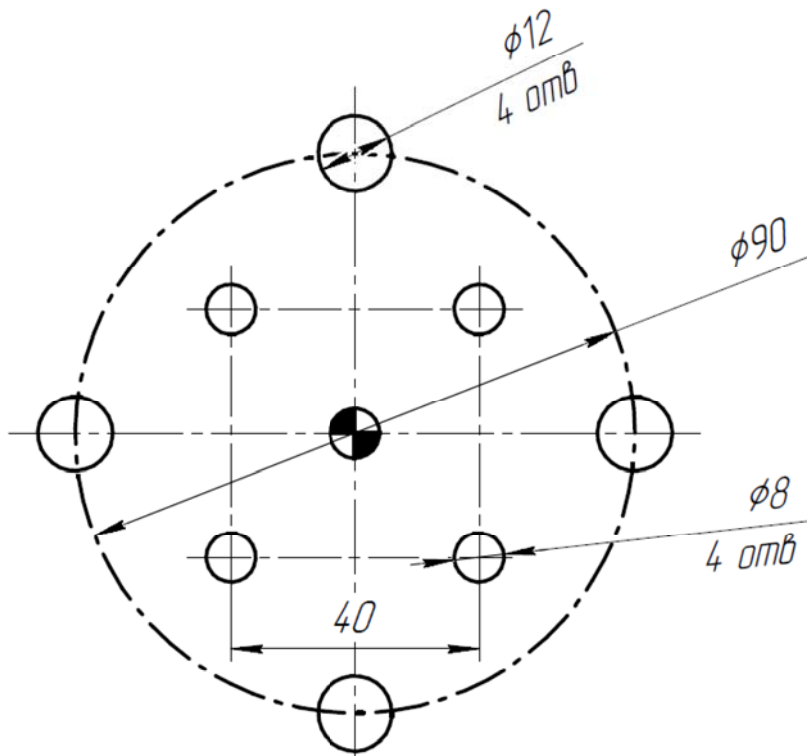


Рис. 11.6. Схема розташування отворів

Таблиця 11.4

Координати отворів в прямокутній системі координат

№ отвору	X	Y
1	20	20
2	20	-20
3	-20	-20
4	-20	20
5	0	45
6	45	0
7	0	-45
8	-45	0



Створюємо керуючу програму (табл. 11.5).

Таблиця 11.5

Керуюча програма обробки отворів

Код програми	Опис
%	
O0005	Номер програми
(PROGRAM NAME – OTVORI)	Назва програми
N100 G21	Робота в метричній системі
N105G00 G17 G40 G49 G80 G90	Рядок безпеки
(CENTRUVANNJA)	Коментар
N110 N1 M6	<b>Виклик центрального свердла</b>
N115G54 X20 Y20 S1200 M3	Переміщення до отвору №1
N120 G43 H1 Z80	Компенсація довжини ін-ту
N125 Z2	Прискор. переміщення до Z2
N130 G99 G81 Z-6 R2 F80	<b>Стандартний цикл свердління</b>
N135 X20 Y-20	Центрування отвору №2
N140 X-20 Y-20	Центрування отвору №3
N145 X-20 Y20	Центрування отвору №4
N150 X0 Y45	Центрування отвору №5
N155 X45 Y0	Центрування отвору №6
N160 X0 Y-45	Центрування отвору №7
N165 X-45 Y0	Центрування отвору №8
N170 G80	<b>Відміна постійного циклу</b>
N175 Z80	Переміщення до Z80
N180M5	Зупинка шпинделя
N185 G91 G28 Z0	Повернення у вих.. поз. по Z
N190 G28 X0 Y0	Повернення у вих. поз. по X,Y
N195 M01	Тимчасова зупинка
(DRILL 4 HOLES 1-4)	Коментар
N200 N2 M6	<b>Виклик свердла 68</b>
N205 G54 X20 Y20	Переміщення до отвору №1
N210 G43 H2 Z80	Компенсація довжини ін-ту
N215 Z2	Прискор. переміщення до Z2
N220G99 G81 Z-20 R2 F60	<b>Стандартний цикл свердління</b>
N225 X20 Y-20	Свердління отвору №2
N230 X-20 Y-20	Свердління отвору №3
N235 X-20 Y20	Свердління отвору №4
N240 G80	<b>Відміна постійного циклу</b>
N245 Z80	Переміщення до Z80
N250 M5	Зупинка шпинделя
N255 G91 G28 Z0	Повернення у вих.. поз. по Z

Код програми	Опис
N260 G28 X0 Y0	Повернення у вих. поз. по X,Y
N265 M01	Тимчасова зупинка
(DRILL 4 HOLES 5-8)	Коментар
N270 T3 M6	<b>Виклик свердла <math>\phi 12</math></b>
N275 G54 X0 Y45 S900 M3	Переміщення до отвору №5
N280 G43 H3 Z80	Компенсація довжини ін-ту
N285 Z2	Прискор. переміщення до Z2
N290 G99 G83 Z-52 Q4.0 R2 F60	<b>Цикл переривчастого свердління</b>
N295 X45 Y0	Свердління отвору №6
N300 X0 Y-45	Свердління отвору №7
N305 X-45 Y0	Свердління отвору №8
N310 G80	<b>Відміна постійного циклу</b>
N315 Z80	Переміщення до Z80
N320 M5	Зупинка шпинделя
N325 G91 G28 Z0	Повернення у вих.. поз. по Z
N330 G28 X0 Y0	Повернення у вих. поз. по X,Y
N335 M30	Кінець програми
%	

### Запитання до розділу 11

1. Що розуміють під постійними циклами та для чого їх використовують?
2. Які основні постійні цикли використовують у верстатах з ЧПК?
3. Які відмінності використання циклів свердління G81 та G82?
4. Як встановлюється площина відведення R?
5. Як встановлюється час витримки при використанні коду G82?
6. В чому різниця понять «площина відведення» та «вихідна площина»?
7. Чим відрізняється використання кодів G98 та G99 для виведення інструменту?
8. Які особливості використання відносних координат в постійних циклах?
9. Яка відмінність у використанні кодів G90 та G91 в постійних циклах?
10. Які особливості використання кадрів переривчастого свердління G83 та G73?
11. Які особливості формату кадру переривчастого свердління?

12. Як враховується в кадрі програми довжина різальної частини свердла для забезпечення довжини отвору, що задана кресленням?
13. Які особливості використання кодів для нарізання різьби G84 та G74?
14. Як виглядає формат кадру для нарізання різьби?
15. Які особливості реалізації стандартного циклу розточування за допомогою коду G85?
16. Як виглядає формат кадру для циклу розточування?
17. Які різновидності циклу розточування можуть використовуватись на верстаті ЧПК?

## 12. Автоматична корекція радіуса інструмента

При контурному фрезеруванні кінцевою фрезою необхідно в керуючій програмі передбачити зміщення фрези на величину радіуса, тому що обробка виконується боковою стороною.

Так, наприклад, ставиться задача фрезерування зовнішньої поверхні заготовки 25X50 кінцевою фрезою діаметром 10мм (рис. 12.1).

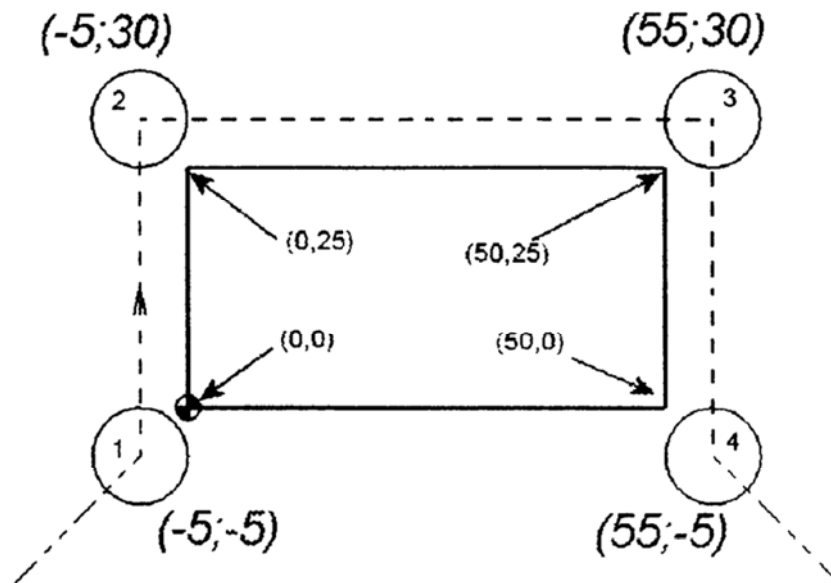


Рис. 12.1. Послідовне переміщення фрези через опорні точки 1, 2, 3, 4

Центр фрези зміщено на величину радіуса вліво від оброблюваної поверхні.

В керуючій програмі вказують координати центра фрези і її можна представити в такому вигляді:

```
...  
G01 X-5 Y-5  
G01 X-5 Y30  
G01 X55 Y30  
G01 X55 Y-5
```

Координати опорних точок розраховані для фрези  $\phi 10$ . Якщо раптом такої фрези не знайшлось, а є фреза  $\phi 9$ , то необхідно перерахувати усі координати траєкторії (рис. 12.2). При оброблюванні новою фрезою її центр буде зміщений вліво на 4,5мм і треба скласти нову програму.

...  
G01 X-4.5 Y-4.5  
G01 X-4.5 Y29.5  
G01 X54.5 Y29.5  
G01 X54.5 Y-4.5  
...

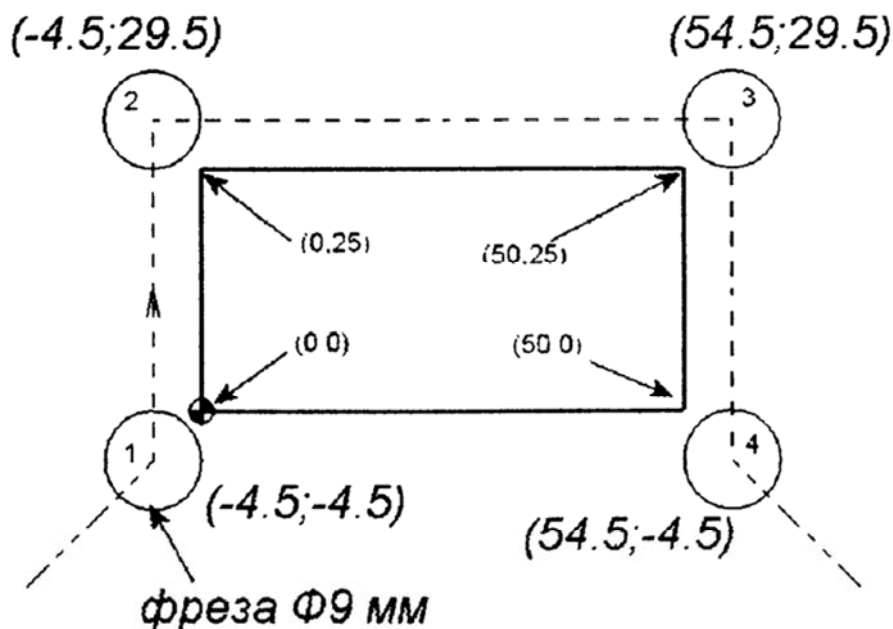


Рис. 12.2. Еквідистантна траєкторія для фрези діаметром 9мм

В даному випадку перерахунок простий, але якщо оброблюваний контур складний, а діаметр фрези не ціле число, то розрахунок координат траєкторії інструменту буде трудомістким.

Якраз функція верстата з ЧПК для автоматичної корекції радіуса інструмента спрощує такі розрахунки і **дозволяє по одній керуючій програмі вести обробку контуру різними інструментами.**

При створенні програми обробки вважають, що центр фрези переміщується по контуру без зміщення.

...  
G01 X0Y0  
G01 Y25  
G01 X50  
G01 Y0  
...

Система ЧПК сама розрахує величину зміщення, тобто виконає **корекцію на радіус інструменту.**

Для цього треба вказати системі радіус інструменту та в керуючій програмі вказати відповідний **G-код**, який викликає зміщення інструменту відносно вихідної траєкторії вправо або вліво (табл. 12.1, рис. 12.3).

Таблиця 12.1

Гкоди зміщення інструменту

Код	Дія
<b>G41</b>	Корекція зліва
<b>G42</b>	Корекція справа
<b>G40</b>	Відміна корекції
<b>D</b>	Номер коректора на радіус інструменту

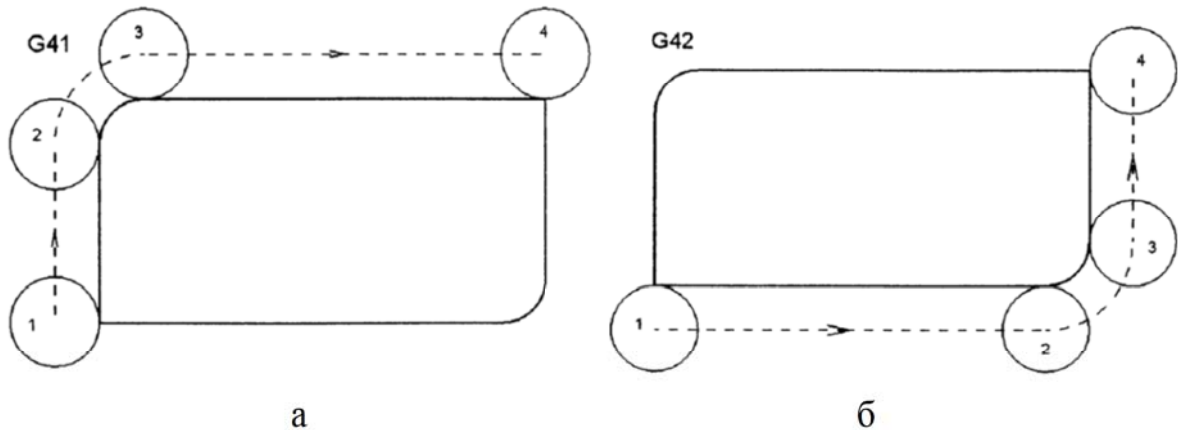


Рис. 12.3. Зміщення інструменту зліва G41 (а) та справа від контуру G42 (б)

Напрямок корекції визначається, якщо дивитись на траєкторію зверху, тобто зі сторони "+Z" в напрямку "-Z".

Тепер можна створити повну програму з корекцією, яка забезпечить правильну обробку даного контуру фрезою діаметром 9мм (рис. 12.4, табл.12.2).

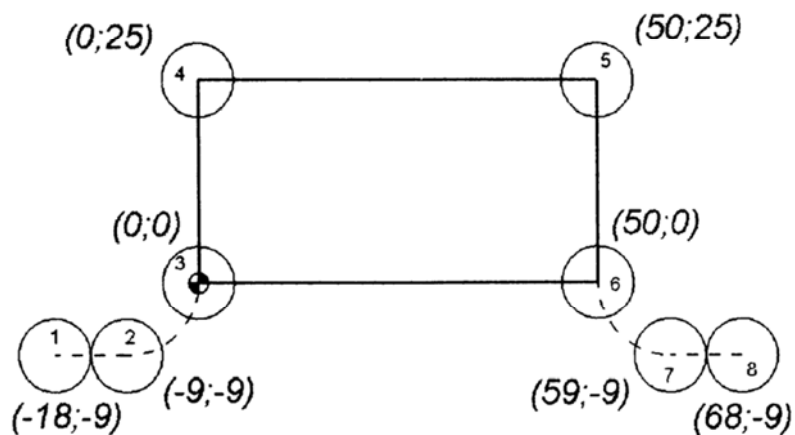


Рис. 12.4. Оброблювання контуру з корекцією, ділянками підведення та відведення (опорні точки траєкторії співпадають з опорними точками контуру деталі)

Таблиця 12.2

Програма з корекцією радіусу інструмента

Код програми	Пояснення
%	
O0006	Номер програми
(FREZERUVANNJA CONTURU)	Назва програми
N100 G21	Робота в метричній системі
N105G00 G17 G40 G49 G80 G90	Рядок безпеки
N110 T2 M6	Виклик інструменту №2
N115 G54 X-18 Y-9 S1000 M3	Прискор. переміщення в поз. №1
N120 H2 Z100	Компенсація довжини ін-ту.
N125 Z10	Прискор. переміщення в Z10
N130 G01 Z-1 F120	Переміщення в Z-1 з 120мм/хв
N135 G41 D2 X-9 F200	Переміщення з корект. в поз. №2
N140X0 Y0 R9	Підхід до контуру по дузі в поз. №3
N145 Y25	Переміщення в поз. №4
N150 X50	Переміщення в поз. №5
N155 Y0	Переміщення в поз. №6
N160 G03 X59 Y-9 R9	Відведення від контуру по дузі в поз. №7
N165 G01 G40 X68	Переміщення і відміна корекції
N170 Z9 F300	Підйом інструменту в Z9
N175 G00 Z100	Прискор. переміщення в Z100
N180 M5	Зупинка шпинделя
N185 M30	Кінець програми
%	

В кадрі №135 активується автоматична корекція радіусу інструмента. Код включає корекцію зліва, а слово даних означає номер коректора із значенням радіусу інструмента.

В пам'яті кожної сучасної системи ЧПК є область коректорів – це таблиця інструментів, де зберігаються значення довжин та радіусів інструментів (табл. 12.3).

Таблиця 12.3

Область коректорів (таблиця інструментів) системи ЧПК

№ інструменту	довжина	радіус	інші параметри
1	100	5	...
2	88	4.5	...
3	134	6	...

Якщо ми працюємо фрезою №2 діаметром 9мм, то у відповідному коректорі №2 повинно знаходитись значення 4.5. Якраз на цю величину і зміщується центр інструменту вліво відносно запрограмованого контуру.

Таким чином, якщо на попередньому рисунку показана розрахункова траєкторія, що співпадає з вихідним контуром, то на даному рис.12.5 показана реальна траєкторія обробки фрезою діаметром 9мм, зміщеною на 4.5мм вліво відносно вихідного контуру.

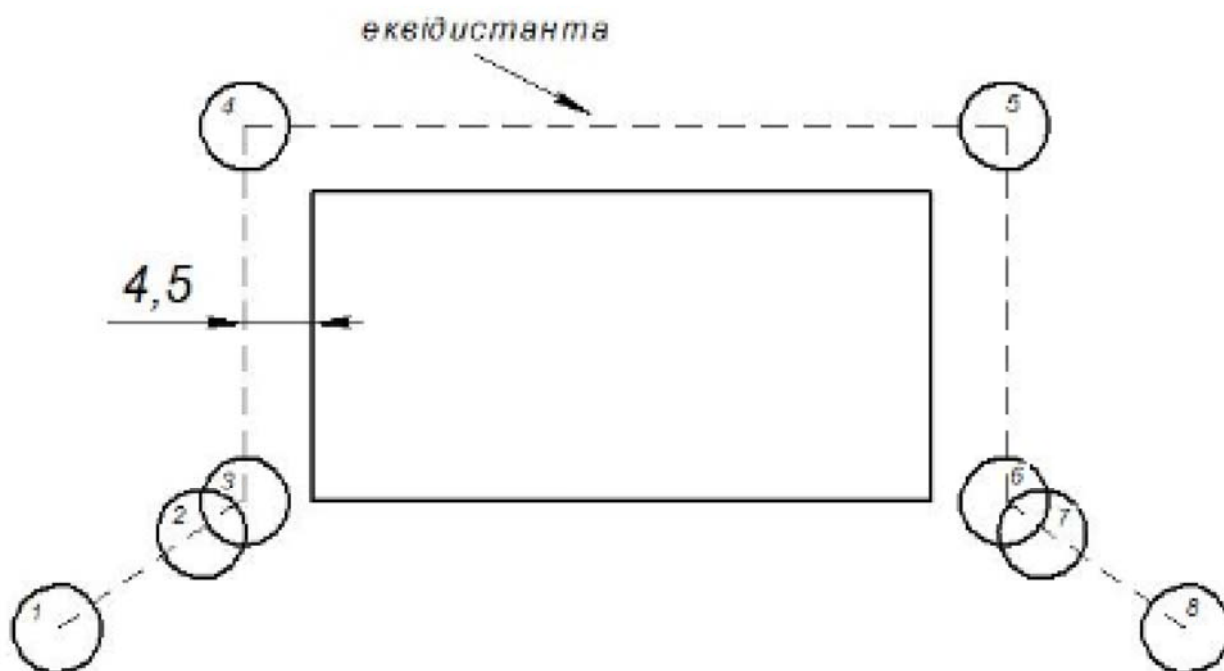


Рис. 12.5. Реальна траєкторія переміщення інструменту при обробці з корекцією радіусу

Для того, щоб система ЧПК встигла виконати зміщення відносно запрограмованого контуру, **необхідно додати до вихідної траєкторії ділянку підведення**. На цій ділянці **активується автоматична корекція радіусу інструмента**.

Більшості систем для активації корекції необхідно пройти відстань, не меншу від радіуса інструмента.

**Обов'язкове виконання компенсації довжини інструменту перед активацією корекції радіусу інструмента.**

Поглянемо на кадр №135

...

**N120G43 H2 Z100**

...



## N135G41 D2 X-9. F200

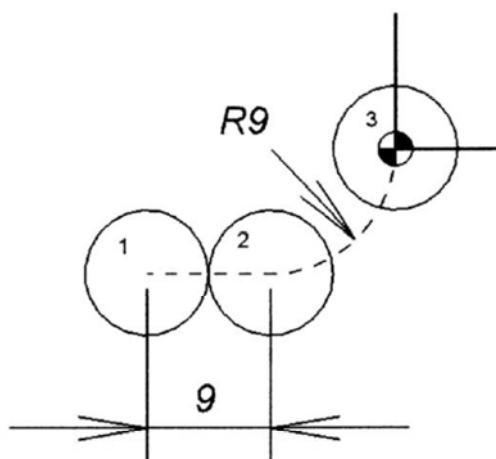


Рис. 12.6. Прямолинійна ділянка для активації корекції та ділянка плавного підходу (по дотичній) до контуру

Код **G41** включає корекцію зліва, **D2** зчитує значення радіуса інструменту з коректора №2, запрограмоване лінійне переміщення в **X-9** (переміщення із позиції №1 в позицію №2) під час якого інструмент зміщується вліво відносно запрограмованого контуру і на позицію №2 інструмент підходить вже по еквідистантній траєкторії.

Потім програмують ділянку плавного підходу інструменту до оброблюваної поверхні – по дотичній (з позиції №2 в позицію №3 – до першої точки контуру).

**Для активації корекції обов'язковою умовою є наявність якраз прямолинійного переміщення на робочій подачі** (рис. 12.6).

При спробі активації корекції на радіус разом з переміщенням по дузі система ЧПК видасть повідомлення про помилку.

При роботі з корекцією на радіус є ряд обмежень.

1. Якщо запрограмований радіус контуру виявиться меншим значення радіусу інструмента в коректорі, то система ЧПК видасть повідомлення про помилку.

2. В багатьох верстатах корекція може виконуватись лише в площині **X-Y**.

3. В деяких випадках помилкою вважається лінійне переміщення з кроком, що менший від значення радіусу у відповідному коректорі.

4. Багато систем ЧПК зберігають інформацію про радіус та довжину інструмента в різних коректорах. В цьому випадку числові значення **D** та **H** слів даних будуть різними (табл. 12.4).

Таблиця 12.4

Зберігання інформації в коректорах

№ коректора	Значення
1	100
2	88
3	134
...	...
101	5
102	4.5
103	23.08
...	...

...

N120 G43 H2 Z100

...

N135G41 D102 X-9. F200

...

Після того, як інструмент обійшов контур повністю, потрібно плавно відвести інструмент та відмінити корекцію за допомогою кодуG40 або D00.

Оптимальний варіант – спочатку відведення по дотичній, а потім кадр з лінійним переміщенням та кодом відміни корекції.

...

N160G3 X59. Y-9. R9 Відхід від контуру по дотичній

N165G1 G40 X68 Відміна корекції на радіус

...

Функція автоматичної корекції дозволяє працювати по одній програмі інструментами різних діаметрів. Все ж рекомендується, щоб розрахунковий та фактичний діаметри відрізнялись не дуже сильно. Якщо розрахунок на фрезу діаметром 10мм, то використання фрези діаметром 9мм не викликає проблем, а при використанні фрези діаметром 40мм може появиться повідомлення про помилку або виникне заріз контуру.

Таким чином, функція автоматичної корекції радіуса інструменту дозволяє працювати різними інструментами по одній програмі, але головне призначення цієї функції – це керування розмірами деталі при фрезеруванні.

Наприклад, після обробки заготовки фрезою діаметром 9мм ми виявили, що розмір деталі 25.02мм, а в кресленні задано розмір 25-0.02мм.

В цьому випадку необхідно просто зменшити значення радіуса фрези в коректорі з 4.5мм до 4.49мм і запустити програму на виконання ще раз. Система ЧПК буде вважати, що використовується інструмент з меншим радіусом і змістить траєкторію центра інструменту діаметром 4.5мм на 4.49мм відносно вихідного контуру. Після обробки контур на 0.02мм зменшиться (по 0.01мм з кожної сторони) і буде отримано заданий розмір в 25мм.

Область коректорів (таблиця інструментів) системи ЧПК з урахуванням реального розміру деталі після обробки представлена в табл.12.5.

Таблиця 12.5

Таблиця інструментів з урахуванням реального розміру деталі

№ інструменту	довжина	радіус	Інші параметри
1	100	5	...
2	88	(4.5) 4.49	...
3	134	6	...
...	....	...	...

Таким чином, змінюючи значення радіусів в коректорах, можна отримати розміри деталі в межах допуску без перерахунку програми.

Ще однією великою перевагою використання функції автоматичної корекції радіуса інструменту є **можливість працювати безпосередньо з кресленням контуру деталі**, тобто координати практично всіх опорних точок очевидні і їх можна «взяти» прямо з креслення без якихось додаткових розрахунків.

## Запитання до розділу 12

1. Чим викликана необхідність корекції радіусу інструмента при контурному фрезеруванні?
2. Яка послідовність виконання корекції на радіус інструменту?
3. Які коди використовують для зміщення інструменту відносно вихідної траєкторії на величину радіусу?
4. Що таке область коректорів в пам'яті системи ЧПК?
5. Для чого необхідно використовувати ділянку плавного підведення до вихідної траєкторії?
6. Чи обов'язкове виконання компенсації довжини інструменту перед активацією корекції радіусу інструменту?

7. Які обмеження існують при роботі верстата з корекцією на радіус інструменту?
8. Як виконується плавне відведення інструменту після повного проходження контуру?
9. За допомогою яких кадрів можна відмінити корекцію на радіус інструменту?
10. Які переваги надає використання функції автоматичної корекції радіусу інструмента?

### 13. Підпрограми

Мова G та M-кодів, як і люба інша мова програмування, дозволяє працювати з підпрограмами та виконувати переходи.

За допомогою функції підпрограми основна керуюча програма може викликати з пам'яті іншу програму (підпрограму) і виконати її потрібну кількість разів.

Якщо в керуючій програмі знаходиться дія, що часто повторюється, то використання підпрограм дозволяє спростити програму обробки і зробити її значно меншою в розмірі.

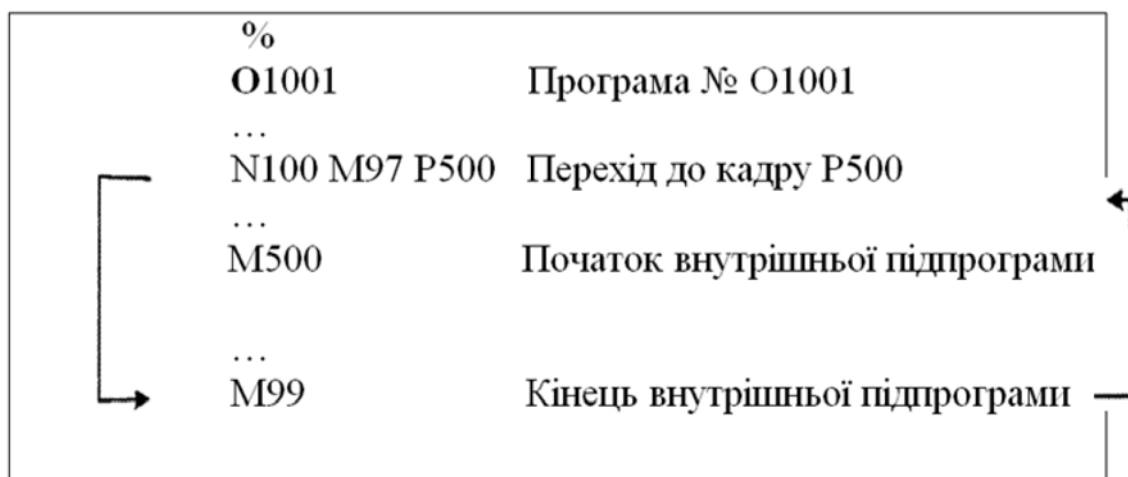
Розрізняють два види підпрограм – внутрішні та зовнішні.

Внутрішні підпрограми викликаються за допомогою коду M97 і знаходяться всередині головної програми, тобто знаходяться в одному файлі.

Зовнішні підпрограми викликаються кодом M98 і не знаходяться в тілі головної програми. В цьому випадку головна програма і підпрограма знаходяться в різних файлах.

Таблиця 13.1

Схема внутрішньої підпрограми



Внутрішня підпрограма виконується, коли система ЧПК зустрічає код M97. При цьому адреса P вказує на номер кадру, де починається внутрішня підпрограма і до якого треба перейти. Коли система ЧПК знаходить кадр з кодом закінчення підпрограми M99, то виконання внутрішньої підпрограми закінчується і керування передається кадру головної програми, наступному за кадром, що викликав закінчену підпрограму.

Таблиця 13.2  
Схема зовнішньої підпрограми



Зовнішні підпрограми працюють подібним чином. Коли в головній програмі зустрічається кадр з кодом M98, то викликається підпрограма з номером, що встановлений за допомогою P адреси, а при знаходженні коду M99 керування повертається головній програмі, тобто виконується кадр головної програми, наступний після кадру M98.

Зовнішня підпрограма знаходиться в окремому файлі, тобто це фактично окрема програма з індивідуальним номером, яка при бажанні може виконуватись незалежно від головної програми.

Для виклику підпрограми необхідно, щоб вона знаходилась в пам'яті ЧПК.

Приклад керуючої програм и з внутрішньою підпрограмою представлено в табл. 13.3.

За допомогою L адреси визначаються скільки разів потрібно викликати ту чи іншу підпрограму.

Якщо підпрограму потрібно викликати лише один раз, то L в кадрі можна не вказувати.

...

M98 P1000 L4 – підпрограма буде викликана 4 рази

...

Таблиця 13.3  
Керуюча програма з внутрішньою підпрограмою

<pre> % O1023 N10 G21 G40 G49 G54 G80 G90 N20 T3 M06 N30 G43 H3 N40 M03 S1000 N50 G00 X0 Y0 N60 Z0.5 N70 <b>M97 P200</b> N80 M05 N90 M02 <b>N200 G01 X-0.5 F50</b> N210 X10 Y10 N220 Z0.5 M230 <b>M99</b> %</pre>	<pre> Програма №1023 Рядок безпеки Виклик інструменту №3 Компенсація довжини <u>ін.-ту</u> Вмикання обертів шпинделя <u>Позиціювання в X0 Y0</u> <u>Позиціювання в Z0.5</u> Виклик внутрішньої <u>підпрогр.</u> Вимикання обертів шпинделя Закінчення програми Початок внутр. програми ... ... Кінець внутр.. програми</pre>
---	--

Великою перевагою використання підпрограм є можливість зручної та ефективної роботи з програмними масивами та шаблонами.

Приклад:

На рисунку зображена деталь, що має чотири групи отворів діаметром 3мм. Нульовою точкою є верхній кут деталі.

Необхідно створити програму обробки, що включає головну програму та підпрограму, щоб спростити роботу.

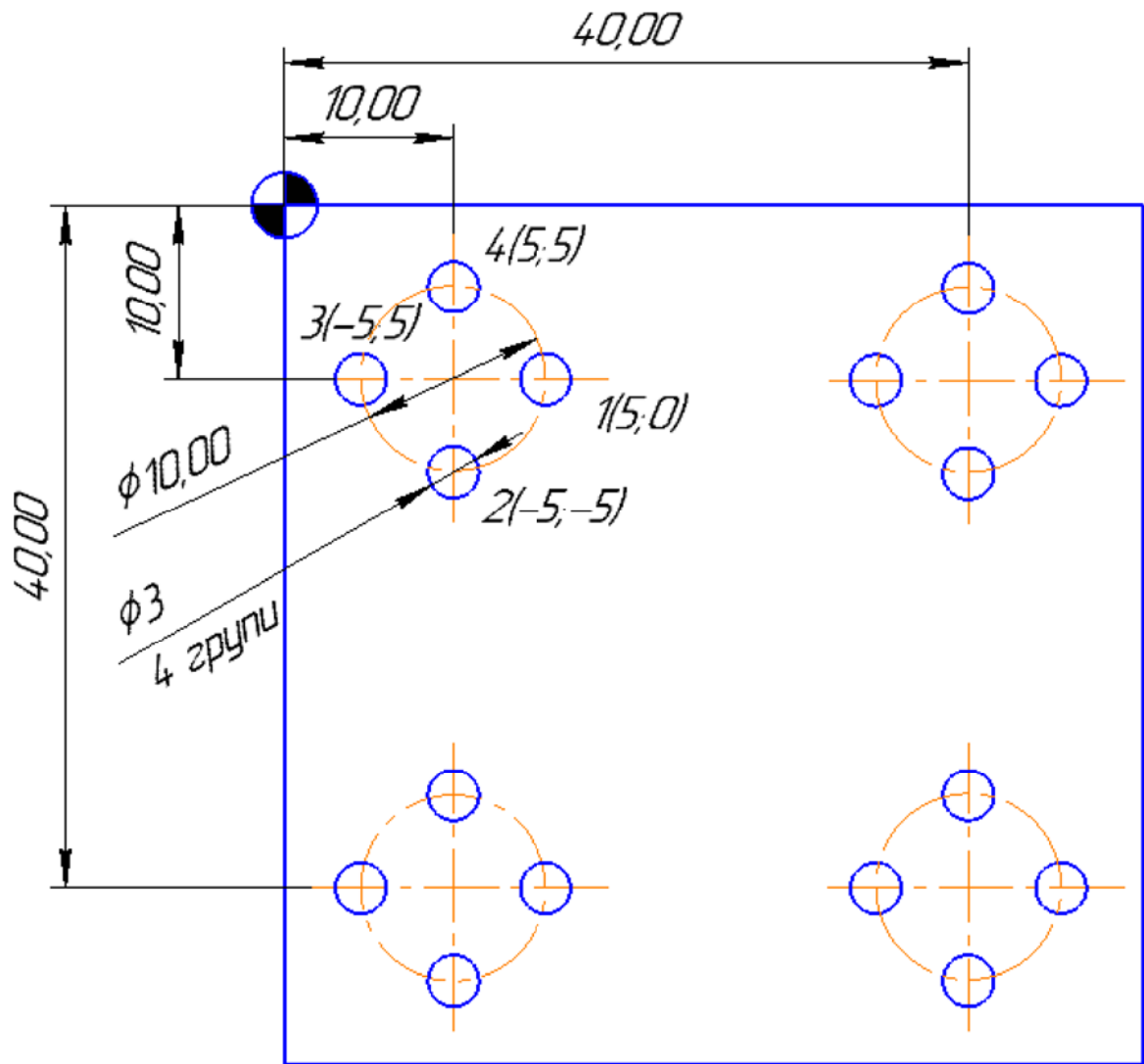


Рис. 13.1. Деталь, що має чотири групи однакових отворів діаметром 3мм

Нульовою точкою є верхній кут деталі.

Спочатку створюється головна програма, яка буде переміщувати інструмент до кожної групи отворів.

Потім створюється підпрограма, що необхідна для свердління чотирьох отворів в одній групі.

Необхідно пам'ятати, що в підпрограмі використовуються відносні координати, а заміна інструменту та основні команди знаходяться в головній програмі.



Таблиця 13.4

## Керуюча головна програма

Головна програма	Пояснення
%	
O0001	Програма №1023
N10 G90 G40 G80 G49 G98 G21	Рядок безпеки
N20 T1 M06	Виклик свердла Ø3
N30 G43 H1	Компенсація довжини ін.-ту
N40 M03 S1000	Вмикання обертів шпинделя
N50 G00 X10 Y-10	<u>Позиціювання</u> до 1-ої групи
N60 Z0.5	Підведення свердла по Z
N70 M98 P1000	Виклик підпрограми O1000
N70 G00 X40 Y-10	<u>Позиціювання</u> до 2-ої групи
N80 <b>M98 P1000</b>	Виклик підпрограми O1000
N90 G00 X10 Y-40	<u>Позиціювання</u> до 3-ої групи
N100 M98 P1000	Виклик підпрограми O1000
N110 G00 X40 Y-40	<u>Позиціювання</u> до 3-ої групи
N120 M98 P1000	Виклик підпрограми O1000
N130 G91 G28 Z0	Відведення свердла по Z
N140 M05	Зупинка шпинделя
N150 M30	Кінець програми
%	

Таблиця 13.5

## Зовнішня підпрограма

Підпрограма	Пояснення
%	
<b>O1000</b>	Підпрограма O1000
N10 G91	Режим відносних координат
N20 G99 G81 X5 Y0 Z-5 R0.5	Цикл свердління та 1-ий отвір
N30 X-5 Y-5	Координати 2-го отвору
N40 X-5 Y5	Координати 3-го отвору
N50 X5 Y5	Координати 4-го отвору
N60 G80	Відміна циклу свердління
N70 G90	Режим абсолютних координат
N80 <b>M99</b>	Повернення в головну програму
%	

---

Обидві ці програми необхідно передати в систему ЧПК. Оператор повинен викликати програму O0001 і натиснути кнопку «Старт циклу».

Головна програма працює з абсолютними координатами і переміщує інструмент послідовно в центр кожної з чотирьох груп отворів.

Коли система ЧПК зустрічає кадр з M98 P1000, то проходить передача керування підпрограмі з номером O1000.

В цей момент інструмент вже знаходиться в центрі групи отворів.

Система переключається в режим відносного (інкрементального) програмування і використовує постійний цикл свердління.

Після завершення свердління чотирьох отворів однієї групи виключається цикл свердління командою G80 і система ЧПК переходить в режим абсолютних координат G90, щоб правильно виконати позиціонування в головній програмі.

Код M99 передає керування кадру головної програми, наступному за кадром, що викликав цю підпрограму.

Потім інструмент переміщується в центр наступної групи отворів і знову викликається підпрограма O1000.

Це повторюється доки не будуть просвердлені усі отвори і система ЧПК не прочитає код M30 завершення програми.

Якщо б створювалась звичайна програма обробки, то її розмір був би значно більший, оскільки прийшлося би вказувати координати усіх 16 отворів.

Працюючи в цьому форматі простіше виконувати зміни.

Наприклад, якщо зміниться діаметр кола, на якому знаходяться отвори групи, то при роботі з підпрограмою достатньо перерахувати координати центрів чотирьох отворів тільки в підпрограмі.

З головної програми можна викликати декілька різних підпрограм. Більш того, з кожної підпрограми можна викликати декілька інших підпрограм, але системи ЧПК можуть накладати обмеження на кількість підпрограм, тому це треба враховувати при написанні програм.

---

### Запитання до розділу 13

1. Які два види підпрограм використовують при створенні керуючої програми?
2. Які коди використовують для виклику внутрішньої та зовнішньої підпрограм?
3. При знаходженні якого коду в підпрограмі керування повертається до головної керуючої програми?
4. Чи може зовнішня підпрограма виконуватись як окрема програма?
5. Яка система координат використовується в підпрограмі – абсолютна чи відносна?
6. Послідовність створення та реалізації керуючої програми з використанням підпрограм?
7. Чи можна викликати з головної керуючої програми або підпрограм різні інші підпрограми?
8. Які переваги програмування з використанням підпрограм?

---

## Список використаної літератури

1. Аверченков, В.И., Жолобов, А.А., Мрочек, Ж.А., Аверченков, А.В., Терехов, М.В., Левкина, Л.Б. Станки с ЧПУ в машиностроительном производстве [Текст]+[Электронный ресурс]: учеб. пособие для вузов. – Брянск: БГТУ, 2010. – Ч.1. – 216 с.
2. Аверьянов О.И. Модульный принцип построения станков с ЧПУ. – М.: Машиностроение, 1987. – 232с.
3. Андреев, Г. И. Работа на токарных станках с ЧПУ с системой ЧПУ FANUC / Г. И. Андреев. - СПб, 2005. - 42 с.
4. Иванов Ю.М. Модульное проектирование технологии механической обработки деталей на станках с ЧПУ. – Львов: Вища школа. Изд-во при Львов. ун-те, 1982. – 108 с.
5. Ковальов В.А., Гаврушкевич А.Ю., Гаврушкевич Н.В. Посібник для практичного програмування верстатів з ЧПК [Електронний ресурс] – Київ : НТУУ «КПІ ім. Ігоря Сікорського», 2019. – 115с.
6. Ковальов В.А. Гаврушкевич А.Ю., Гаврушкевич Н.В. Програмування в системі Heidenhain TNC 640 [Електронний ресурс] : навчальний посібник – Київ : НТУУ «КПІ ім. Ігоря Сікорського», 2019. – 80с.
7. Кудинов В.А. Динамика станков. - М: Машиностроение, 1967. - 360с.
8. Кузнецов Ю.И., Маслов А.Р., Байков А.Н. Оснастка для станков с ЧПУ: Справочник. – М.: Машиностроение, 1990. –512 с.
9. Кузнецов Ю.И., Сафраган Р.Э., Гончаренко Б.А. Станочные приспособления для металлорежущих станков с ЧПУ. – К: Техніка, 1984. – 156с.
10. Кузнецов Ю.М. Технологічне обладнання з ЧПК: механізми і оснащення: Навч. посібник/Ю.М. Кузнецов, О.Ф. Саленко, О.О.Харченко, В.Т. Щетинін.–Київ-Кременчук-Севастополь: Вид-во «Точка», 2014.–500 с.
11. Кузнецов Ю.М., Придальний Б.І. Приводи затискних механізмів металообробних верстатів. – Луцьк: Вежа-Друк, 2016. – 352с.
12. Ловыгина А.А., Васильев А.В., Кривцов С.Ю. Современный станок с ЧПУ и САД/САМ система. – М: «Эльф ИПР», 2006. – 286с.
13. Маслов А.Р. Конструкции и эксплуатация прогрессивного инструмента. М.: Издательство „ИТО“, 2006. – 166 с.
14. Маслов А.Р. Приспособления для металлообрабатывающего инструмента. / М.: Машиностроение, 2008. – 240 с.
15. Мещерякова В.Б., Стародубов В.С. Металлорежущие станки с ЧПУ: Учеб. пособие. – М.: ИНФРА-М, 2015. – 336 с. – (Высшее образование: Бакалавриат). – [www.dx.doi.org/10.12737/5721](http://www.dx.doi.org/10.12737/5721).

- 
16. Проектирование металлорежущих станков и станочных систем: Справочник- учебник. В 3-х т. Т3. Проектирование станочных систем / А.С.Проников и др. – М.: Изд-во МВТУ им. Н.Э.Баумана: Машиностроение, 2000. – 584 с.
  17. Пуш В.Э. Конструирование металлорежущих станков. - М: Машиностроение, 1977. - 390с.
  18. Решетов Д.Н., Портман В.Т. Точность металлорежущих станков. - М: Машиностроение, 1986. - 336с.
  19. Станки с ЧПУ: устройство, программирование, инструментальное обеспечение и оснастка [Электронный ресурс]: учеб. пособие / А.А. Жолобов, Ж.А. Мрочек, А.В. Аверченков, М.В. Терехов, В.А. Шкаберин. – 2-е изд. – М.: ФЛИНТА, 2014. – 355 с.
  20. Судоплатов И.П. Обработка деталей на станках с ЧПУ. - М: Машиностроение, 1976. - 102с.
  21. Тепинкичиев В.К., и др. Металлорежущие станки. - М: Машиностроение, 1972. - 464с.
  22. Шарин Ю.С. Обработка деталей на станках с ЧПУ. - М: Машиностроение, 1983. - 117с.
  23. Ящерицын П.И., Ефремов В.Д. Металлорежущие станки. – Мн: БГАТУ, 2001 – 446с.
  24. Офіційний сайт групи компаній Fanuc [Електронний ресурс]. - Режим доступу: <http://www.fanuc.com>.
  25. Офіційний сайт ООО "Модмаш-Софт" [Електронний ресурс]. - Режим доступу: <http://www.fms3000.ru>.
  26. Офіційний сайт FABRYKA OBRABIAREK PRECYZYJNYCH AVIA S.A. [Електронний ресурс]. - Режим доступу [www.avia.com.pl](http://www.avia.com.pl).
  27. Офіційний сайт групи компаній HAAS [Електронний ресурс]. - Режим доступу: [www.HaasCNC.com](http://www.HaasCNC.com).
  28. Офіційний сайт групи компаній HEIDENHAIN [Електронний ресурс]. - Режим доступу: [www.heidenhain.de](http://www.heidenhain.de).
  29. Офіційний сайт групи компаній Fanuc [Електронний ресурс]. - Режим доступу: [www.knuth.de](http://www.knuth.de).
  - 30 Офіційний сайт [Електронний ресурс] Режим доступу: <https://www.youtube.com/watch?v=92ztzCP76ho>