

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Механіко-технологічний факультет

Кафедра «Технологія машинобудування»



Методичні вказівки до виконання лабораторних робіт по курсу:

«ПРОЕКТУВАННЯ ТЕХНОЛОГІЧНИХ ПРОЦЕСІВ ДЛЯ ВЕРСТАТІВ З ЧПУ»

для студентів спеціальностей:
133 Галузеве машинобудування
131 Прикладна механіка

Кропивницький

2018

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Механіко-технологічний факультет

Кафедра «Технологія машинобудування»



Методичні вказівки до виконання лабораторних робіт по курсу:

«ПРОЕКТУВАННЯ ТЕХНОЛОГІЧНИХ ПРОЦЕСІВ ДЛЯ ВЕРСТАТІВ З ЧПУ»

для студентів спеціальностей:
133 Галузеве машинобудування
131 Прикладна механіка

Затверджено на засіданні кафедри
«Технологія машинобудування»
Протокол № 6 від 30.11.2018 р.

Кропивницький
2018

Проектування технологічних процесів для верстатів з ЧПУ : метод. вказівки до викон. лаб. робіт студ. спец. 133 «Галузеве машинобудування», 131 «Прикладна механіка» / [уклад. О.І. Скібінський, В.М. Селехова] ; М-во освіти і науки України, Центральноукр. нац. техн. ун-т, каф. технології машинобудування. – Кропивницький : ЦНТУ, 2018. – 66 с.

Укладачі:

Скібінський О.І. – канд.техн.наук, доцент

Селехова В.М.

Рецензент: Кириченко А.М. – д-р.техн.наук, професор

Зміст

Вступ.....	4
Лабораторна робота №1 Геометричні основи: точки на деталі, абсолютне вказання розмірів, інкрементальне вказання розмірів, прискорене переміщення, лінійна подача.....	5
Лабораторна робота №2 Геометричні основи: подача колова, закруглення та фаски, створення контуру.....	14
Лабораторна робота №3 Технологічні основи: швидкість, подача, функції T та M.....	31
Лабораторна робота №4 Постійні цикли. Свердління отворів.....	42
Література.....	52
Додатки.....	53

Вступ

Курс «Проектування технологічних процесів для верстатів з ЧПУ» (ПТПВ ЧПУ) включає цикл лекцій, лабораторних та самостійних занять і спрямований на підвищення кваліфікації майбутнього фахівця.

Курс «ПТПВ ЧПУ» має загальну мету, яка передбачає підготовку студентів до самостійної роботи з проектування технологічних процесів для верстатів з ЧПУ.

Курс «ПТПВ ЧПУ» складається з двох модулів.

Передбачено виконання семи лабораторних робіт. Виконана лабораторна робота захищається студентом у формі опитування.

Опитування з виконаної лабораторної роботи (ВЛР) складається з 5 запитань, які формуються з тих, що вказані у відповідній лабораторній роботі.

Критерії оцінювання відповіді в опитуванні з лабораторної роботи:

- Неправильна відповідь – 0 балів;
- Вірна відповідь – 1 бал.

Критерії оцінювання лабораторної роботи:

- Невиконана лабораторна робота – 0 балів;
- Виконана, але незахищена лабораторна робота – 3 бали;

Загальна сума балів (ЗСБ), яку студент може набрати протягом виконання лабораторних робіт з курсу – 56. Оцінки в балах по одній лабораторній роботі вказані в табл.1, по модулю та по курсу в цілому в табл.2.

Студент вважається атестованим з виконання лабораторної роботи, якщо він набрав кількість балів, що відповідає вказаній в табл.1 в стовбцях «відмінно», «добре» або «задовільно».

Студент вважається не атестованим з виконання лабораторної роботи, якщо він набрав кількість балів, що відповідає вказаній в табл.1 в стовбці «незадовільно».

Таблиця 1

Оцінювання знань при виконанні лабораторної роботи

Показники	Оцінка в балах					
	Зараховано					Не зараховано
	«5» Відмінно	«4» Добре		«3» Задовільно		«2» Незадовільно
	A	B	C	D	E	FX
ВЛР	8	7-6	5-4	3	2	1-0

Студент вважається атестованим з виконання лабораторних робіт з модуля та курсу в цілому, якщо він набрав кількість балів*, що відповідає вказаній в табл.2 в стовбцях «відмінно», «добре» або «задовільно». *Примітка. Кількість балів отриманих студентом при виконанні лабораторних робіт враховується тільки при умові виконання та захисту всіх лабораторних робіт передбачених до виконання.

Студент вважається не атестованим з виконання лабораторних робіт з модуля та курсу в цілому, якщо він набрав кількість балів, що відповідає вказаній в табл.2 в стовбці «незадовільно».

Таблиця 2

Оцінювання знань при виконанні лабораторних робіт з модуля та курсу

Показники	Оцінка в балах					
	Зараховано					Не зараховано
	«5» відмінно	«4» добре		«3» задовільно		«2» Незадовільно
	A	B	C	D	E	FX
ВЛР	56-52	51-42	41-38	37-31	30-22	21-0

До іспиту допускаються тільки ті студенти, які атестовані з виконання лабораторних робіт з курсу.

Лабораторна робота №1

Геометричні основи: точки на деталі, абсолютне вказання розмірів, інкрементальне вказання розмірів, прискорене переміщення, лінійна подача

Мета роботи: ознайомитися з геометричними основами для токарної обробки на верстатах ф. Haas.

Теоретичні відомості

Токарні верстати ф. Haas.

Токарний верстат Haas TL-1 (рис. 1.1) серії Toolroom володіє широкими функціональними можливостями, від ручного режиму управління до повнофункціонального ЧПУ, і оснащений запатентованою діалоговою операційною системою, при використанні якої обробка деталей і створення програм обробки практично не вимагають зусиль.



Рис. 1.1. Токарний верстат Haas TL-1

Оператор просто вибирає необхідну операцію, дотикається до поверхонь по осі X і Z, а потім вводить базову розмірну інформацію. Значення подачі на оберт, швидкості шпинделя і глибини різання вводяться системою управління автоматично, але оператор може змінити ці значення. Коли вся необхідна інформація введена, потрібна операція виконується при натисненні клавіші «Cycle Start» (Запуск циклу). Функція запису дозволяє зберігати інформацію для подальшого використання.

Верстат TL-1 працює в чотирьох режимах. У всіх режимах система управління Haas забезпечує високоточне цифрове зчитування позиції (DRO) з точністю відтворення до 0,013 мм при використанні ручних маховичків і до 0,003 мм при використанні електронних маховичків.

У ручному режимі переміщення по осях X і Z здійснюються за допомогою ручних маховичків. У напівавтоматичному режимі TL-1 проводить одночасну лінійну інтерполяцію по

обох осях при використанні одного ручного маховичка. У напівавтоматичному режимі легко виконувати такі операції, як зняття внутрішніх і зовнішніх фасок та точіння конусів. У автоматичному режимі передбачені вбудовані цикли обробки (як для однопрохідних, так і для багатопрохідних операцій) для виконання операцій чорнового і чистового профілювання, зняття фасок, проточування канавок, відрізання, нарізування різьб різцем, свердління і нарізування різьб мітчиком. У повнофункціональному режимі ЧПУ верстат TL-1 програмують за допомогою стандартного G-коду, а управління рухом по всіх осях здійснюється системою управління Haas через G-кодову програму.

Операції, які важко або неможливо було виконати на верстатах з ручним управлінням (радіуси, конуси, профілі, канавки, нарізання внутрішніх і зовнішніх різьб різцем або мітчиком), тепер легко здійсними на верстаті TL-1 – без необхідності програмування за допомогою G-коду. Меню допомоги доступне безпосередньо з екрану, а функція графічної імітації дозволяє операторам перевірити свою роботу до початку обробки деталі.

Компанія Haas Automation випускає повну гамму токарних верстатів з ЧПУ, призначених для задоволення потреб сучасних механічних цехів. Серія SL забезпечує широкий вибір можливостей, а компактна опція Big Bore додатково дозволяє розширити можливості при збереженні початкової займаної площі. Максимальні розміри оброблюваного виробу для моделі SL-20 (рис. 1.2) складають 262 x 508 мм, патрон 210 мм, максимальний діаметр прутка 51 мм.



Рис. 1.2. Токарний (багатоцільовий) верстат Haas SL-20

Високопродуктивні багатоцільові токарні верстати Haas мають масивну литу передню бабку з симетричними ребрами для додання жорсткості і термостабільності; миттєве перемикання схеми з'єднання обмоток «зірка-трикутник», що забезпечує максимальну продуктивність у всьому діапазоні частоти обертання; вбудовані піддони для стружки і високопродуктивні системи подачі ЗОР для ефективного видалення стружки. У системі

управління Haas передбачені розширені функції управління інструментом, функції управління натисненням однієї кнопки, кольоровий 15-дюймовий РК-дисплей і порт USB. Компанія Haas піднімає токарну обробку з ЧПУ на новий рівень надійності, простоти і продуктивності.

На рис. 1.3 показана загальна структура токарних верстатів Haas.

Числове програмне управління.

Числове програмне управління (ЧПУ) означає комп'ютеризовану систему управління, яка прочитує інструкції спеціалізованої мови програмування (наприклад, G-код) і яка управляє приводами верстатів і верстатним оснащенням.

Верстати, обладнані числовим програмним управлінням, називаються верстатами з ЧПУ.

Інтерпретатор системи ЧПУ проводить переклад програми з вхідної мови в команди управління головним приводом, приводами подач, контролерами управління вузлів верстата (включити/виключити охолодження, наприклад). Для визначення необхідної траєкторії руху робочого органу в цілому (інструменту/заготовки) відповідно до управляючої програми (УП), використовується інтерполятор, що розраховує положення проміжних точок траєкторії по заданим в програмі кінцевим.

Абревіатура ЧПУ відповідає двом англомовним — NC і CNC, — що відображає еволюцію розвитку систем управління устаткуванням.

Системи типу NC (англ. Numerical control) передбачали використання жорстко заданих схем управління обробкою — наприклад, завдання програми за допомогою штекерів або перемикачів, зберігання програм на зовнішніх носіях (магнітні стрічки, перфоровані паперові стрічки). Яких-небудь пристроїв оперативного зберігання даних, керуючих мікропроцесорів не передбачалося. Системи ЧПУ, описувані як «CNC» (англ. Computer numerical control), засновані на мікропроцесорі з оперативною пам'яттю, з операційною системою, приводи управляються власними мікроконтролерами.

Програма для устаткування з ЧПУ може бути завантажена із зовнішніх носіїв, наприклад, дискет або із звичайних або спеціалізованих флеш-накопичувачів. Крім цього, сучасне устаткування підключається до заводських мереж зв'язку.

Найбільш поширена мова програмування ЧПУ для металорізального устаткування описана документом ISO 6983 Міжнародного комітету по стандартам і називається «G-код». В окремих випадках — наприклад, системи управління верстатами гравіювань — мова управління принципово відрізняється від стандарту. Для простих завдань, наприклад, розкрою плоских заготовок, система ЧПУ в якості вхідної інформації може використовувати текстовий файл у форматі обміну даними — наприклад DXF або HP-GL.

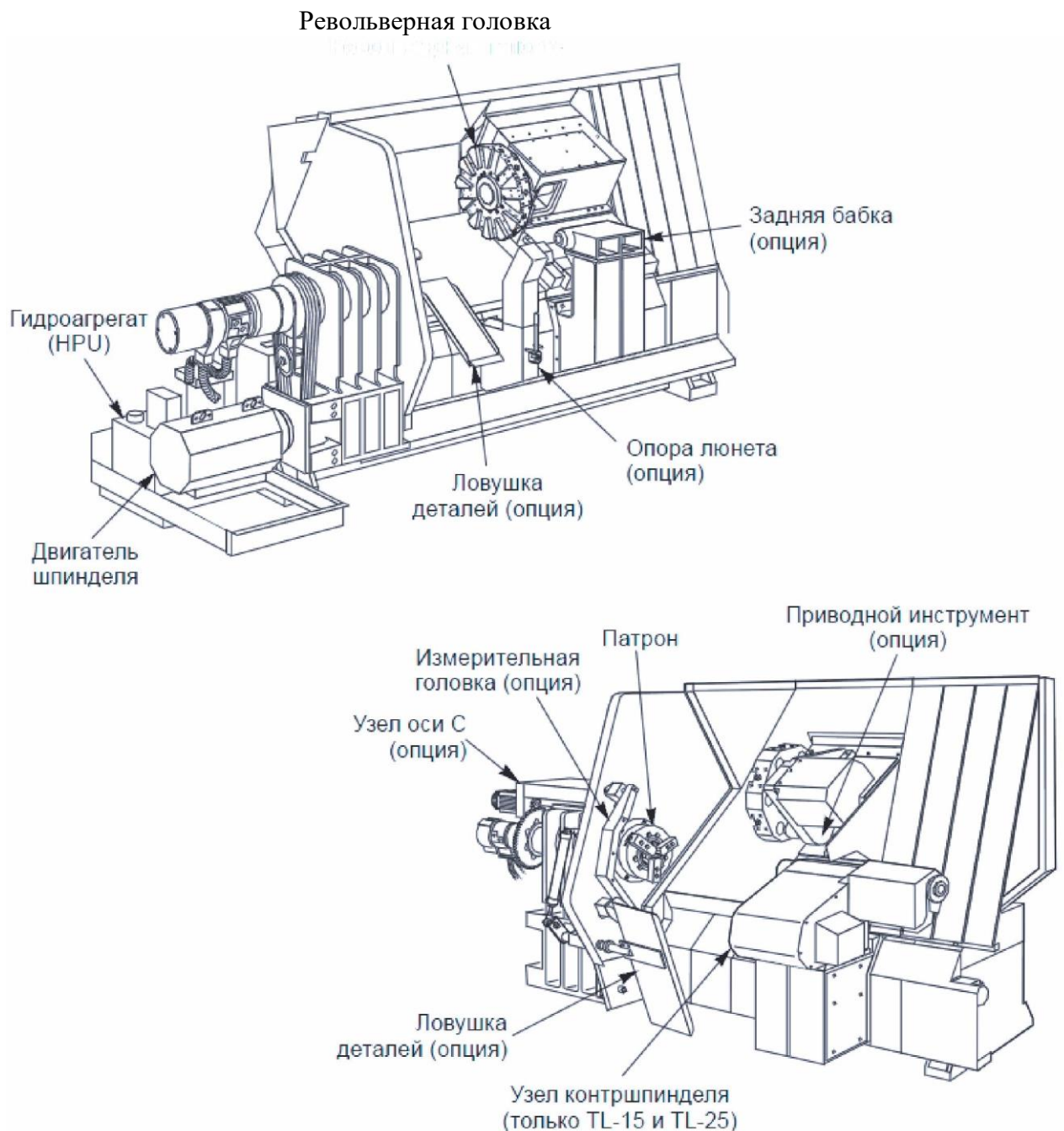


Рис. 1.3. Структура токарных верстатів Haas

Структура управляючої програми.

Кожна УП представляє собою послідовність пронумерованих речень, які називаються кадрами. Номер кадру – це мітка, по якій можна знайти необхідний кадр, з метою його редагування, або розпочати виконання УП із даного кадру. При створенні УП в кадрах записується тільки та інформація, яка змінюється по відношенню до попередньої частини програми.

Кадр складається із слів. Кожне слово має адресу (одну із букв латинського алфавіту) і десяткове число.



Слова в кадрах УП можна записувати в довільній послідовності, пристрій ЧПУ відпрацює спочатку команди технологічних функцій S, F, T, M а потім підготовчих G, з виконанням розмірних переміщень.

Функції з адресою G, які називаються підготовчими, визначають режим і умови роботи верстата з ЧПУ. Вони кодуються від G00 до G99.

Порядок виконання роботи

1. Геометричні основи.

1.1 Точки на деталі.

1.1.1 Верстат з плоскою станиною (серія TL)

1.1.1		<p>X – це завжди діаметр!</p> <p>Задайте значення для X:</p> <p>X</p> <p>P1 <input type="text"/></p> <p>P2 <input type="text"/></p> <p>P3 <input type="text"/></p>
		<p>Z – це завжди відстань до торцьової поверхні (із знаком « - »)!</p> <p>Задайте значення для Z:</p> <p>Z</p> <p>P4 <input type="text"/></p> <p>P5 <input type="text"/></p> <p>P6 <input type="text"/></p>

Задайте координати точок:

	X	Z
P7	<input type="text"/>	<input type="text"/>
P8	<input type="text"/>	<input type="text"/>

1.1.2 Верстат з нахиленою станиною (серія SL)

1.1.2

X – це завжди діаметр!

Задайте значення для X:

	X
P1	<input type="text"/>
P2	<input type="text"/>
P3	<input type="text"/>

Z – це завжди відстань до торцьової поверхні (із знаком « - »)!

Задайте значення для Z:

	Z
P4	<input type="text"/>
P5	<input type="text"/>
P6	<input type="text"/>

Задайте координати точок:

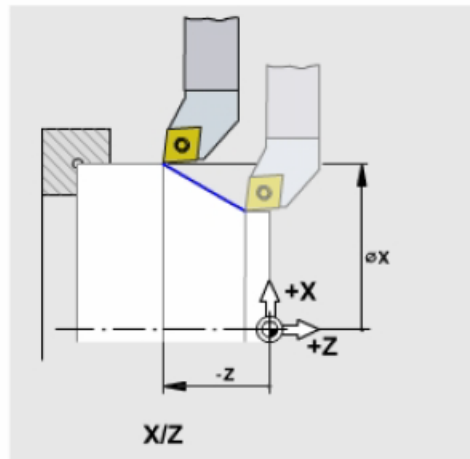
	X	Z
P7	<input type="text"/>	<input type="text"/>
P8	<input type="text"/>	<input type="text"/>

1.2 Абсолютне та інкрементальне завдання розмірів.

Абсолютне

1.2

X / Z Абсолютне вказання розмірів
(...переміщення ДО ...)

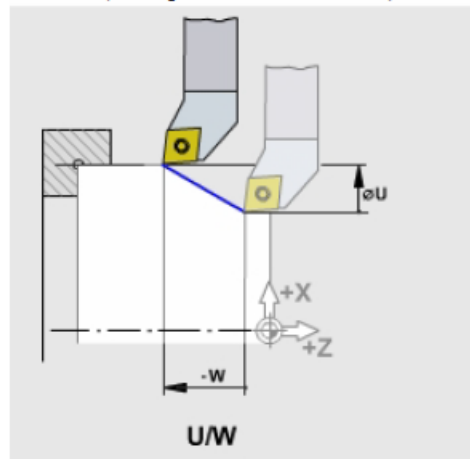


При вказанні **абсолютних** розмірів ці значення завжди відносяться до **нульової точки деталі**.

Інкрементальне

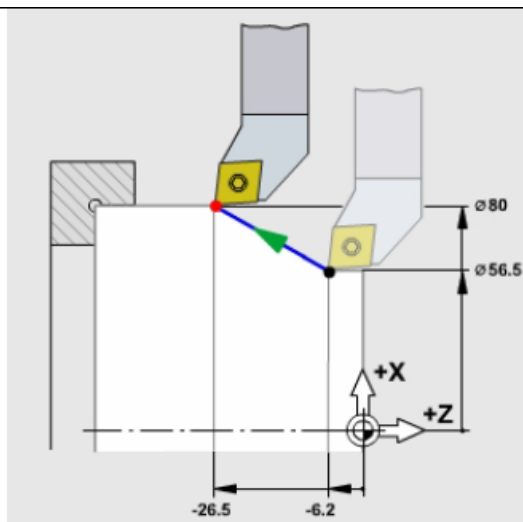
1.2

U / W Інкрементальне вказання розмірів
(...переміщення НА ...)



При вказанні **інкрементальних** розмірів завжди потрібно вказувати **різницю значень** між **початковою точкою** і **кінцевою точкою** з врахуванням напрямку.

U відноситься до **діаметру!**



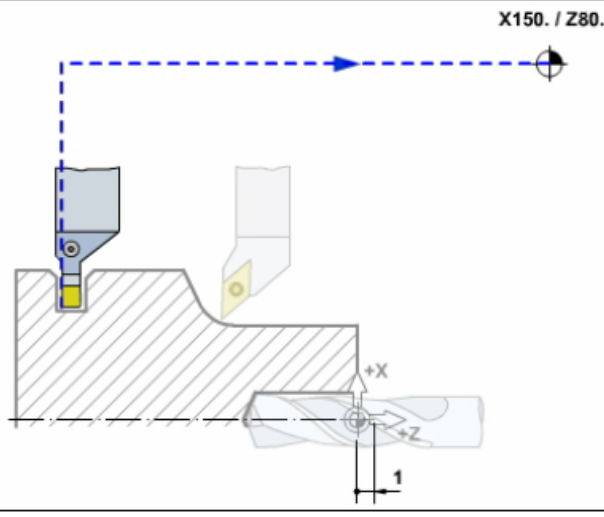
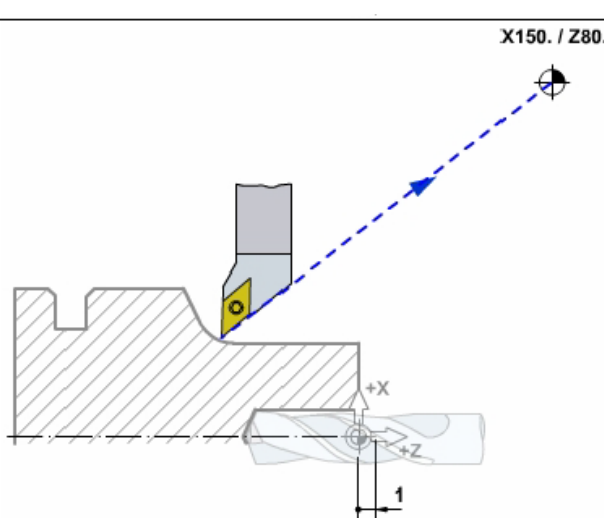
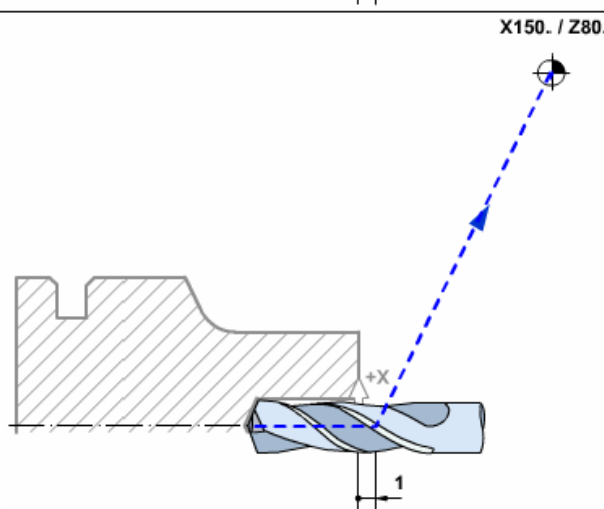
Задайте координати кінцевої точки в **абсолютних** значеннях:

X	<input type="text"/>
Z	<input type="text"/>

1.3 G00 або G0 Прискорений хід.

За допомогою **G00** виконується позиціонування інструменту **на прискореному ході**.

Запрограмуйте три шляхи інструменту:

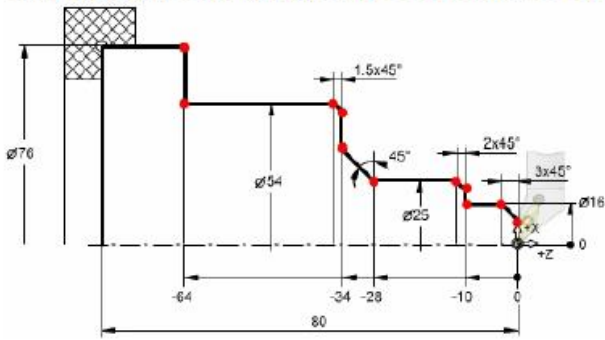
1.3		<p>T6</p> <table border="1" data-bbox="1101 224 1340 336"><tr><td>G</td><td> </td><td>X</td><td></td></tr><tr><td>G</td><td></td><td>Z</td><td></td></tr></table>	G		X		G		Z			
G		X										
G		Z										
		<p>T2</p> <table border="1" data-bbox="1037 761 1404 817"><tr><td>G</td><td> </td><td>X</td><td></td><td>Z</td><td></td></tr></table> <p>Обидві вісі переміщуються з максимальною швидкістю. В залежності від швидкості і шляху вісі досягають своїх цільових координат не одночасно.</p>	G		X		Z					
G		X		Z								
		<p>T12</p> <table border="1" data-bbox="1037 1265 1404 1377"><tr><td>G</td><td> </td><td>Z</td><td></td></tr><tr><td>G</td><td></td><td>X</td><td></td><td>Z</td><td></td></tr></table>	G		Z		G		X		Z	
G		Z										
G		X		Z								

1.4 G01 або G1 Подача лінійна.

1.4.1 Запрограмуйте шляхи переміщення з вказанням **абсолютних** розмірів

За допомогою **G1** виконується **лінійне позиціонування** інструмента при **подачі**.

1.4.1 Запрограмуйте контур (початкова точка X0 Z0).



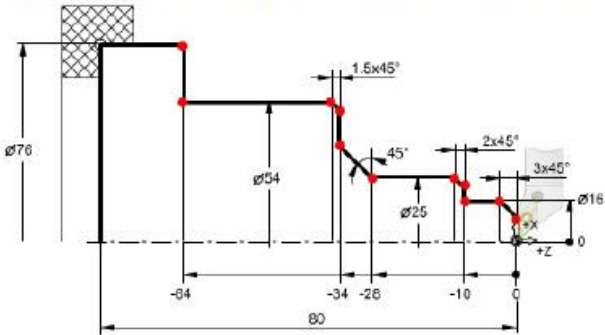
Абсолютне вказання розмірів

N	G	X	Z
N1	G1		
N2	G1		
N3			
N4			
N5			
N6			
N7			
N8			
N9			
N10			
N11			

1.4.2 Запрограмуйте шляхи переміщення з вказанням **інкрементальних** розмірів

При вказанні **інкрементальних** розмірів завжди необхідно вказувати **різницю значень** між **початковою точкою** і **кінцевою точкою** з врахуванням **напрямку**.

1.4.2 Запрограмуйте контур (початкова точка X0 Z0).



Інкрементальне вказання розмірів

N	G	U	W
N1	G1		
N2	G1		
N3			
N4			
N5			
N6			
N7			
N8			
N9			
N10			
N11			

Контрольні запитання:

1. Як здійснюється зміна інструменту на токарних верстатах з ЧПУ та токарних багатоцільових верстатах з ЧПУ?
2. Що таке контршпindel, для чого він призначений, в яких моделях верстатів із представлених, використовується контршпindel (протишпindel)?
3. Що розуміють під аббревіатурою ЧПУ? Яке призначення інтерпретатора системи ЧПУ?
4. Яка мова використовується для програмування ЧПУ?
5. Яка структура управляючої програми?
6. Абсолютне та інкрементальне завдання розмірів. Координати X, Z, U, W.
7. Призначення функції G00.
8. Призначення функції G01.

Лабораторна робота №2

Геометричні основи: подача колова, закруглення та фаски, створення контуру

Мета роботи: ознайомитися з геометричними основами для токарної обробки на верстатах ф. Наас.

Теоретичні відомості

Поняття інтерполяції.

Якщо, наприклад, необхідно перемістити інструмент вздовж лінійної вісі X вправо, на 1 мм, то необхідно записати в управляючу програму, рядок X1 (передбачається абсолютна система кодування). Як наслідок, верстат переміщатиме інструмент строго вздовж однієї прямої.

Якщо необхідно сумістити рух, розглянутий вище, з переміщенням уздовж вісі Z на 1 мм вгору, то в цьому випадку інструмент рухатиметься по похилій траєкторії. Якщо ж таке переміщення супроводжується ще і зняттям металу, то необхідно додатково призначити швидкість переміщення в кінцеву точку (швидкість подачі F). Для реалізації такого руху використовується лінійна інтерполяція.

Виконуючи лінійну інтерполяцію, система з ЧПУ буде автоматично і дуже точно обчислювати серію вельми малих переміщень уздовж однієї з лінійних осей, переміщаючи інструмент якомога ближче до запрограмованого лінійного переміщення. На сучасному устаткуванні з ЧПУ така траєкторія буде здаватися саме лінійним рухом (на відміну від реально ступінчастого рис. 2.1).

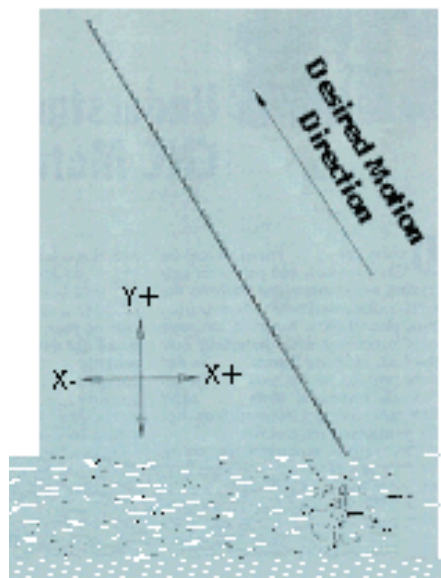


Рис. 2.1. Лінійна інтерполяція (безліч дрібних переміщень поперемінно по одній з лінійних осей)

Крок інтерполяції для різних верстатів різний, зазвичай він дорівнює 0,001 мм. Величину кроку інтерполяції називають дискретою.

Програміст-технолог розраховує також на те, що на верстаті можна переміщати інструмент по колу. Такий тип інтерполяції називається коловою інтерполяцією. Як і у випадку лінійної інтерполяції, система з ЧПУ повинна найкращим чином згенерувати траєкторію руху, максимально наближену до кола (рис. 2.2). Переміщення з лінійною та коловою інтерполяціями є дискретними.

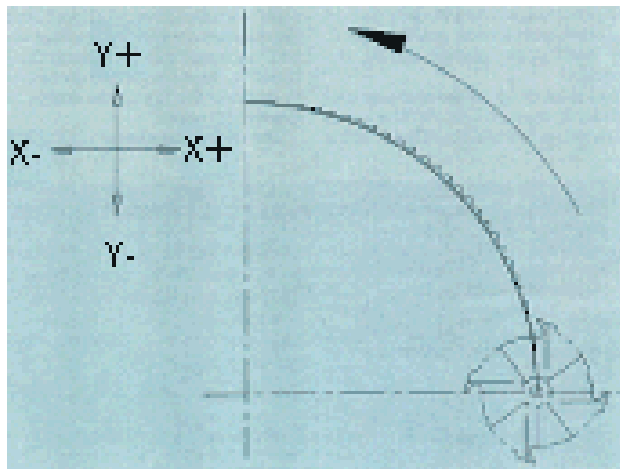


Рис. 2.2. Колова інтерполяція

Залежно від типу конкретної обробки можна застосовувати і інші типи інтерполяції. Наприклад, при фрезеруванні деталей типу шнеків верстат повинен виконувати колову інтерполяцію (переміщення уздовж осей X і Y) і в той же час лінійну (рухатися уздовж вісі Z). Такий рух називається фрезеруванням по спіралі, а тип інтерполяції - спіральним. Багато систем з ЧПУ мають вбудовану спіральну інтерполяцію.

Зовсім інший тип інтерполяції потрібний на токарному устаткуванні, яке має поворотний різцетримач і вісь C для обертання самої деталі! Тут застосовується полярна інтерполяція для програмування руху різця в процесі зняття металу уздовж деякого складного контуру. Полярна координатна інтерполяція в цьому випадку як би «замінює» складніші обчислення, прискорюючи процес розрахунку.

Колова інтерполяція задається за допомогою функцій G02/ G03 (circular/helical interpolation).

Функції G02 і G03 використовуються для переміщення інструмента по коловій траєкторії (дузі), на швидкості подачі, заданій за допомогою F.

G02 (clockwise) – колова інтерполяція по годинниковій стрілці CW.

G03 (counterclockwise) – колова інтерполяція проти годинникової стрілки CCW.

Існує два способи для формування кадру колової інтерполяції:

- вказання центра кола за допомогою I,J,K;
- вказання радіуса кола за допомогою R.

Більшість сучасних верстатів з ЧПУ підтримує обидва варіанта запису.

Приклад:

N50 G03 X0. Y-17. I0. J17.

Приклад:

N50 G03 X0. Y-17. R 17

F – Функція швидкості подачі (Feed rate definition)

Функція швидкості подачі використовує адресу F, за якою слідує число, що вказує на швидкість подачі при обробці.

Встановлена швидкість подачі залишається незмінною, до тих пір, поки не буде указано нове числове значення разом з F або не буде змінено режим переміщень за допомогою G00.

N45 G01 Z-1 F40 – переміщення на глибину 1 мм на подачі 40 мм/хв.

N50 G01 X12 Y22 – переміщення інструмента (40 мм/хв.)

N55 G01 Y50 – переміщення інструмента (40 мм/хв.)

N60 G01 Y50 F22 – переміщення інструмента (22 мм/хв.)

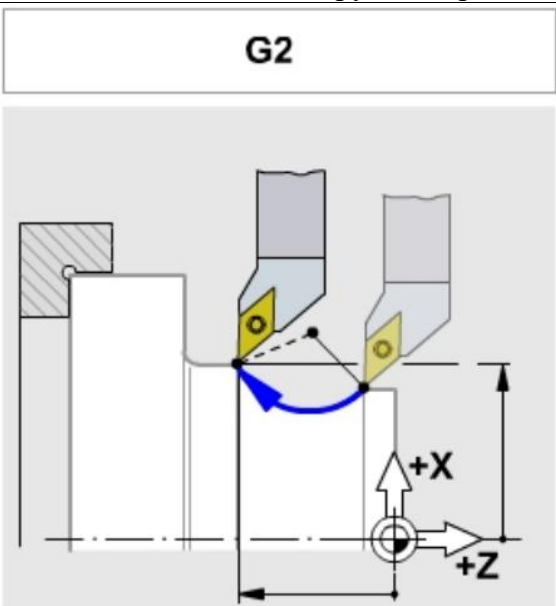
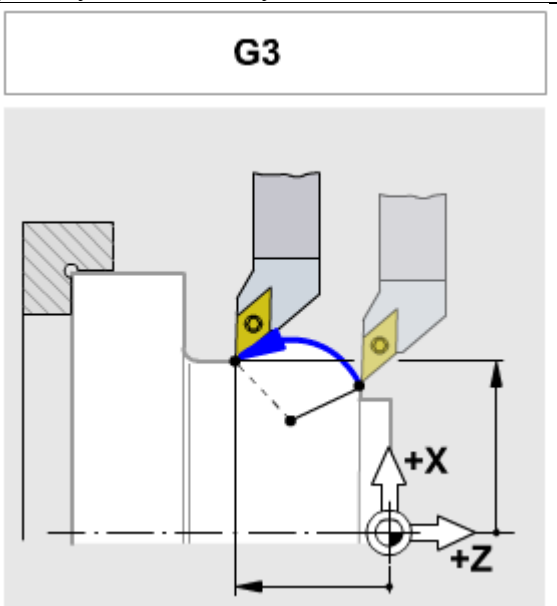
N65 G01 X30 Y120 – переміщення інструмента (22 мм/хв.)

N70 G00 Z5 – швидке переміщення по Z

N75 X00 Y00 – швидке переміщення.

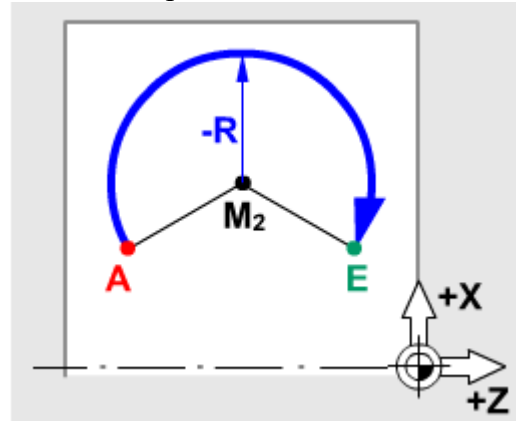
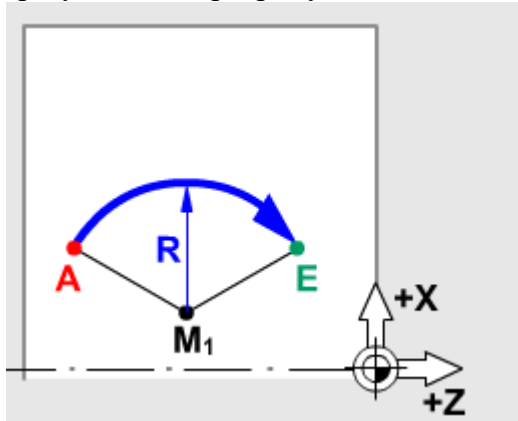
G-коди можуть бути **модальними** і **немодальними**. Команди з **модальним** G-кодом, діють до кінця програми або до тих пір, поки не буде задано інший G-код тієї ж групи. **Немодальний** G-код діє тільки в межах строчки, на наступну строчку немодальний G-код із попередніх строчок не діє. Немодальними є коди групи 00, інші групи кодів модальні.

Порядок виконання роботи

1. Геометричні основи.	
1.5 Подача колова.	
Для G2/G3 інструмент при подачі переміщується по колу	
G2	G3
	
Дуга По годинниковій стрілці	Дуга Проти годинникової стрілки
Програмування з радіусом R значно простіше, ніж з координатами середньої точки I та	

К. Але необхідно знати:

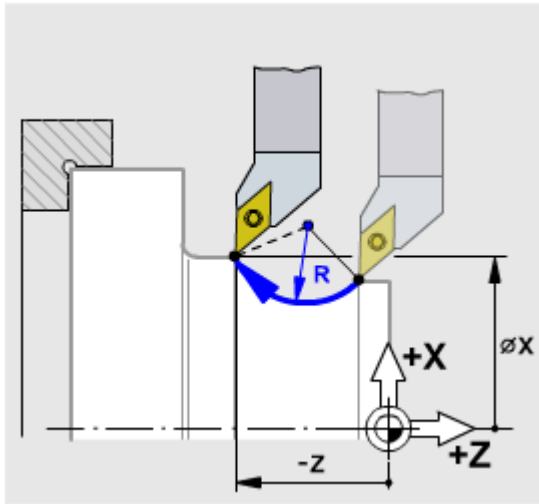
- програмування повного кола за допомогою R не можливо.
- результатом програмування з R завжди є подвійне рішення.



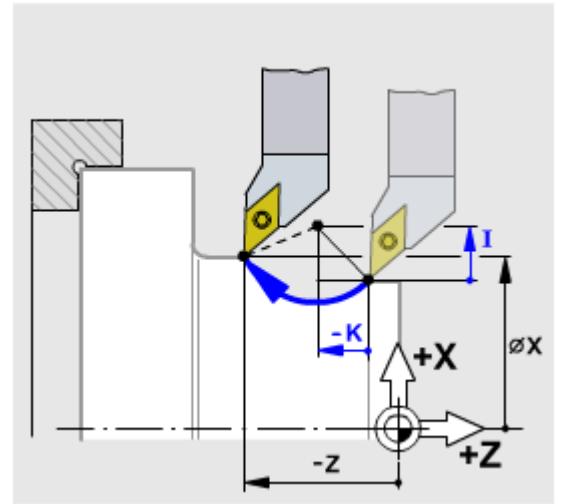
Для лівого випадку програмується, наприклад, R10, а для правого випадку, наприклад, R-10.

При програмуванні дуг кіл є два варіанти:

G2 [G3] X... Z... R...
Кінцева Радіус
точка

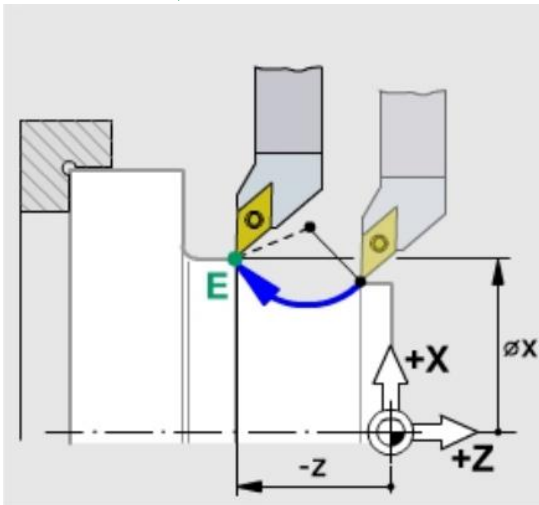


G2 [G3] X... Z... I... K...
Кінцева Середня
точка точка



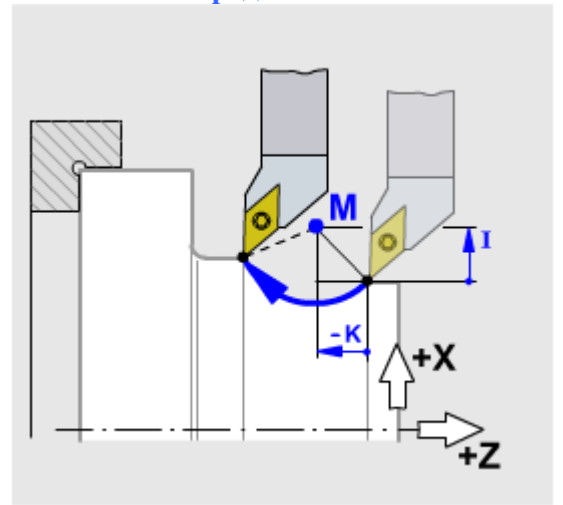
Програмування дуг кіл з X/Z і I/K

G2 [G3] X... Z...
Кінцева точка

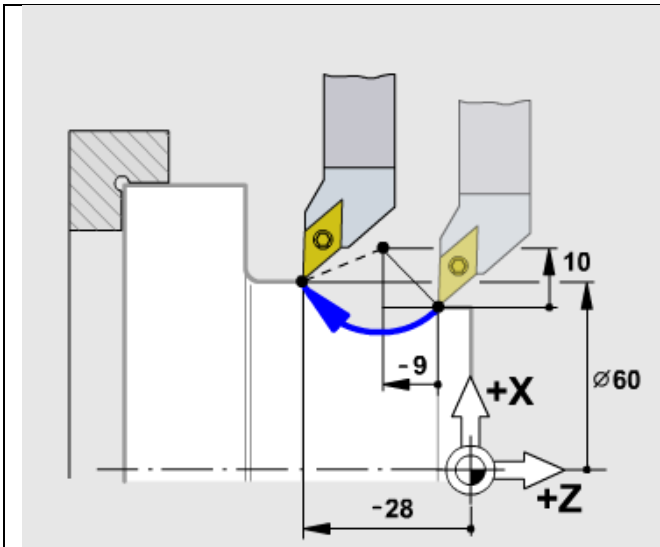


Кінцева точка програмується в абсолютних координатах

G2 [G3] I... K...
Середня точка



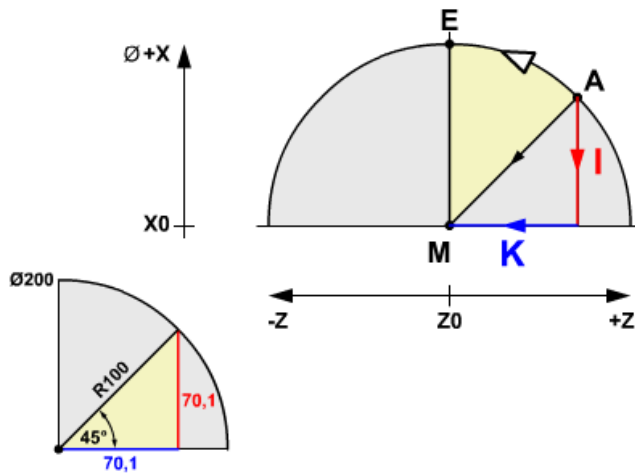
Середня точка програмується в інкрементальних значеннях відносно початкової точки



- значення $X=60$ тому, що кінцева точка дуги лежить на діаметрі 60 мм;
- значення $Z=-28$ тому, що кінцева точка лежить на відстані 28 мм від нульової точки деталі у від'ємному напрямку;
- значення $K=-9$ тому, що відстань між початковою і середньою точками складає 9 мм і тому, що напрямок погляду (від початкової точки до середньої) протилежний додатному напрямку осі Z ;
- значення $I=10$ тому, що відстань між початковою і середньою точками складає 10 мм і тому, що напрямок погляду (від початкової точки до середньої) співпадає з додатним напрямком осі X .

1.5.1

I відноситься до **X** **K** до **Z**



При визначенні знаку перед **I** і **K** завжди діє правило:
Дивитися потрібно від початкової точки до середньої!

Виберіть правильні відповіді:

+	0	-	2	3
70.1	100	140.2	200	

G

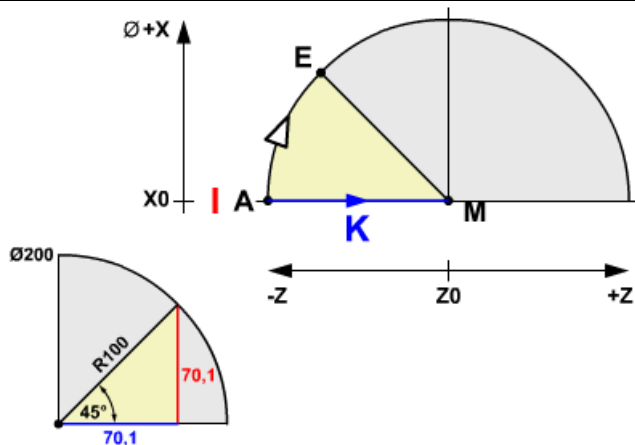
X

Z

I

K

1.5.2



Виберіть правильні відповіді:

+	0	-	2	3
70.1	100	140.2	200	

G

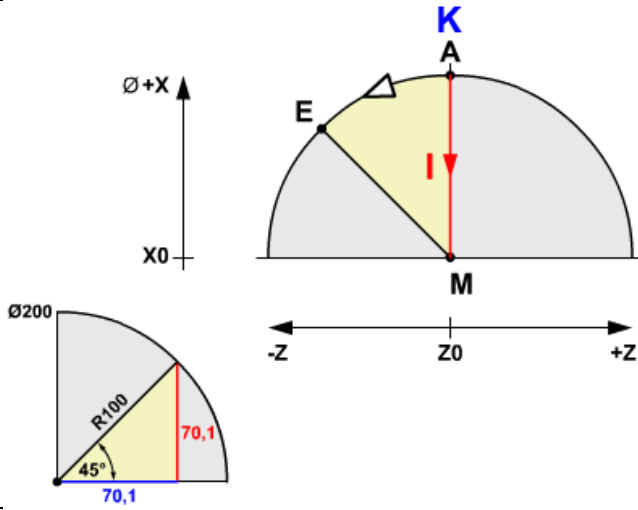
X

Z

I

K

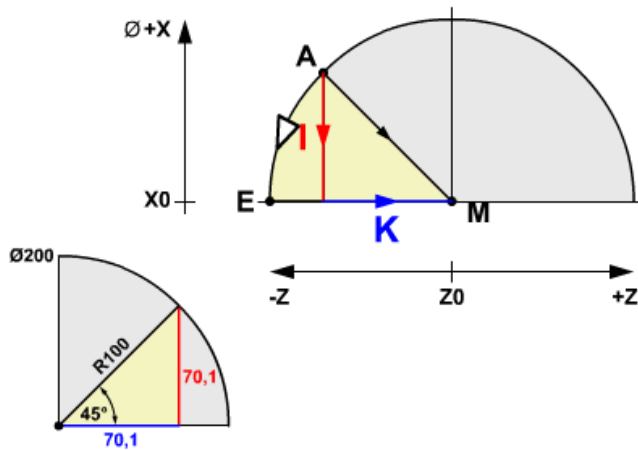
1.5.3



Виберіть правильні відповіді:

+ 0 - 2 3
 70.1 100 140.2 200
 G
 X
 Z
 I
 K

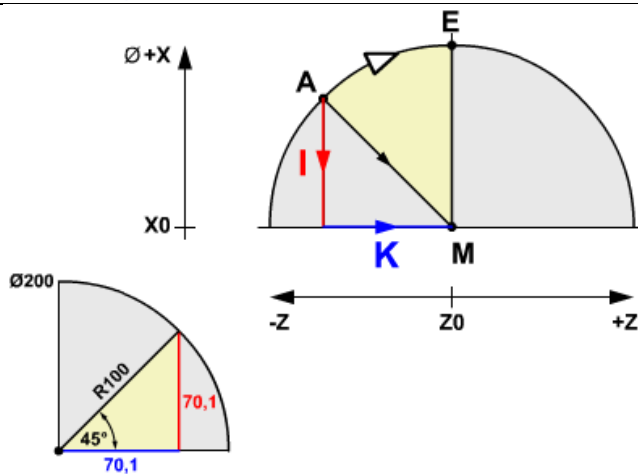
1.5.4



Виберіть правильні відповіді:

+ 0 - 2 3
 70.1 100 140.2 200
 G
 X
 Z
 I
 K

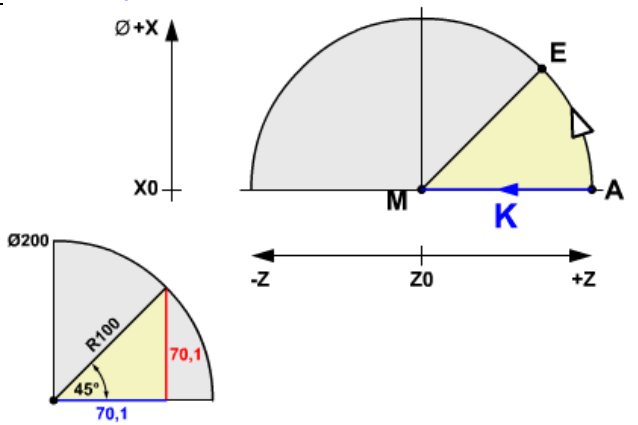
1.5.5



Виберіть правильні відповіді:

+ 0 - 2 3
 70.1 100 140.2 200
 G
 X
 Z
 I
 K

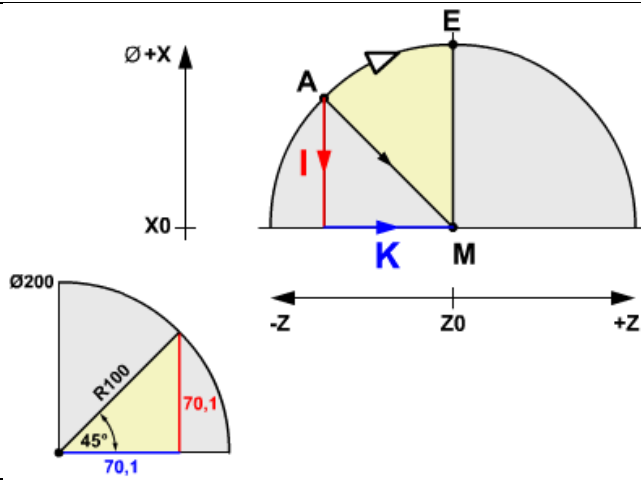
1.5.6



Виберіть правильні відповіді:

+ 0 - 2 3
 70.1 100 140.2 200
 G
 X
 Z
 I
 K

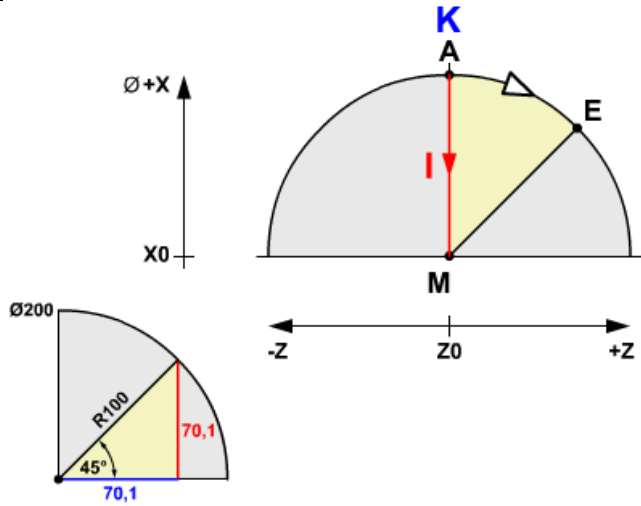
1.5.7



Виберіть правильні відповіді:

+ 0 - 2 3
 70.1 100 140.2 200
 G
 X
 Z
 I
 K

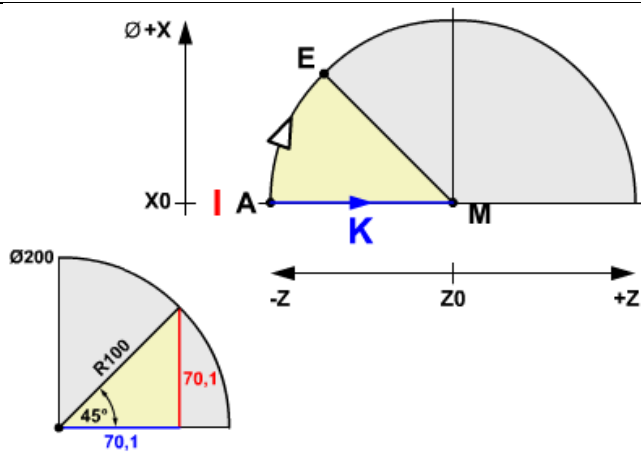
1.5.8



Виберіть правильні відповіді:

+ 0 - 2 3
 70.1 100 140.2 200
 G
 X
 Z
 I
 K

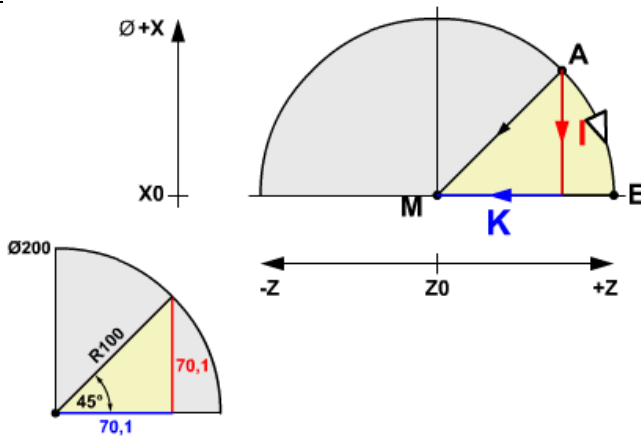
1.5.9



Виберіть правильні відповіді:

+ 0 - 2 3
 70.1 100 140.2 200
 G
 X
 Z
 I
 K

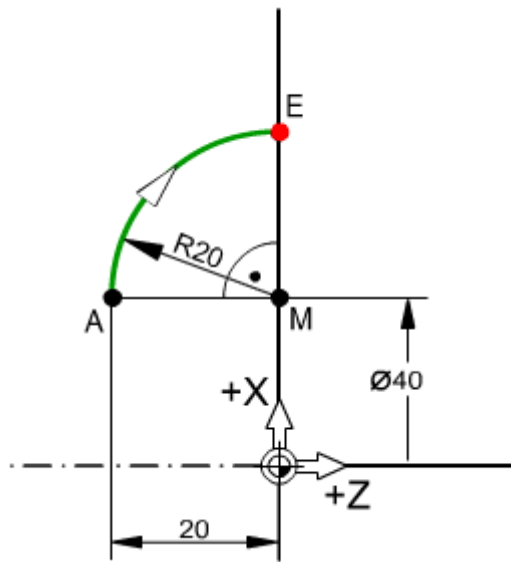
1.5.10



Виберіть правильні відповіді:

+ 0 - 2 3
 70.1 100 140.2 200
 G
 X
 Z
 I
 K

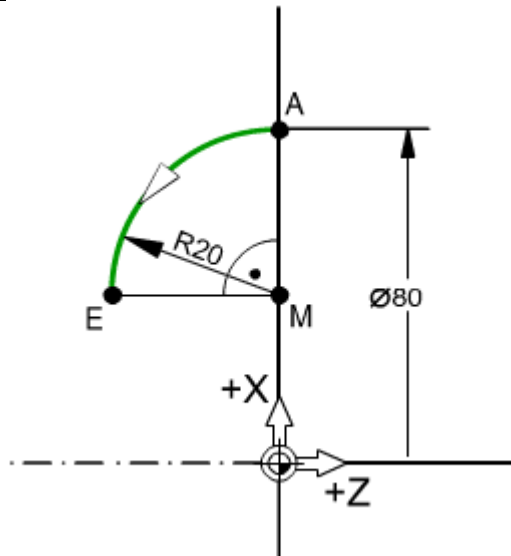
1.5.11



Запишіть правильні значення:

G	<input type="text" value=""/>
X	<input type="text" value=""/>
Z	<input type="text" value=""/>
I	<input type="text" value=""/>
K	<input type="text" value=""/>

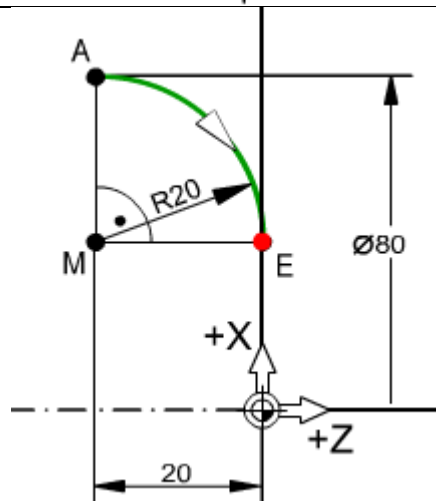
1.5.12



Запишіть правильні значення:

G	<input type="text" value=""/>
X	<input type="text" value=""/>
Z	<input type="text" value=""/>
I	<input type="text" value=""/>
K	<input type="text" value=""/>

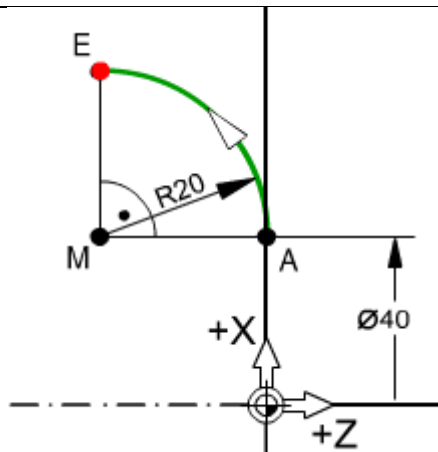
1.5.13



Запишіть правильні значення:

G	<input type="text" value=""/>
X	<input type="text" value=""/>
Z	<input type="text" value=""/>
I	<input type="text" value=""/>
K	<input type="text" value=""/>

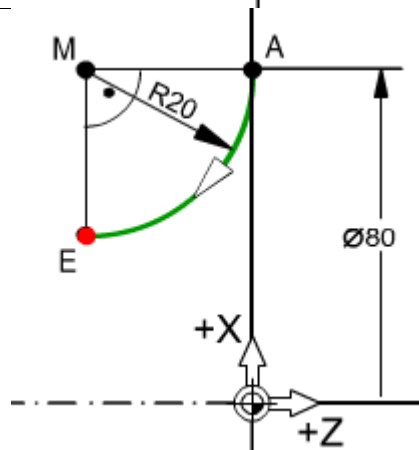
1.5.14



Запишіть правильні значення:

G	<input type="text"/>
X	<input type="text"/>
Z	<input type="text"/>
I	<input type="text"/>
K	<input type="text"/>

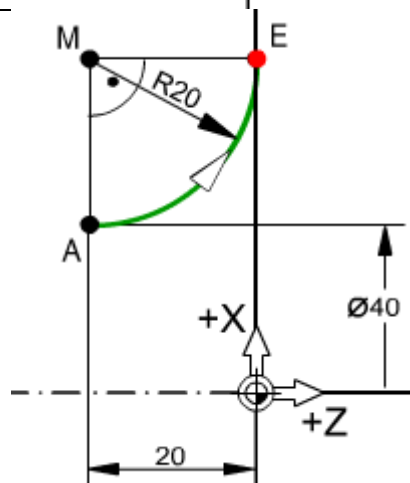
1.5.15



Запишіть правильні значення:

G	<input type="text"/>
X	<input type="text"/>
Z	<input type="text"/>
I	<input type="text"/>
K	<input type="text"/>

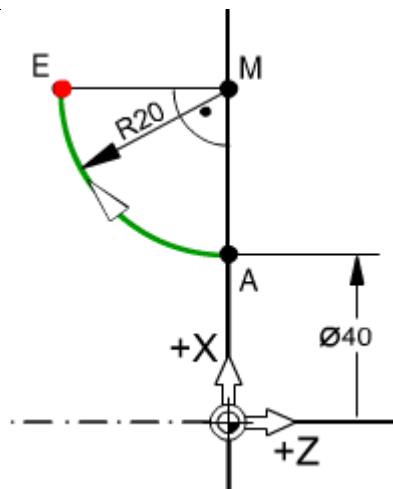
1.5.16



Запишіть правильні значення:

G	<input type="text"/>
X	<input type="text"/>
Z	<input type="text"/>
I	<input type="text"/>
K	<input type="text"/>

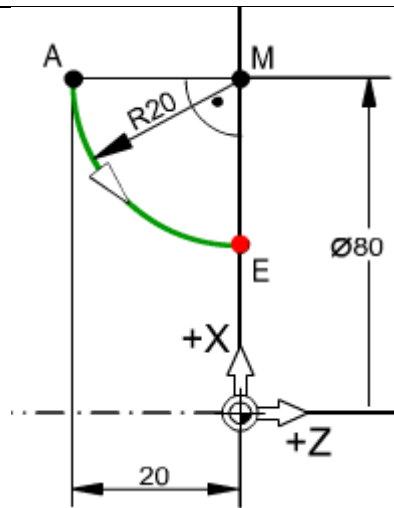
1.5.17



Запишіть правильні значення:

G	<input type="text"/>
X	<input type="text"/>
Z	<input type="text"/>
I	<input type="text"/>
K	<input type="text"/>

1.5.18

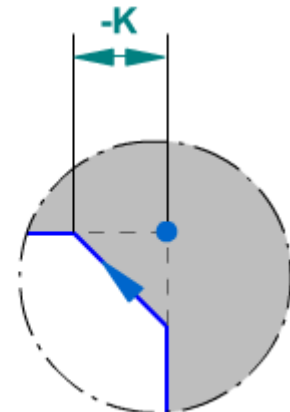
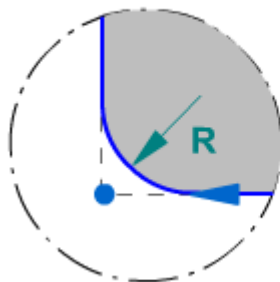
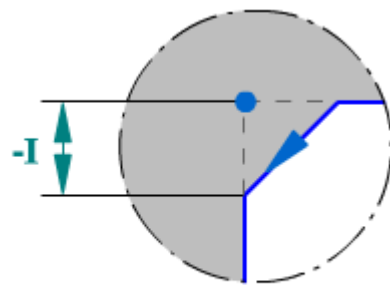
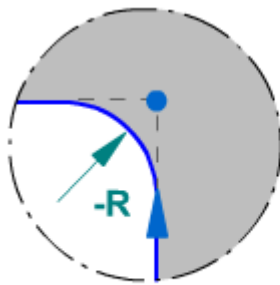


Запишіть правильні значення:

G	<input type="text"/>
X	<input type="text"/>
Z	<input type="text"/>
I	<input type="text"/>
K	<input type="text"/>

1.6 Закруглення і фаски.

Закруглення і фаски можна виконувати там, де сходяться елементи з паралельними осями.



Знак **мінуса** необхідно поставити в тому випадку, якщо закруглення або фаска проходить у **від'ємному напрямку** вісі X або вісі Z.

I використовується на горизонтальних відрізках.

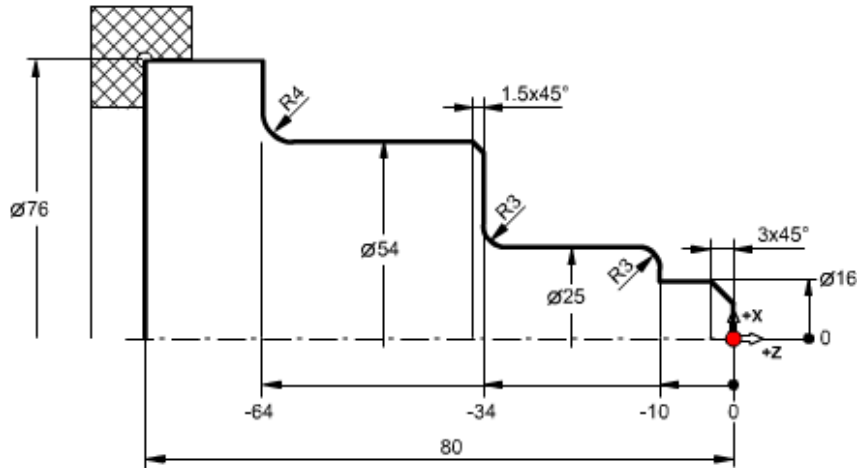
K використовується на вертикальних відрізках.

Створення контуру.

Створення контуру з R і K.

1.6.1

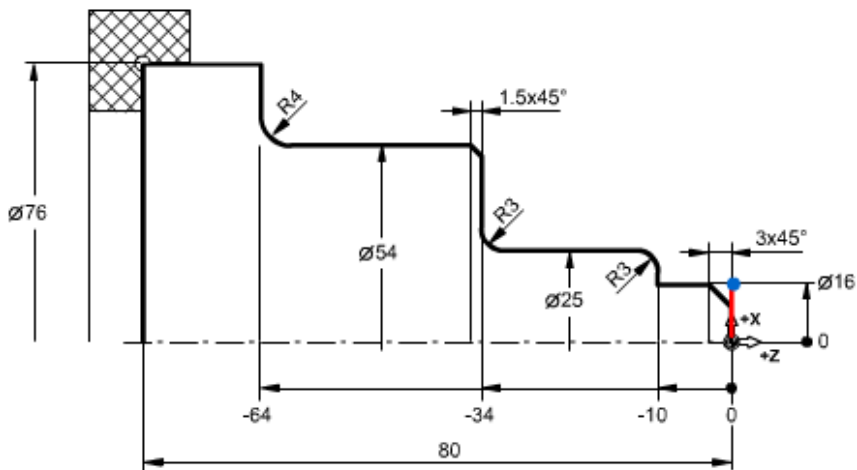
Складіть зазначений контур. Введіть спочатку початкову точку.



X	<input type="text"/>
Z	<input type="text"/>

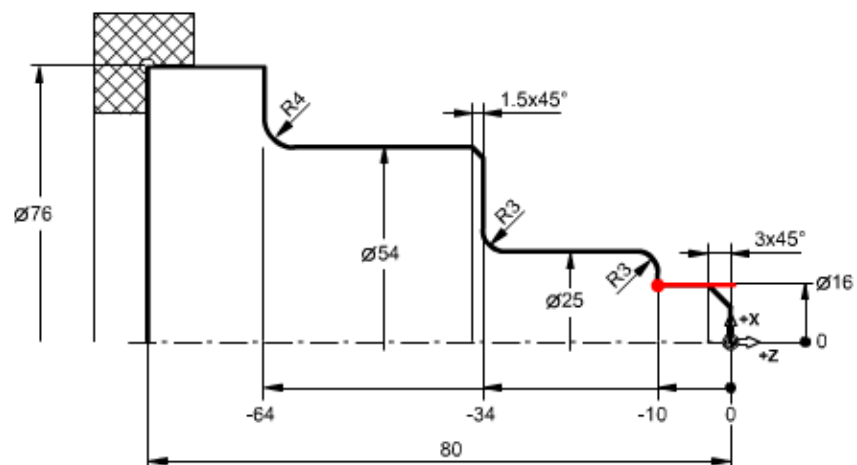
1.6.2

Звертайте увагу на вид вказування розмірів (абсолютні X/Z або інкрементальні K/I)
Перехідний елемент $3 \times 45^\circ$ може бути повернутим лише після того, як буде запрограмованим наступний елемент.



U	<input type="text"/>
K	<input type="text"/>

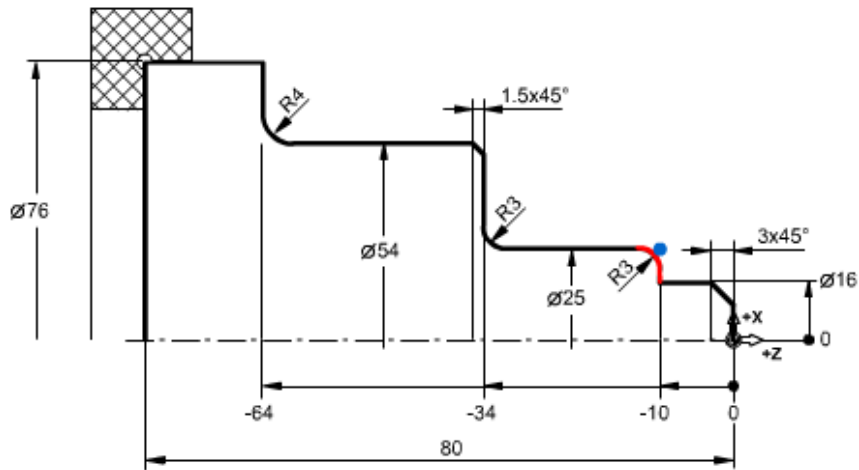
1.6.3



X	<input type="text"/>
Z	<input type="text"/>

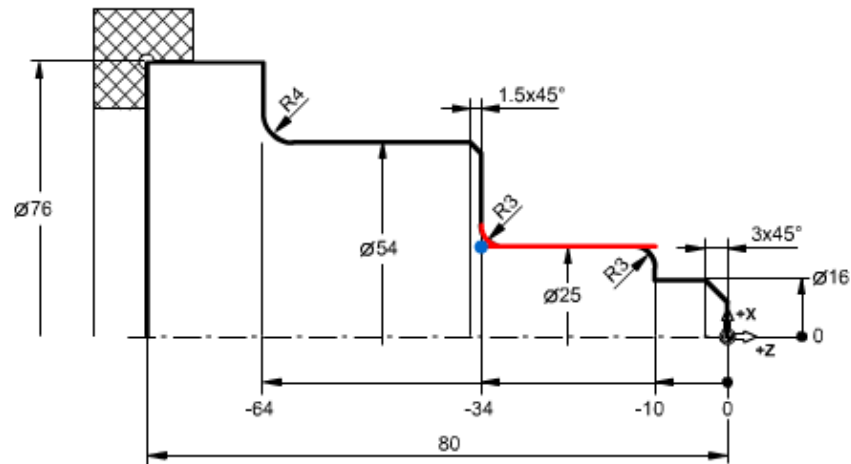
1.6.4

Перехідний елемент R3 може бути повернутим лише після того, як буде запрограмованим наступний елемент.



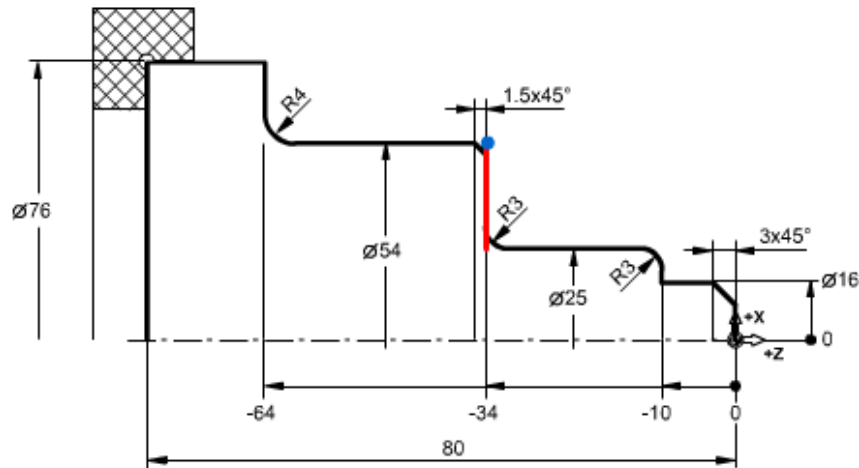
U	<input type="text"/>
R	<input type="text"/>

1.6.5



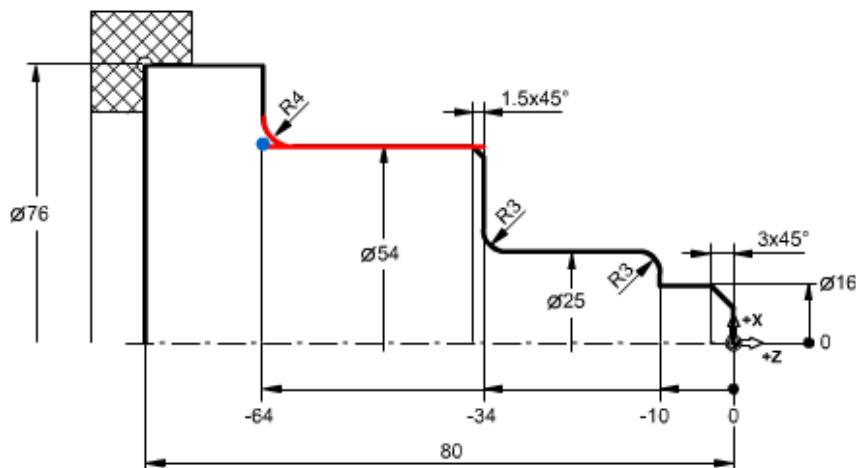
W	<input type="text"/>
R	<input type="text"/>

1.6.6



U	<input type="text"/>
K	<input type="text"/>

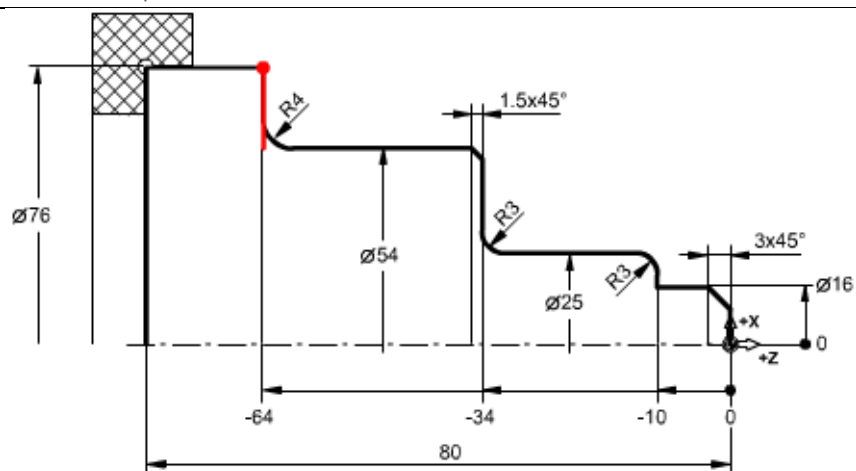
1.6.7 Перехідний елемент R3 може бути повернутим лише після того, як буде запрограмованим наступний елемент.



W

R

1.6.8

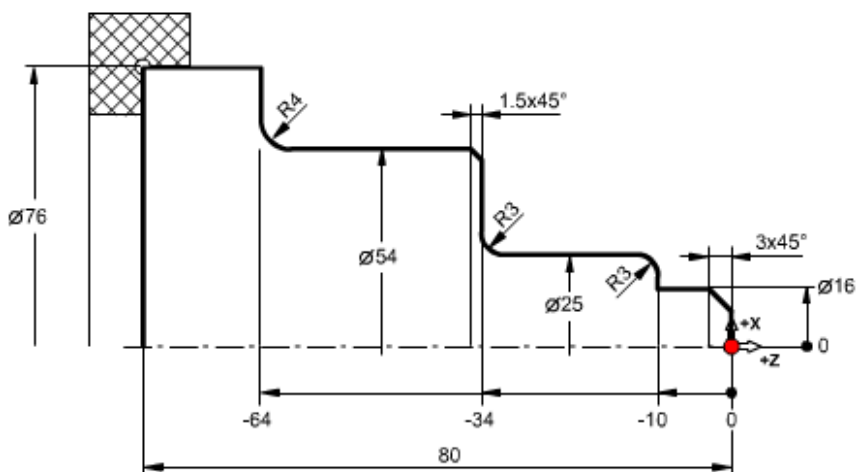


X

Z

Створення контуру з I/К.

1.6.9 Складіть зазначений контур. Введіть спочатку початкову точку.



немодальний

модальний

X

Z

Модальна активність

При русі з **паралельними осями** це значення координат програмувати не потрібно. Воно береться із попередньої позиції.

Приклад:

Початкова точка: X15 Z-15

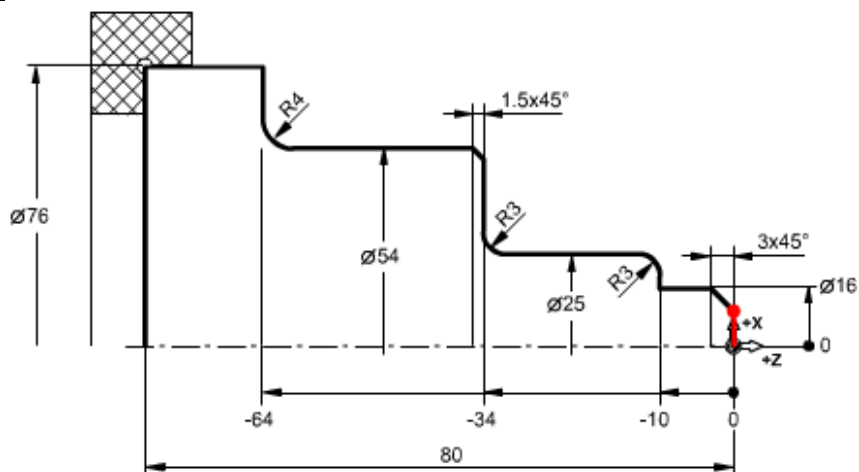
Кінцева точка: X15 Z-85

Введення значення X не є необхідним. Верстат приймає значення X15.

Введення абсолютного значення: Z-85

Введення у випадку інкрементальних значень: W-70.

1.6.10



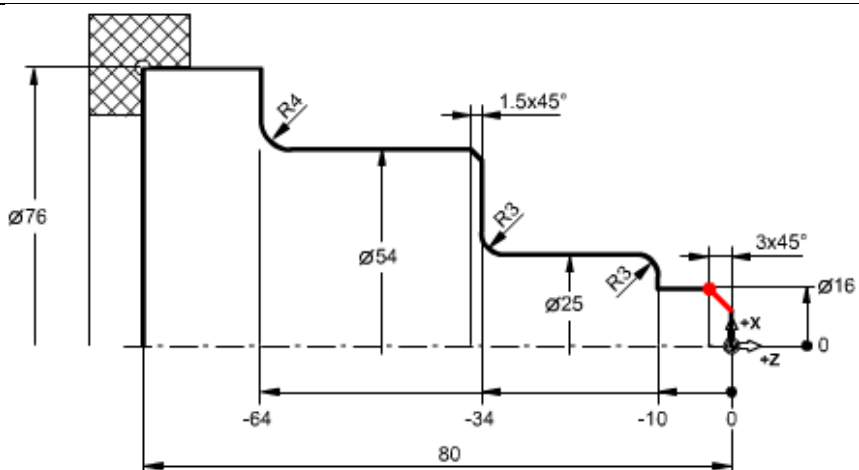
немодальный

модальный

X

Z

1.6.11



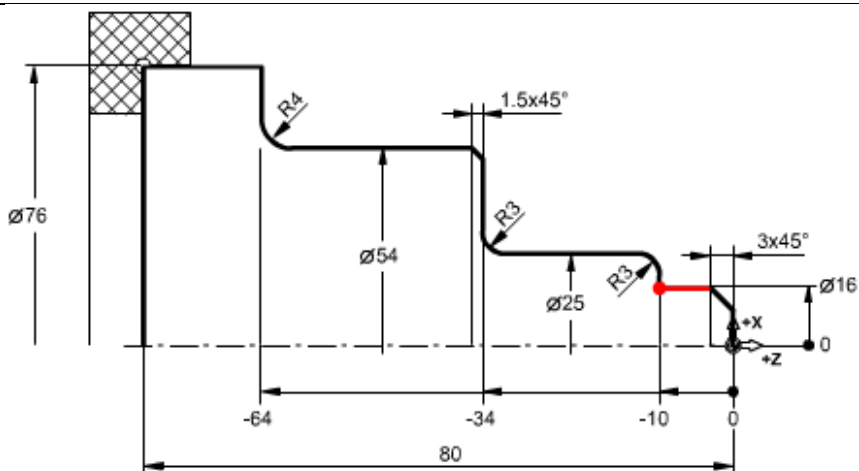
немодальный

модальный

X

Z

1.6.12



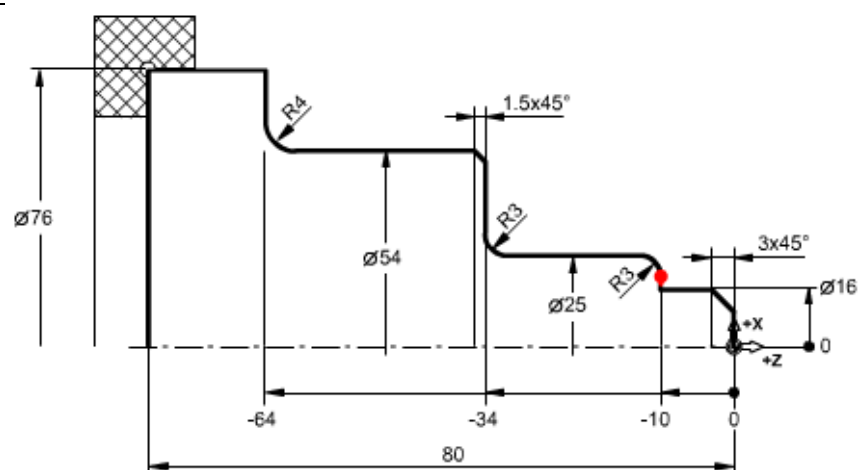
немодальный

модальный

X

Z

1.6.13



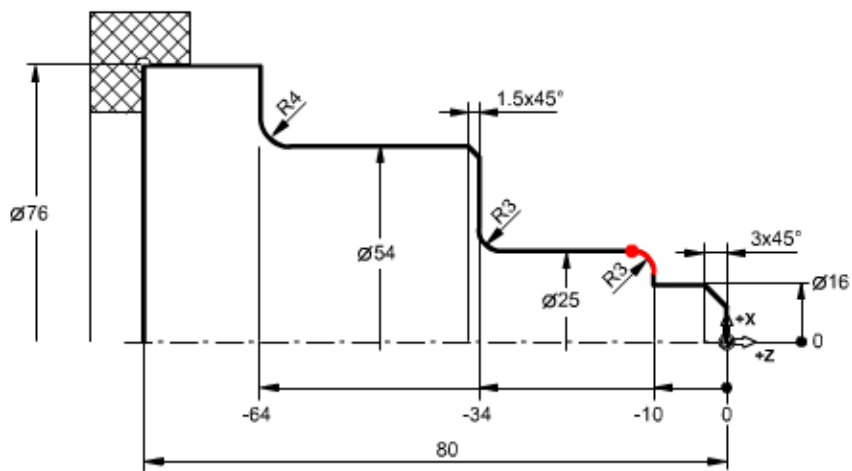
немодальный

модальный

X

Z

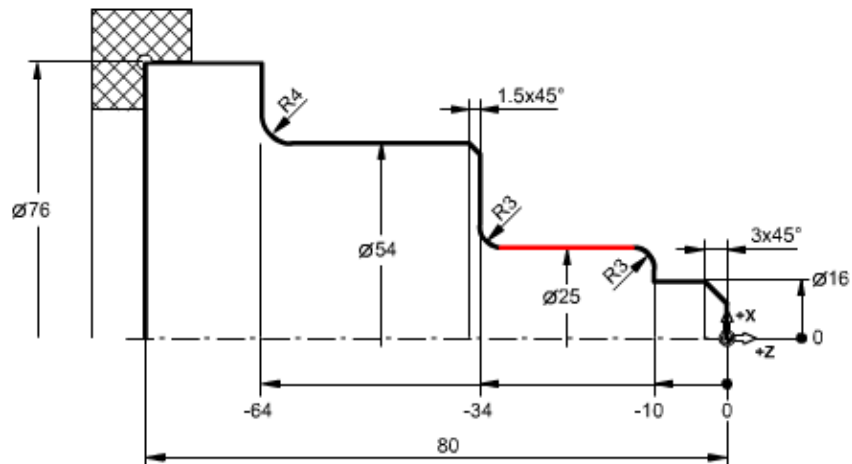
1.6.14


 немодальный

 модальный

G	<input checked="" type="checkbox"/>
X	<input type="checkbox"/>
Z	<input type="checkbox"/>
I	<input type="checkbox"/>
K	<input type="checkbox"/>

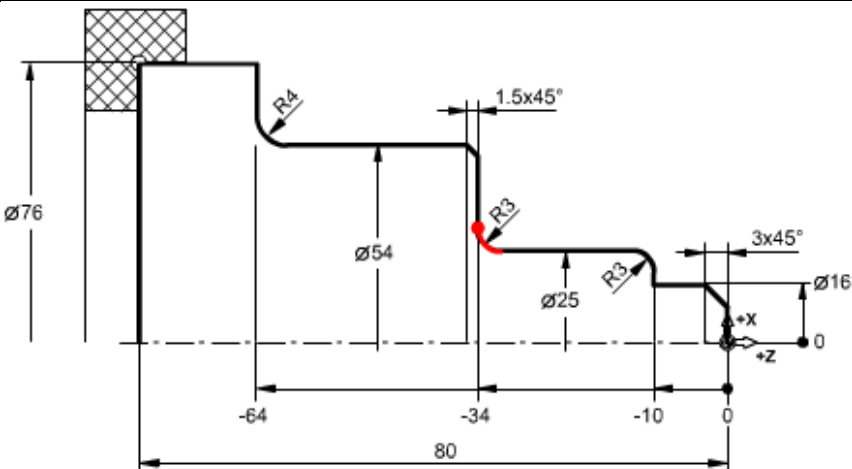
1.6.15


 немодальный

 модальный

X	<input checked="" type="checkbox"/>
Z	<input type="checkbox"/>

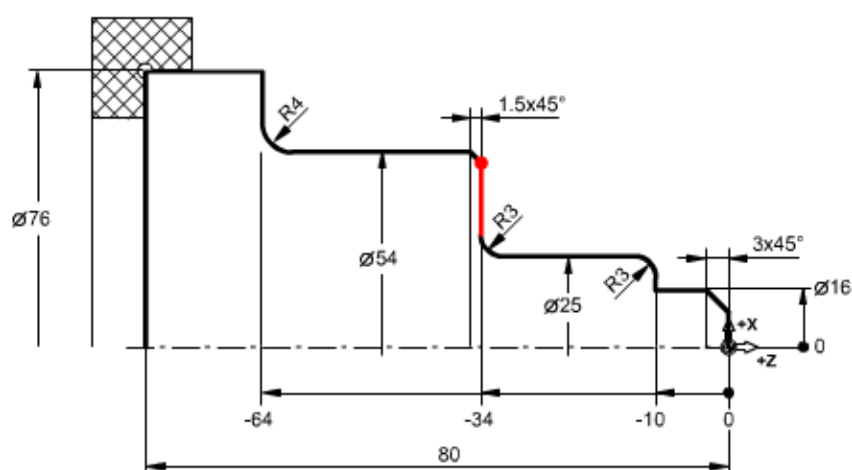
1.6.16


 немодальный

 модальный

G	<input checked="" type="checkbox"/>
X	<input type="checkbox"/>
Z	<input type="checkbox"/>
I	<input type="checkbox"/>
K	<input type="checkbox"/>

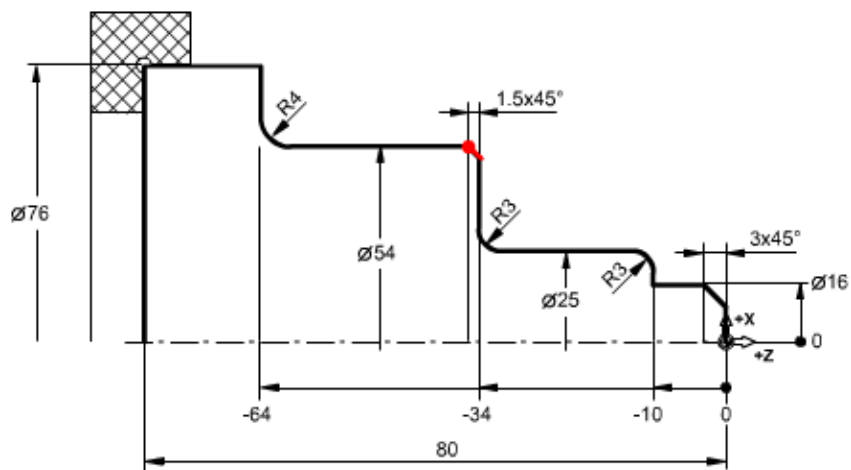
1.6.17


 немодальный

 модальный

X	<input checked="" type="checkbox"/>
Z	<input type="checkbox"/>

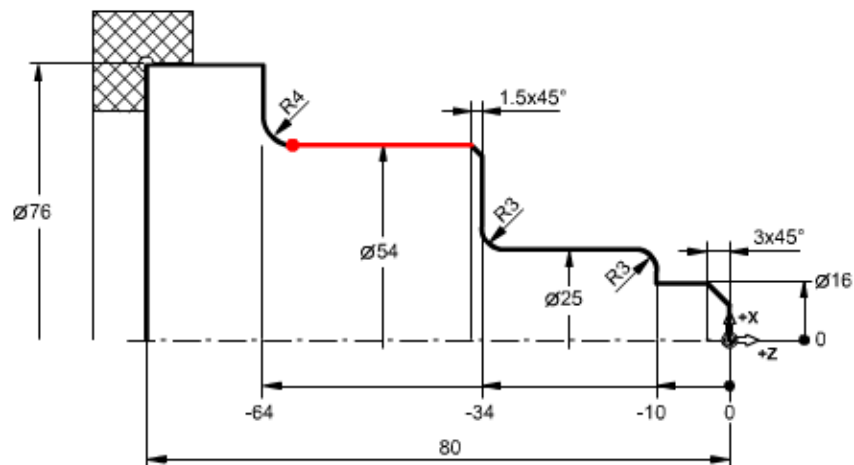
1.6.18


 немодальный

 модальный

X	<input type="text"/>
Z	<input type="text"/>

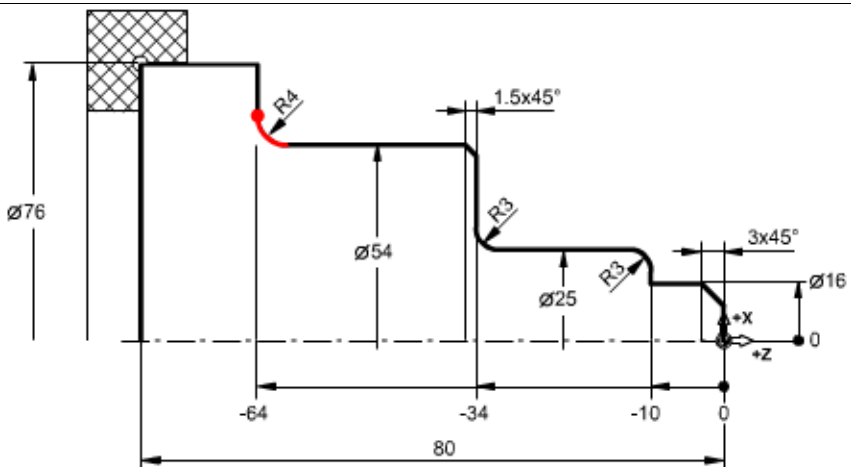
1.6.19


 немодальный

 модальный

X	<input type="text"/>
Z	<input type="text"/>

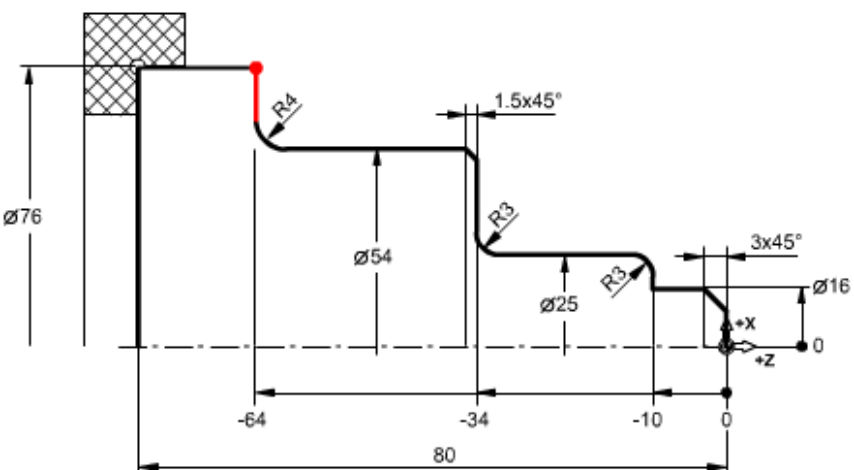
1.6.20


 немодальный

 модальный

G	<input type="text"/>
X	<input type="text"/>
Z	<input type="text"/>
I	<input type="text"/>
K	<input type="text"/>

1.6.21


 немодальный

 модальный

U	<input type="text"/>
W	<input type="text"/>

Контрольні запитання:

1. Що розуміють під лінійною інтерполяцією?
2. Що розуміють під коловою інтерполяцією?
3. Які типи інтерполяцій Вам відомі?
4. Що називають дискретою?
5. За допомогою яких функцій задається колова інтерполяція?
6. Для чого використовують функцію F?
7. Поняття модальні та немодальні коди.
8. Варіанти програмування дуг кіл, їх сутність.
9. Варіанти програмування фасок та округлень.

Лабораторна робота №3

Технологічні основи: швидкість, подача, функції T та M

Мета роботи: ознайомитися з технологічними основами для токарної обробки на верстатах ф. Наас.

Теоретичні відомості

Швидкість обертання.

Одиницею швидкості обертання n є об/хв. або 1/хв. При програмуванні швидкість обертання вказується з адресою S.

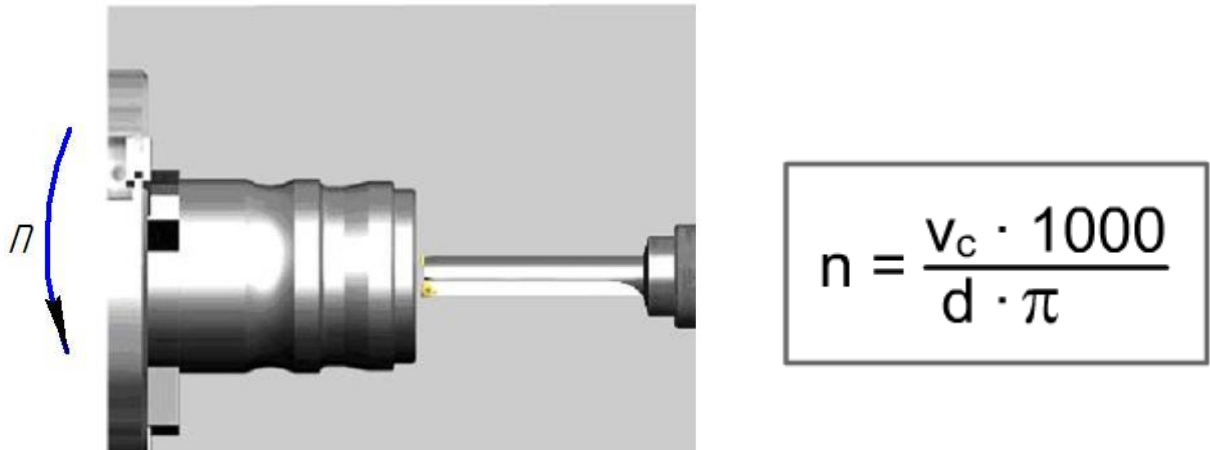


Рис. 3.1. До визначення швидкості обертання

Наприклад:

- для сталі з високою міцністю $n=1000$ об/хв. (**G97 S1000**);
- для сталі з низькою міцністю $n=3000$ об/хв. (**G97 S3000**).

G97 – постійна швидкість різання «Вимкн.» Пристрою ЧПУ дається завдання НЕ регулювати швидкість шпинделя в залежності від радіуса різання. Команда використовується для відміни G96. Під час дії коду G97 S виражається в об/хв.

Подача F (Feed) відноситься до шляху на оберт.



Рис. 3.2. До визначення подачі

Чим більша подача, тим більша висота мікронерівностей. Наприклад:

- при обдирці $f=0,4$ мм/об (**F0,4**);
- при чистовій обдирці $f=0,2$ мм/об (**F0,2**).

Швидкість подачі V_f . Чим більша подача F (при однаковій швидкості обертання), тим більша швидкість подачі V_f . Так як подача відноситься до подачі на оберт, то швидкість подачі (при однаковому значенні подачі F) чим більша, тим більша швидкість обертання.

Наприклад:

- при F0.2, S1000 - $V_f = 200$ мм/хв.;
- при F0.2, S2000 - $V_f = 400$ мм/хв.;
- при F0.4, S1000 - $V_f = 400$ мм/хв.;
- при F0.4, S2000 - $V_f = 800$ мм/хв.

$$v_f = n \cdot f$$

Швидкість різання V_c . При обдиранні, чистовій обробці і прорізанні зняття стружки, як правило, практично завжди відбувається з постійною швидкістю різання, особливо для забезпечення незмінної якості поверхні. Відповідною функцією є G96 (Постійна швидкість різання «Вкл.»). Пристрою ЧПУ дається завдання на підтримування постійної швидкості різання. Це означає, що при точінні менших діаметрів швидкість обертання шпинделя збільшується. Швидкість різання залежить від відстані між вершиною різця і віссю обертання шпинделя (радіус різання). Для визначення швидкості різання служить S-код.

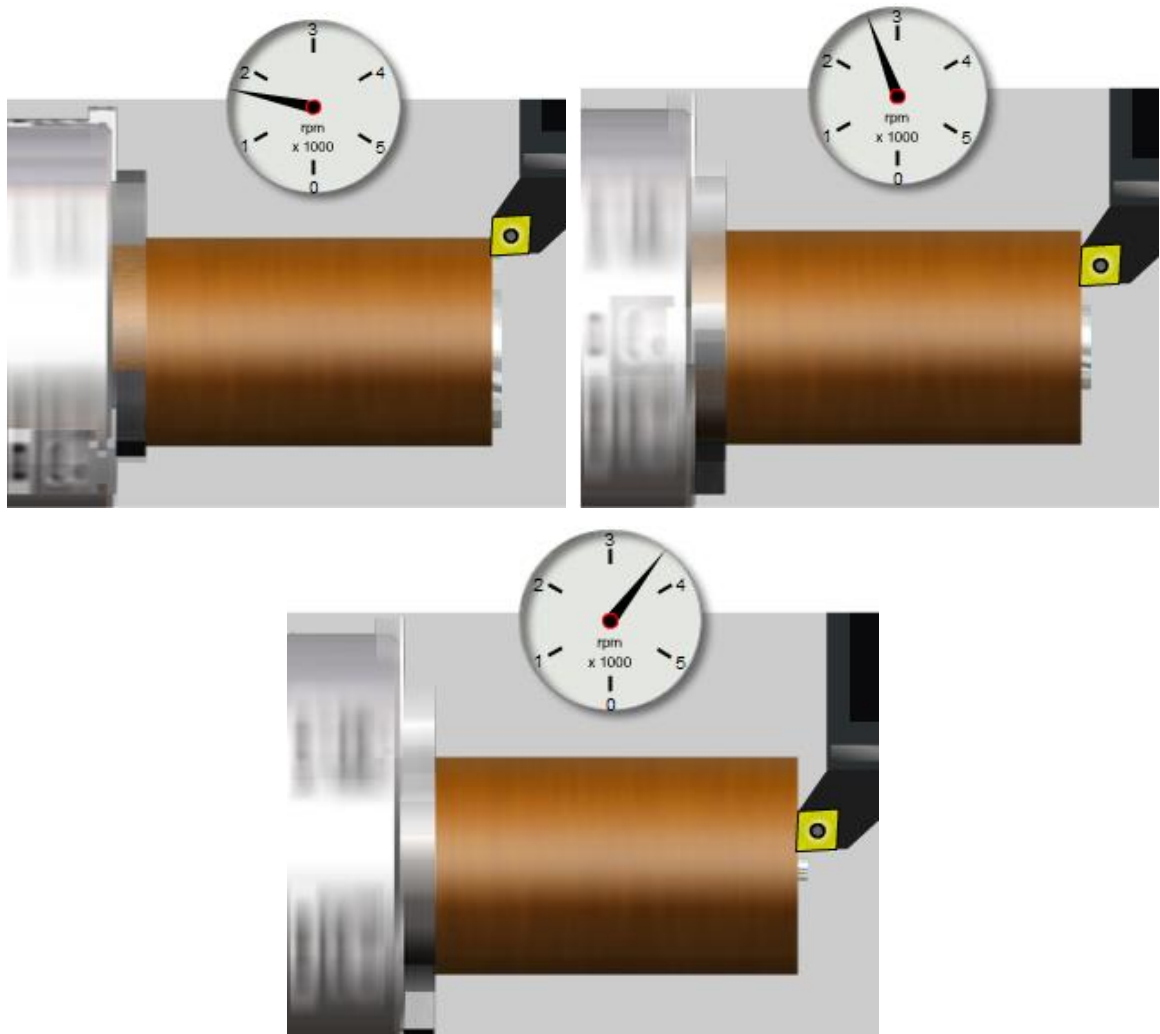


Рис. 3.3. Зменшення швидкості різання при зменшенні діаметра обробки

Наприклад:

- $V_c = 150$ м/хв. (**G96 S150**);

- $V_c = 300$ м/хв. (**G96 S300**).

Обмеження швидкості обертання. Для обмеження швидкості обертання служить функція G50. Дана функція впливає на:

- час виготовлення (витрати);
- якість поверхні;
- безпеку роботи.

G50 можна використовувати для обмеження максимальної швидкості шпинделя. Керування не допустить перевищення значення швидкості шпинделя, вказаного в S адресі команди G50. Це застосовується в режимі постійної швидкості робочої подачі (G96).

Наприклад:

N1 G50 S3000; (Оберти шпинделя не перевищать значення 3000 об/хв.)

N2 G97 M3; (Відміна постійної швидкості робочої подачі, обертання шпинделя вправо).

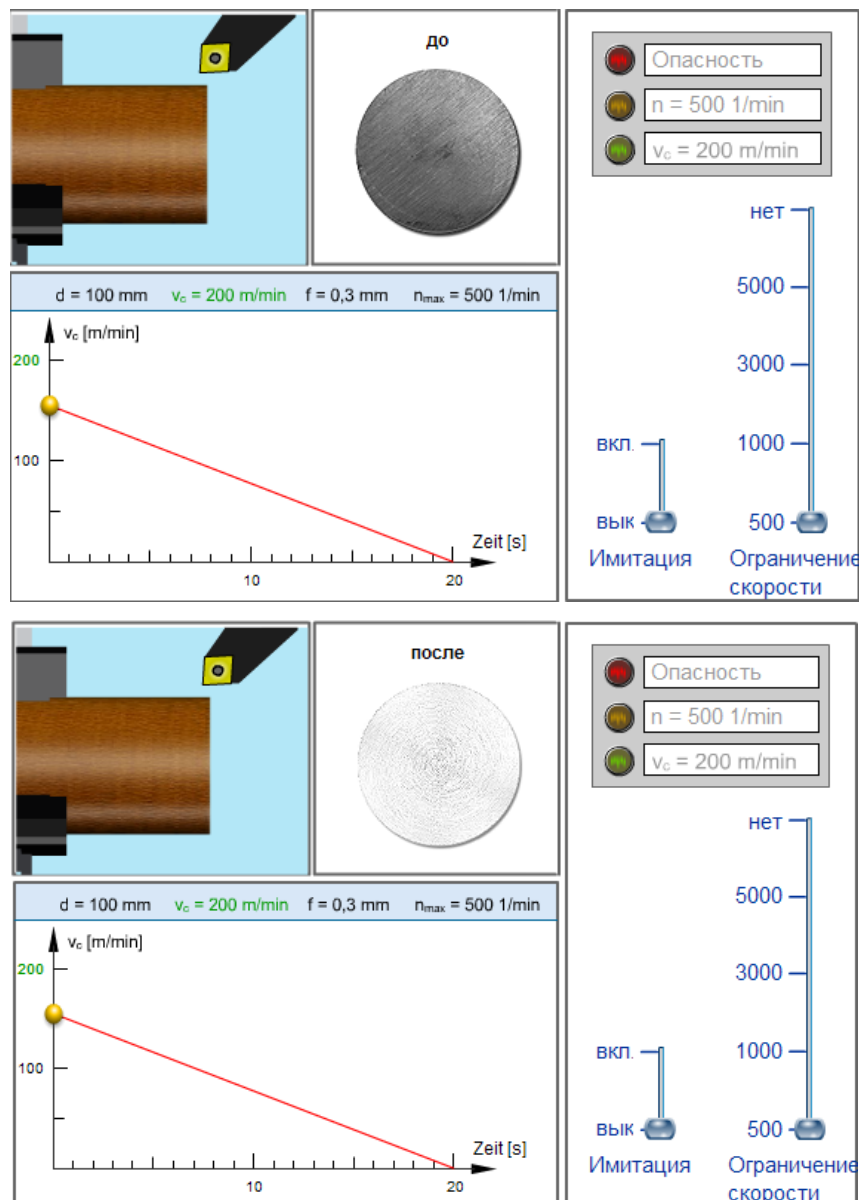


Рис. 3.4. Підрізання торця із обмеженням частоти обертання до 500 об/хв. з активною функцією постійної швидкості різання 200 м/хв.

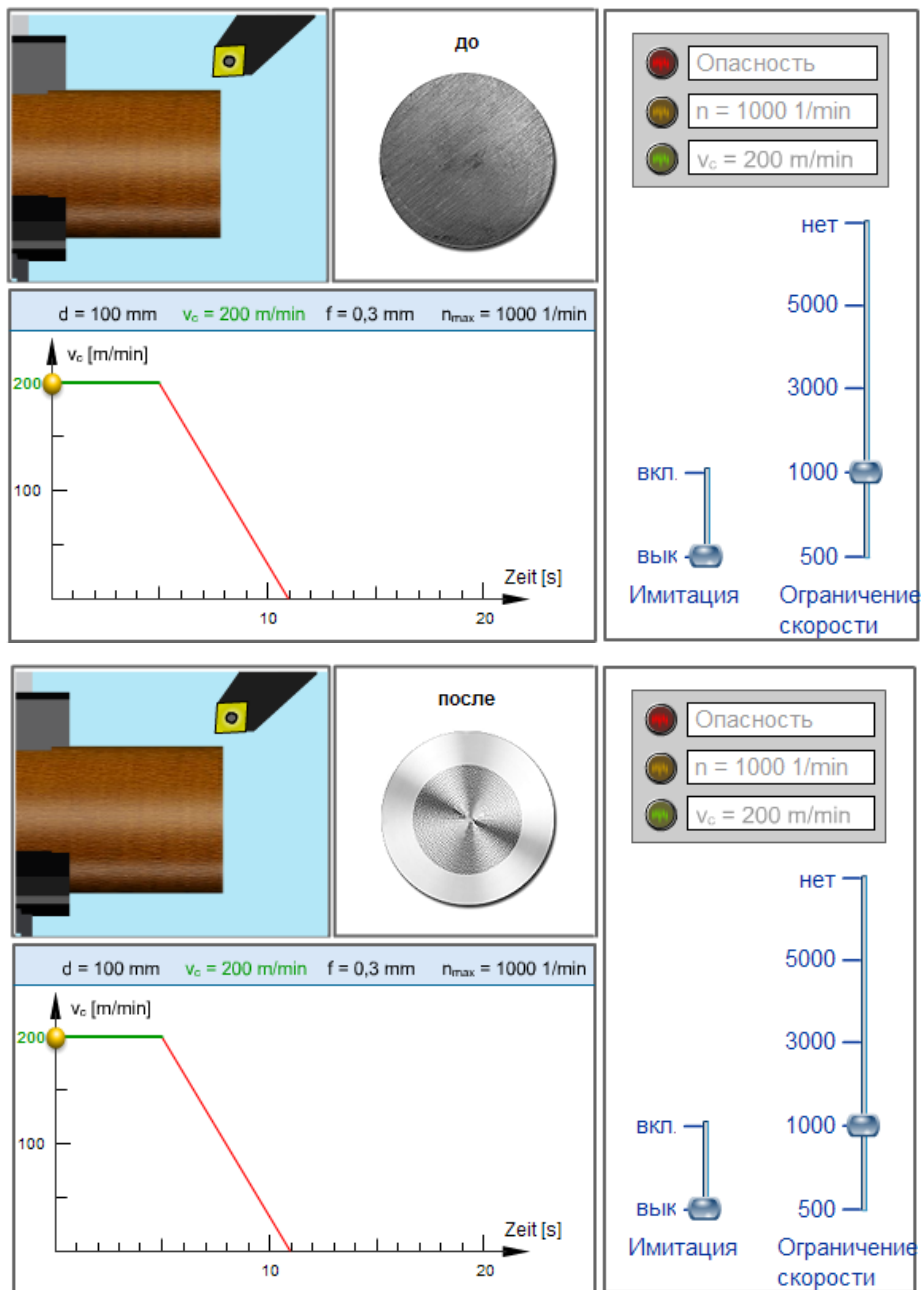


Рис. 3.5. Підрізання торця із обмеженням частоти обертання до 1000 об/хв. з активною функцією постійної швидкості різання 200 м/хв.

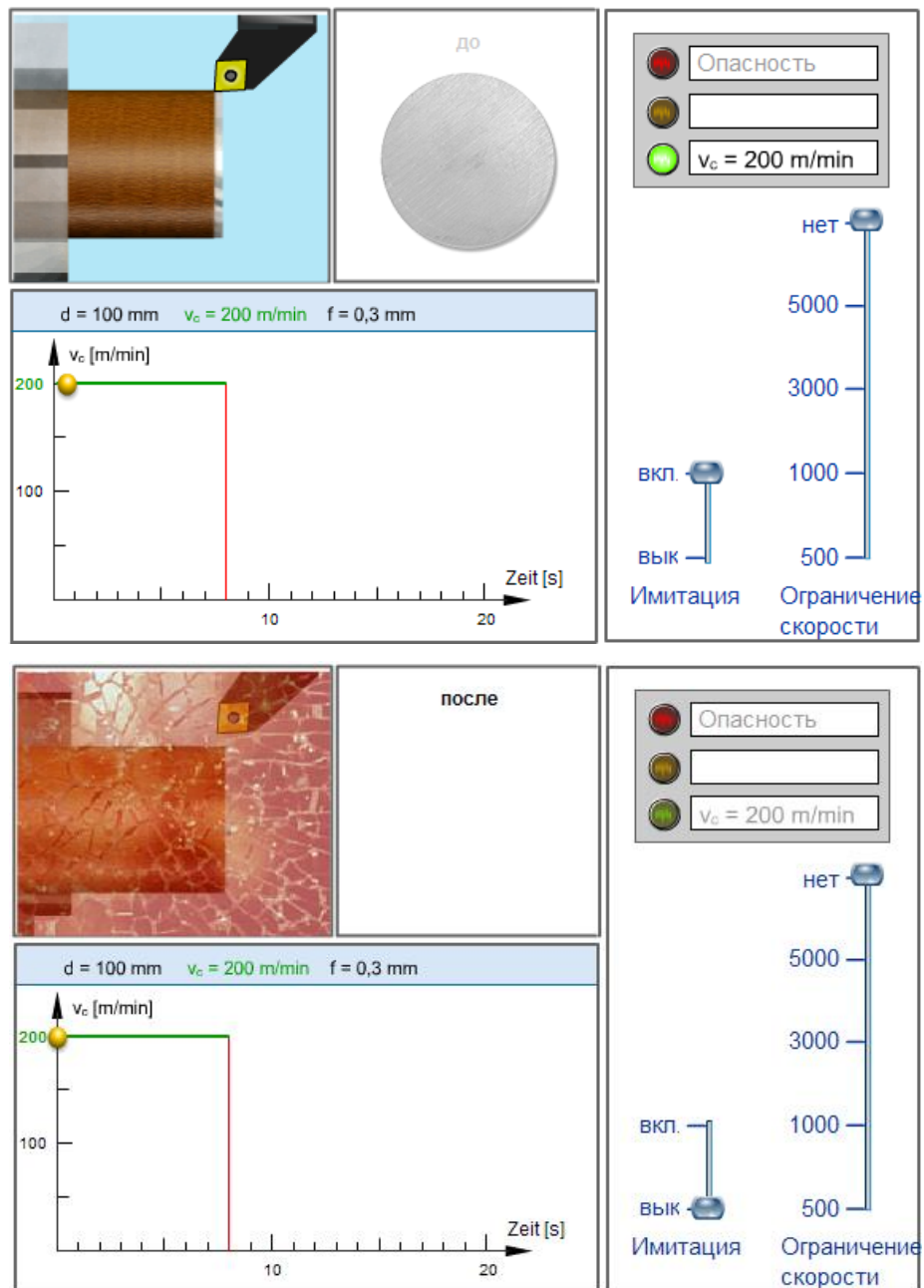


Рис. 3.6. Підрізання торця без обмеження частоти обертання з активною функцією постійної швидкості різання 200 м/хв.

Функція інструмента T.

Код **Tnno** використовується для вибору наступного інструмента (nn) або корекції (oo). Використання цього коду дещо різне в залежності від системи координат FANUC або YASNAC.

Система координат FANUC: T-коди мають формат Txxуу, де xx - вказує номер інструмента; уу - вказує індекси геометрії і зносу інструмента від 1 до 50. Якщо уу = 00, геометрія або знос інструмента не враховується.

Система координат YASNAC: T-коди мають формат Tnno, де nn має різні значення в залежності від того, як розміщений T-код всередині або зовні блока G50. Значення oo вказує знос інструмента от 1 до 50. Зовні блоку G50 nn вказує номер інструмента. Всередині блока G50 nn вказує індекс корекції на інструмент.

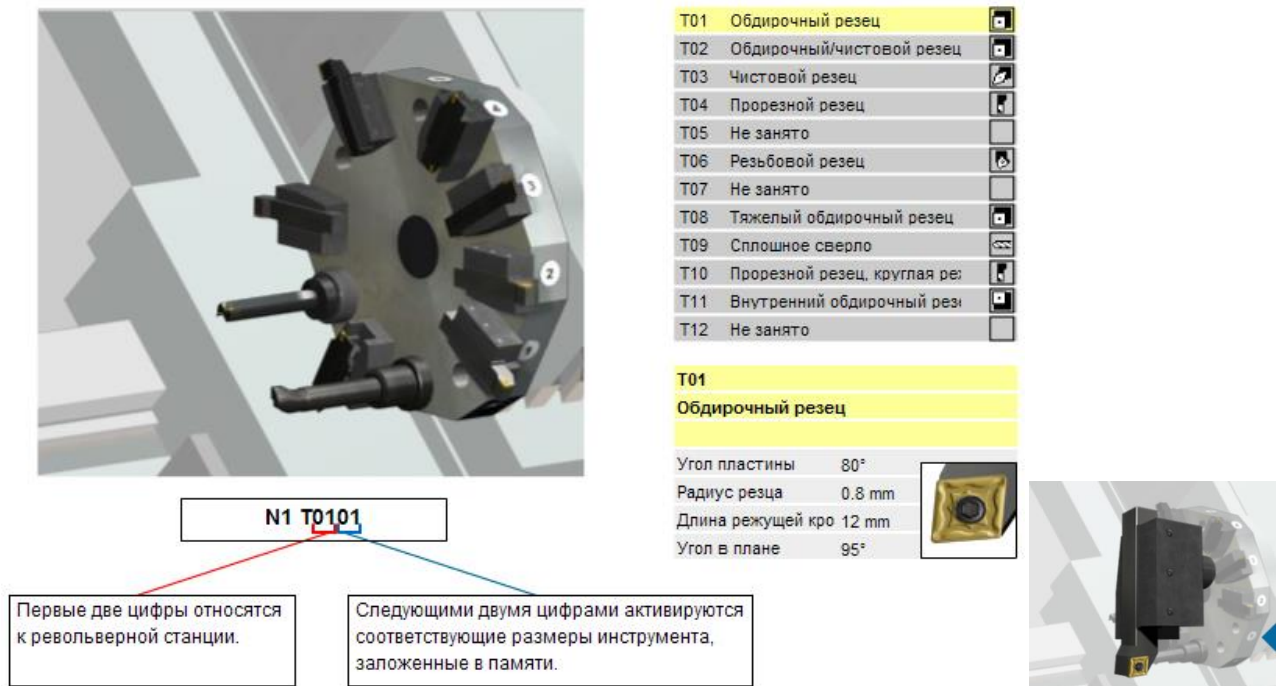


Рис. 3.7. Вибір необхідного інструмента

М-коди - це команди верстата, не пов'язані з рухом осей. Коди, що позначаються буквою М (Miscellaneous), називаються допоміжними і призначені для управління режимами роботи верстата. М-код може стояти як окремо, так і знаходитися в кадрі з G кодами. Деякі М коди працюють спільно з іншими адресами. Наприклад, М-код, що відповідає за напрям обертання шпинделя, зазвичай указується з адресою S, яка необхідна для завдання числа його обертів при обертанні:

N10 S1000 M03

Формат М-коду - це буква "М", за якою слідують дві цифри, наприклад M03.

У кожному рядку програми може бути запрограмований тільки один М-код. Всі М-коди діють в кінці блоку.

M00 Зупинка програми

M00 зупиняє програму. Код зупиняє осі, шпиндель, вимикає ЗОР (зокрема ЗОР, що пропускається через шпиндель). Наступний блок (після M00) буде виділений при перегляді в редакторі програм. При натисненні Cycle Start (Запуск циклу) виконання програми продовжиться з виділеного блоку.

M01 Додаткова зупинка програми

M01 працює так само, як M00, але для цього необхідно, щоб функція Optional Stop (Додаткова зупинка) була включена.

M02 Кінець програми

M02 закінчує програму. Проте самий поширений спосіб завершити програму - код M30.

Команди шпинделя M03 / M04 / M05

M03 - включає шпиндель в прямому напрямку. M04 - включає шпиндель у зворотному напрямку. M05 - зупиняє шпиндель.

Код M03 відповідає за пряме (за годинниковою стрілкою), а M04 - за зворотне обертання шпинделя (проти годинникової стрілки). Напрямок обертання визначається, якщо дивитися у від'ємному напрямку вісі Z (з боку шпинделя у бік заготовки).

Швидкість обертання шпинделя управляється кодом адреси S, наприклад, S1500 задасть швидкість обертання шпинделя 1500 об/хв.

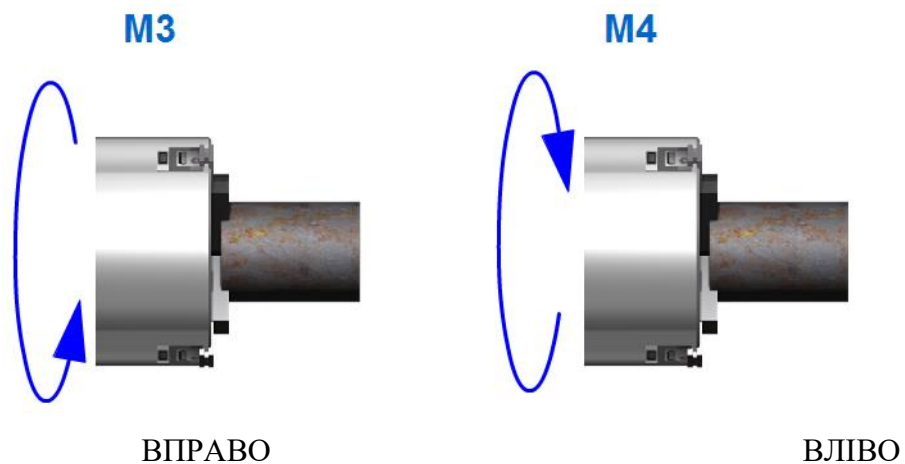


Рис. 3.8. Обертання шпинделя

M08 Включення подачі ЗОР / M09 Виключення подачі ЗОР

M08 включає опцію подачі ЗОР, а M09 вимикає її.

M88 Система подачі ЗОР під високим тиском Вкл. / M89 Система подачі ЗОР під високим тиском Викл.

M88 включає ЗОР високого тиску, а M89 вимикає ЗОР. Перед зміною інструменту потрібно вимкнути систему подачі ЗОР під високим тиском. Слід використовувати код M89 для виключення системи подачі ЗОР під високим тиском до початку обертання revolverної головки.

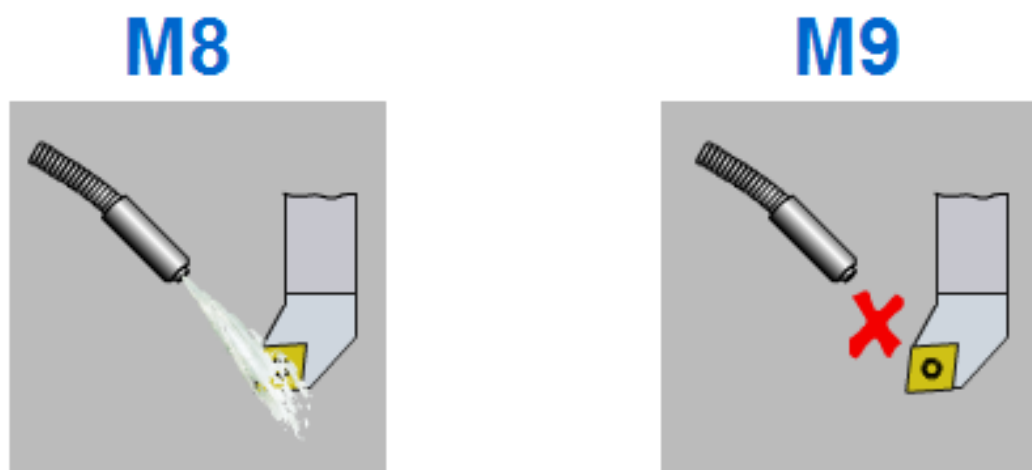


Рис. 3.9. Подача ЗОР

M88



M89

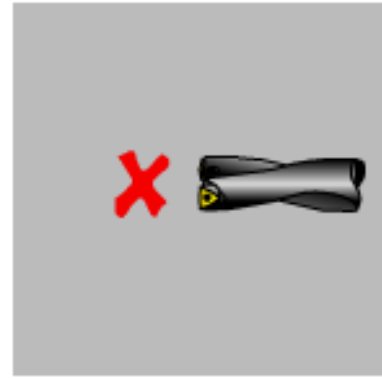


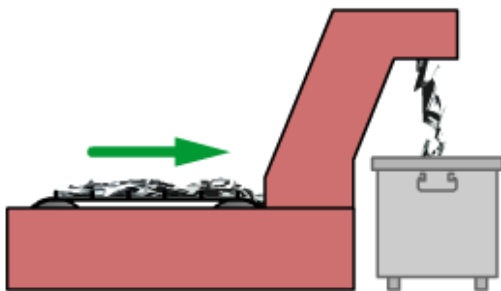
Рис. 3.10. Внутрішня подача ЗОР

M31 Запуск конвеєра стружки / M33 Зупинка конвеєра стружки

M31 запускає двигун додаткового конвеєра стружки в напрямку вперед - у напрямі видалення стружки з верстата. Конвеєр не працюватиме, якщо відкриті двері. Рекомендується використовувати конвеєр стружки з перервами. Безперервна робота приведе до перегріву двигуна.

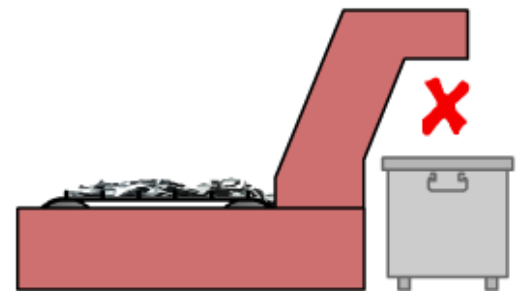
M33 зупиняє рух транспортера.

M31



ВКЛ.

M33



ВИКЛ.

Рис. 3.11. Стружковий транспортер

M97 Виклик місцевої підпрограми

Цей код використовується для виклику підпрограми, що починається в рядку N поточної програми. Код необхідний, і він повинен співпадати з номером одного з рядків в тій же програмі. Це зручно для простих підпрограм усередині програми; даний код не вимагає окремої програми. Підпрограма повинна закінчуватися M99. Lnn код в блоці M97 приводить до повторення виклику підпрограми nn разів. Приклад з кодом M97:

O0001

M97 P1000 L2 (Команда L2 змусить програму виконати рядок N1000 двічі)

M30

N1000 G00 G90 G55 X0 Z0 (Виконується рядок N, який виконуватиметься після M97 P1000)

S500 M03

G00 Z-.5

G01 X.5 F100.

G03 ZI-.5

G01 X0

Z1. F50.

G91 G28 Z0

G90

M99

M98 Виклик підпрограми

Цей код використовується для виклику підпрограми, його формат M98 Pnnnn (Pnnnn - це номер програми, що викликається). Підпрограма повинна бути в списку програм і повинна містити M99 для повернення в основну програму. У одному рядку з M98 можна встановити лічильник Lnn, що приведе до виклику підпрограми nn разів перед переходом до наступного блоку.

O0001 (Номер головної програми)

M98 P100 L4; (Виклик підпрограми, номер підпрограми, повтор 4 рази)

M30 (Кінець програми)

O0100 (Номер підпрограми)

G00 G90 G55 X0 Z0 (Виконується рядок N, який виконуватиметься після M97 P1000)

S500 M03

G00 Z-.5

G01 X.5 F100.

G03 ZI-.5

G01 X0

Z1. F50.

G91 G28 Z0

G90

M99

M99 - Повернення з підпрограми або циклу

Цей код використовується для повернення в головну програму з підпрограми або макросу, його формат M99 Pnnnn (Pnnnn - це рядок в головній програмі, до якої слід повернутися). В результаті використання цієї команди в головній програмі, головна програма перейде назад до початку без зупинки.

Примітки по програмуванню – для системи Fanuc використовується наступний код:

виклик програми: Haas Fanuc

O0001 O0001

... ..

N50 M98 P2 N50 M98 P2

N51 M99 P100 ...

... N100 (продовжити тут)

N100 (продовжити тут) ...

... M30

M30

підпрограма: O0002 O0002

M99 M99 P100

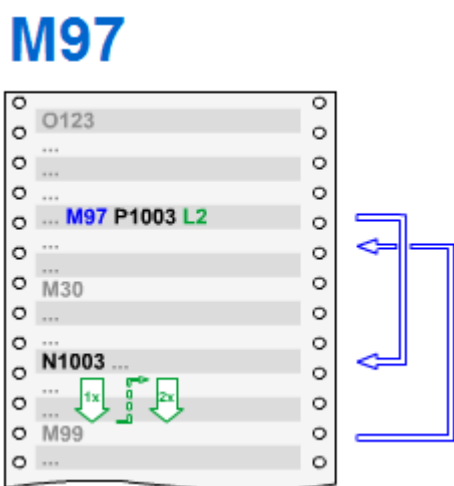


Рис. 3.12. Локальна підпрограма

P – номер програми

L – кількість повторень

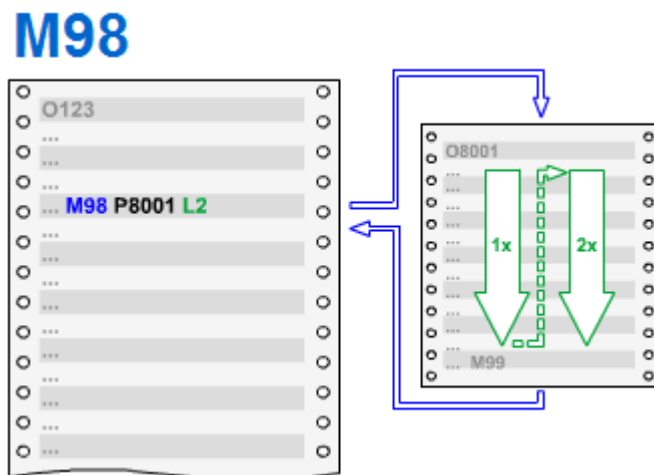
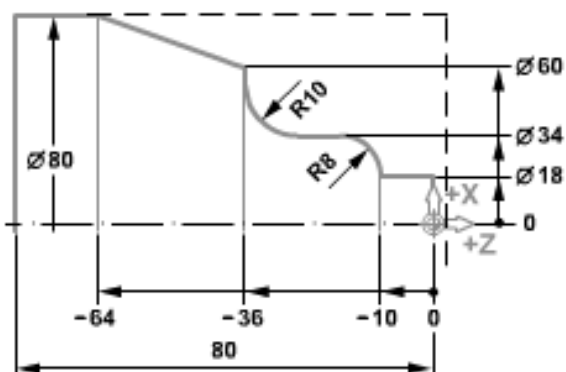


Рис. 3.13. Виклик підпрограми

M99 – кінець підпрограми

Порядок виконання роботи

1. За допомогою відомих кодів розробити управляючу програму обробки контуру деталі:



2. Розробити управляючу програму обробки зовнішнього контуру деталі за варіантами креслень (додаток 1).

Контрольні запитання:

1. Що розуміють під швидкістю обертання?
2. Функції G96 / G97.
3. Що розуміють під подачею?
4. Швидкість подачі.
5. Швидкість різання та її обмеження.
6. Функція G50 та її вплив на якість обробки.
7. Функція інструмента T.
8. Призначення M-кодів.
9. Призначення кодів: M01, M02.
10. Призначення кодів: M03, M04, 05.
11. Призначення кодів: M08, M09, M88, M89.
12. Призначення кодів: M31, M33.
13. Призначення кодів: M97, M98, M99.

Лабораторна робота №4

Постійні цикли. Свердління отворів

Мета роботи: ознайомитися з постійними циклами для верстатів з ЧПУ.

Теоретичні відомості

Постійними циклами називаються спеціальні макропрограми, закладені в ПЧПУ для виконання стандартних операцій механічної обробки. Практично всі верстати з ЧПУ мають набір циклів для обробки отворів - цикли свердління, розточування та нарізання різьб. Ці цикли спрощують процес написання УП і заощаджують час, тому що дозволяють за допомогою одного кадру виконати безліч переміщень.

Припустимо, що необхідно просвердлити кілька отворів у деталі. Щоб просвердлити один отвір потрібно на робочій подачі ввести свердло на необхідну глибину, потім вивести його вгору на прискореній подачі, і перемістити до наступного отвору. Програма, представлена в табл. 4.1 демонструє, як просвердлити кілька отворів без використання постійних циклів.

Таблиця 4.1

УП для свердління отворів без використання постійних циклів

%	Початок програми
O0005	Номер програми
N100 G21	Введення метричних даних
N102 G0 G17 G40 G49 G80 G90	Строчка безпеки
N104 T1 M6	Виклик інструмента
N106 G0 G90 G54 X5. Y5. S1000 M3	Переміщення до отвору №1
N108 G43H1Z100.	Корекція на довжину інструмента
N110 Z10.	
N112G1Z-8. F70.	Свердління отвору №1
N114G0Z10.	Вивід свердла на прискореній подачі
N116X15.	Переміщення до отвору №2
N118G1Z-8.F70	Свердління отвору №2
N120 G0Z10.	Вивід свердла на прискореній подачі
N122 X-5.	Переміщення до отвору №3
N124G1Z-8. F70	Свердління отвору №3
N126 G0Z10.	Вивід свердла на прискореній подачі
N128 X-15.	Переміщення до отвору №4
N130G1Z-8. F70	Свердління отвору №4
N132 G0Z10.	Вивід свердла на прискореній подачі
N134X5. Y-5.	Переміщення до отвору №5
N136G1Z-8. F70	Свердління отвору №5

N138G0Z10.	Вивід свердла на прискореній подачі
N140X15.	Переміщення до отвору №6
N142G1Z-8. F70	Свердління отвору №6
N144 G0Z10.	Вивід свердла на прискореній подачі
N146 X-5.	Переміщення до отвору №7
N148G1Z-8. F70	Свердління отвору №7
N150 G0Z10.	Вивід свердла на прискореній подачі
N152 X-15.	Переміщення до отвору №8
N154G1Z-8. F70	Свердління отвору №8
N156 G0Z10.	Вивід свердла на прискореній подачі
N158Z100.	
N160 M5	
N166 M30	Кінець програми
%	

Використання постійного циклу спрощує процес створення програми для обробки отворів, робить її легко читаємою й суттєво зменшує в розмірі. Створимо нову УП для обробки цих же отворів з постійним циклом свердління (табл. 4.2).

Таблиця 4.2

УП для свердління отворів з використанням постійних циклів

%	Початок програми
O0005	Номер програми
N100 G21	Введення метричних даних
N102 G0 G17 G40 G49 G80 G90	Строчка безпеки
N104 T1 M6	Виклик інструмента
N106 G0 G90 G54 X5. Y5. S1000 M3	Переміщення до отвору №1
N108 G43H1Z100.	Корекція на довжину інструмента
N110 Z10.	
N112G99G81Z-8. RIO. F70.	Виклик циклу свердління
N114X15.	Координати отвору №2
N116 X-5.	Координати отвору №3
N118 X-15.	Координати отвору №4
N120 X5. Y-5.	Координати отвору №5
N122X15.	Координати отвору №6
N124 X-5.	Координати отвору №7
N126 X-15.	Координати отвору №8
N128 G80	Скасування циклу свердління

N130Z100.	
N132 M5	
N138 M30	Кінець програми
%	

Очевидно, що нова програма має менший розмір. У кадрі N112 є код G81 для виклику циклу свердління. У цьому ж кадрі є адреси, відповідальні за налаштування параметрів циклу. Адреса Z позначає глибину свердління, а R визначає висоту відводу свердла з отвору щодо нульової площини. У наступних кадрах вказані координати оброблюваних отворів. У них не потрібно ставити коди виклику циклу свердління, тому що G81 буде залишатися активним, поки його не скасують за допомогою коду G80.

Працювати з постійними циклами дуже зручно. Наприклад, необхідно змінити глибину свердління й висоту виводу свердла з отвору. При роботі із програмою без постійного циклу доведеться відредагувати її практично повністю. Якщо ж використовуються постійний цикл свердління, то для досягнення потрібного ефекту досить змінити кілька параметрів.

Верстати з ЧПУ можуть мати різноманітні цикли: від досить простих - для свердління, розточування й нарізання різьб до більш складних - для обробки контурів і карманів. Деякі цикли стандартизовані, хоча більшість із них розробляються виробниками верстатів і систем ЧПУ самостійно, без оглядки на формати інших компаній, виходячи із власних можливостей і бажань. Тому на різних верстатах, однакові, по суті, цикли можуть записуватися по-різному, що, звичайно ж, ускладнює програмування. В табл. 4.3 вказані стандартні цикли для обробки отворів, що використовуються на переважній більшості сучасних верстатів з ЧПУ.

Таблиця 4.3

Постійні цикли для свердління отворів

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

Код G81 призначений для виклику **стандартного циклу свердління**. Наступний кадр демонструє типовий формат цього циклу:

G81 X10.0 Y15.3 Z-3.0 R0.5 F50.

Адреси X і Y визначають координати оброблюваних отворів. Адреса Z вказує кінцеву глибину свердління, а R застосовується для встановлення площини відводу. Площина відводу - це координата по вісі Z, з якої починається свердління на робочій подачі. Площина відводу встановлюється трохи вище поверхні деталі, тому значення при R позитивне. Не варто встановлювати площину відводу дуже високо, інакше свердло на робочій подачі буде переміщатися занадто довго. Робоча подача для циклу встановлюється за допомогою слова даних F.

Постійні цикли і їх параметри є модальними. Викликавши цикл за допомогою відповідного G коду, у наступних кадрах вказуються координати отворів, які необхідно обробити, не програмуючи ніяких інших кодів і параметрів. Після кадру, що містить координати останнього отвору необхідно запрограмувати G80 - код скасування (закінчення) постійного циклу. Якщо цього не зробити, то всі наступні координати переміщень будуть вважатися координатами оброблюваних отворів.

Цикл свердління з витримкою викликається за допомогою команди G82. Функціонує цей цикл аналогічно стандартному циклу свердління, з єдиною різницею в тому, що при G82 на дні отвору запрограмований час очікування (витримка). Цикл свердління з витримкою часто застосовується для свердління глухих отворів, тому що запрограмований час очікування забезпечує краще видалення стружки із дна отвору. Адреса P установлює час очікування на дні отвору. Як правило, час витримки вказується в 1/1000 сек. без десяткової крапки. Наприклад, у наступному кадрові виконується цикл свердління з витримкою на дні отвору рівною 6.5 секунд:

G82 X10.0 Y15.3 Z-3.0 P6500 R0.5 F50.

При виконанні механічної обробки отворів за допомогою постійних циклів необхідно знати, що таке вихідна площина й площина відводу (рис. 4.1). Дві ці площини використовуються для керування переміщеннями по вісі Z між отворами. Про площину відводу уже було сказано. **Площина відводу** - це координата (рівень) по вісі Z, яка задається R адресою, з якої починається свердління на робочій подачі і у яку повертається інструмент, після того, як він досяг дна оброблюваного отвору. **Вихідна площина** - це координата (рівень), по вісі Z у якій розташовувався інструмент перед викликом постійного циклу. Код G98 використовується для роботи з вихідною площиною, а код G99 - із площиною відводу.

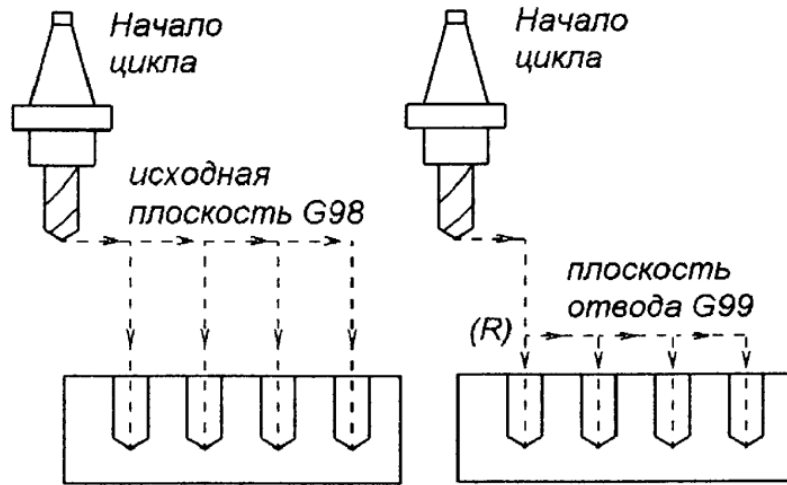


Рис. 4.1. Вихідна площина і площина відводу

Припустимо, що інструмент перебуває в координаті Z20.0 у момент виклику циклу свердління. Тоді вихідна площина буде розташовуватися на відстані 20 мм вище нульової точки по вісі Z. Тобто для встановлення вихідної площини не потрібно вказувати які-небудь спеціальні адреси. Однак для встановлення площини відводу необхідно використовувати адресу R. Формат кадру для циклу свердління має наступний вигляд:

G98 G81 X10.0 Y15.3 Z-3.0 R0.5 F50

або

G99 G81 X10.0 Y15.3 Z-3.0 R0.5 F50.

Якщо цикл свердління працює разом з кодом G98, то інструмент повертається до вихідної площини наприкінці кожного циклу й між усіма оброблюваними отворами. Код G98 застосовується, коли потрібна збільшена відстань відводу для того, щоб уникнути зіткнення інструмента з деталлю (рис. 4.2). Необхідно мати на увазі, що при використанні G98 відразу після зміни інструмента вихідна площина, швидше за все, буде встановлена дуже високо, і інструмент буде переміщатися до отвору недозволєно довго.

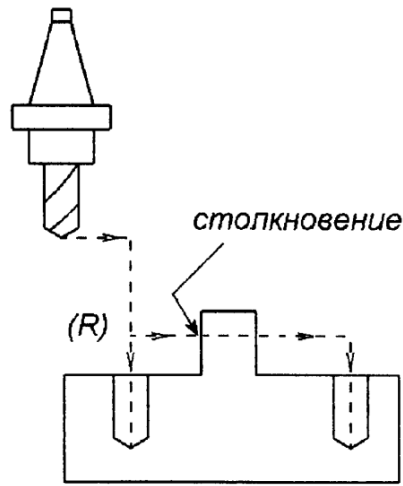


Рис. 4.2. Зіткнення інструмента із деталлю

Коли немає небезпеки зіткнення інструмента з деталлю, то звичайно використовують код G99, який дозволяє скоротити час при обробці безлічі отворів. У цьому випадку інструмент переміщається між отворами й виводиться вгору наприкінці циклу до координати по Z, установленій R словом даних. Зазвичай, системи ЧПУ дозволяють перемикатися між G98 і G99 прямо всередині постійного циклу між оброблюваними отворами:

G99 G81 X10.0 Y15.3 Z-3.0 R0.5 F50.

X20 Y20

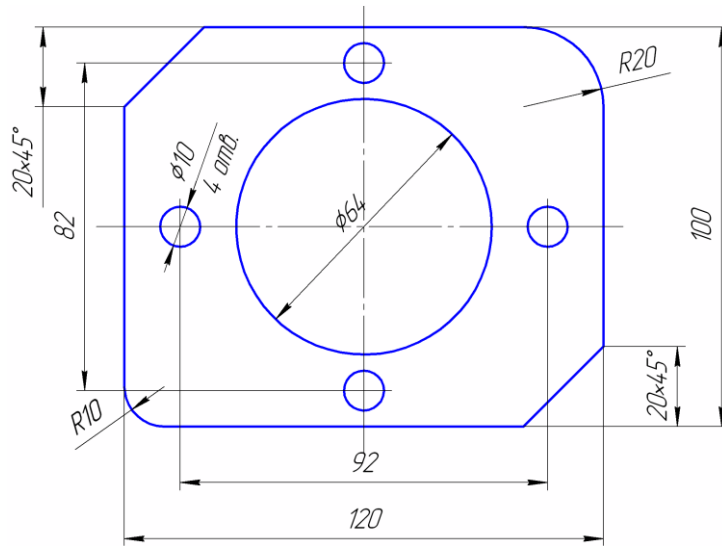
G98 X30 Y30

X40 Y40

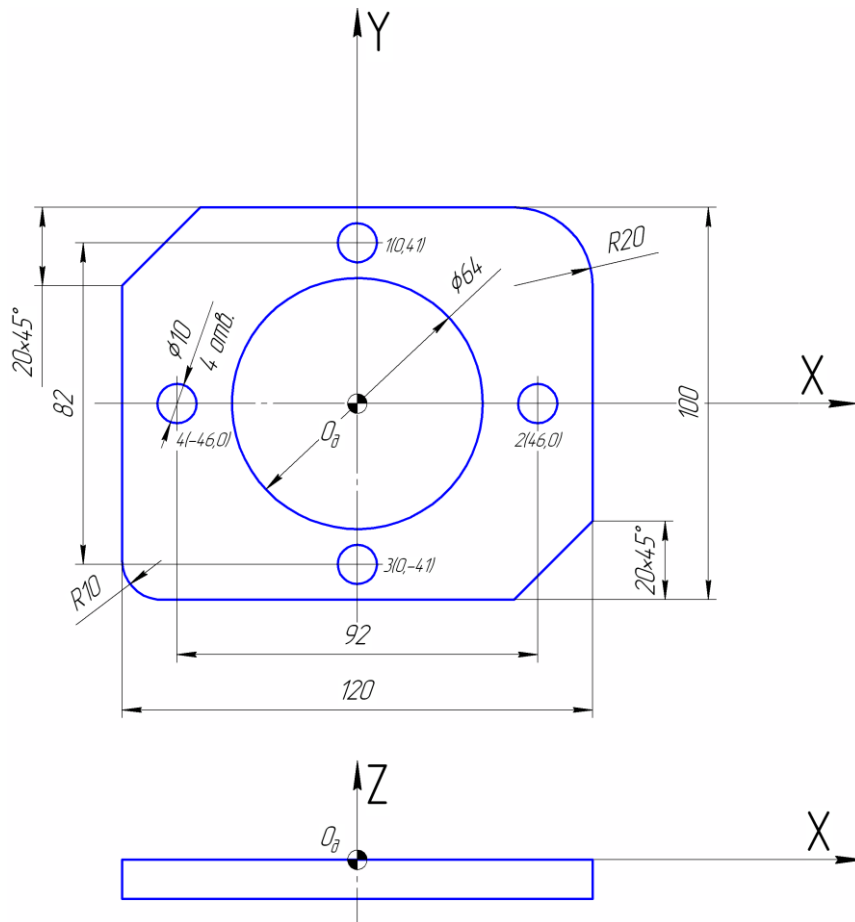
Порядок виконання роботи

Завдання: Розробити управляючу програму для свердління отворів (додаток 2) використовуючи постійні цикли, розробити технологічне креслення деталі (позначити нульову точку програми, інструмента, напрямки осей координат, вузлові точки та їх координати) вибрати інструмент, призначити режими різання.

Приклад: Розробити управляючу програму для свердління отворів Ø10 використовуючи постійні цикли, розробити технологічне креслення деталі (позначити нульову точку програми, інструмента, напрямки осей координат, вузлові точки та їх координати) вибрати інструмент, призначити режими різання.



1. Розробляємо технологічне креслення деталі на якому визначаємо положення нульової точки програми, напрямки координатних осей, вузлові точки та їх координати.



2. Розробляємо управляючу програму свердління отворів з використанням постійних циклів:

%

O0001 - номер програми

N10 G21 G40 G49 G54 G80 G90 - строчка безпеки

N20 M06 T01 - виклик інструмента №1

N30 G43 H01 - корекція на довжину інструмента №1

N40 M03 S1000 - включення обертання шпинделя за годинниковою стрілкою з частотою 1000 об/хв.
N50 G00 X0 Y41 - прискорене переміщення в точку (0,41)
N60 G00 Z10 - позиціонування в точку (0,41,10)
N70 G99 G81 Z-12.5 R10 F70 M08 - виклик стандартного циклу свердління, ввімкнення подачі ЗОР, свердління отвору 1 з подачею 70 мм/хв.
N80 X46 Y0 - свердління отвору 2
N90 X0 Y-41 - свердління отвору 3
N95 X-46 Y0 - свердління отвору 4
N100 G80 - відміна циклу свердління
N110 G00 Z50 - прискорене переміщення в точку (-46,0,50)
N120 G91 G28 X0 Y0 Z0 - повернення у вихідну позицію
N130 M05 - вимкнення обертання шпинделя
N135 M09 - вимкнення ЗОР
N140 M30 - завершення програми
%

3. В якості інструменту застосовуємо свердло $\varnothing 10$ B411A10000 ф. Кенаметал.

Контрольні запитання:

1. Що розуміють під постійними циклами?
2. Які переваги від застосування постійних циклів?
3. Назвіть стандартні постійні цикли.
4. Який код використовується для відміни постійних циклів.
5. Призначення циклу свердління з витримкою, код виклику, формат запису.
6. Площина відводу, код виклику, формат запису.
7. Вихідна площина, код виклику, формат запису.

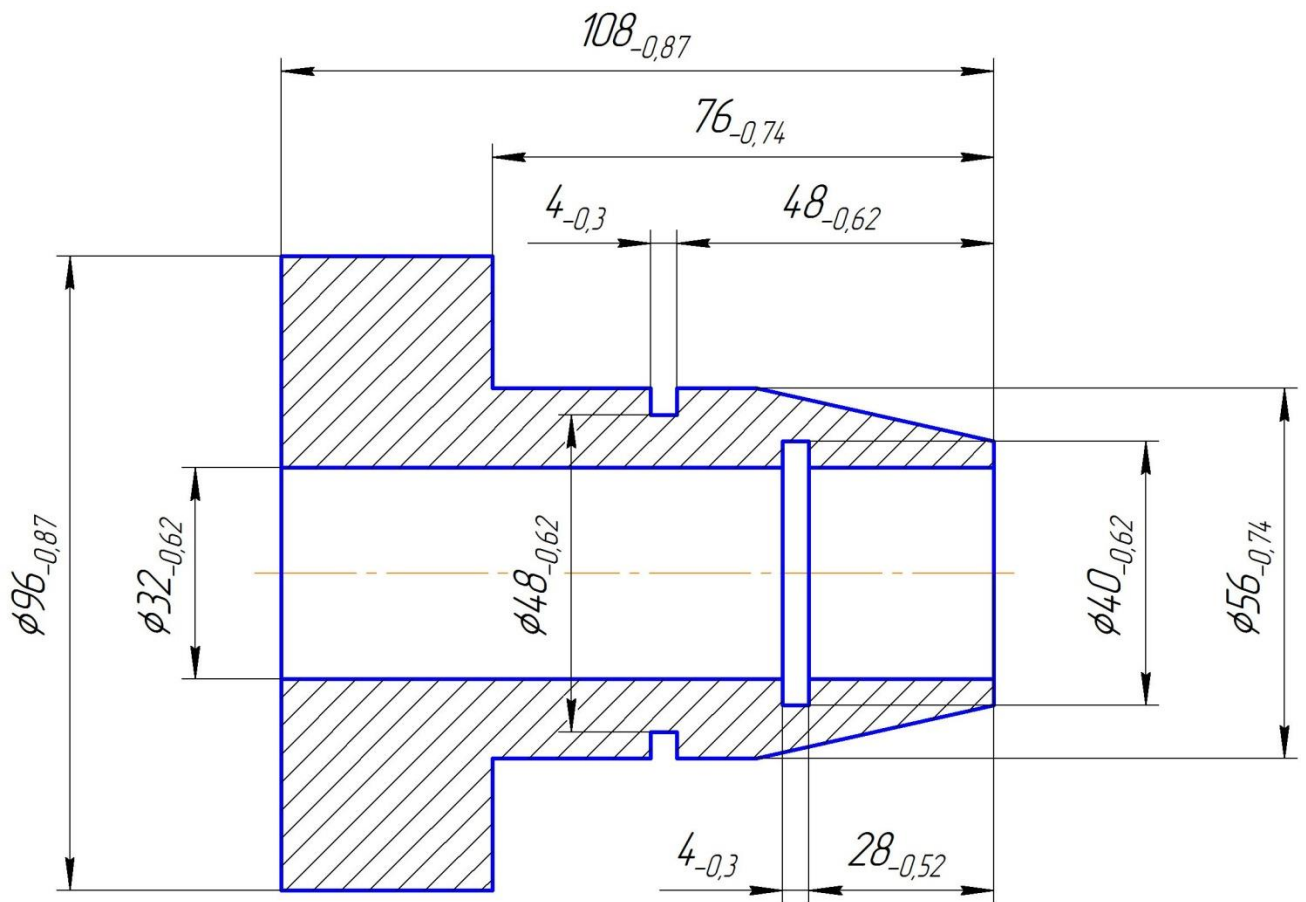
Література

1. Волченко И.О. Обработка деталей на станках с ЧПУ / И.О. Волченко, К.В. Стругов // . – СПб. : 2005. – 69 с.
2. Ловыгин, А. А. Современные станки с ЧПУ и САБ/САМ-системы / А. А. Ловыгин, А. В. Васильев, С. Ю. Кривцов // . – М. : Эльф ИПР, 2006. - 286 с.
3. Технологічні основи програмування для верстатів з ЧПК: методичні вказівки та завдання щодо виконання практичних робіт з використання програми «CNC SIMULATOR» / М-во освіти і науки України, Первомайський коледж нац. ун-ту кораблебудування ім. адмірала Макарова; [уклад. В.А. Гайворонський, Д.А. Бабиц]. – Первомайськ: ПК НУК ім. адм. Макарова, 2014. – 130 с.
4. www.cnc-keller.com
5. www.fanuc.eu
6. www.haascnc.com

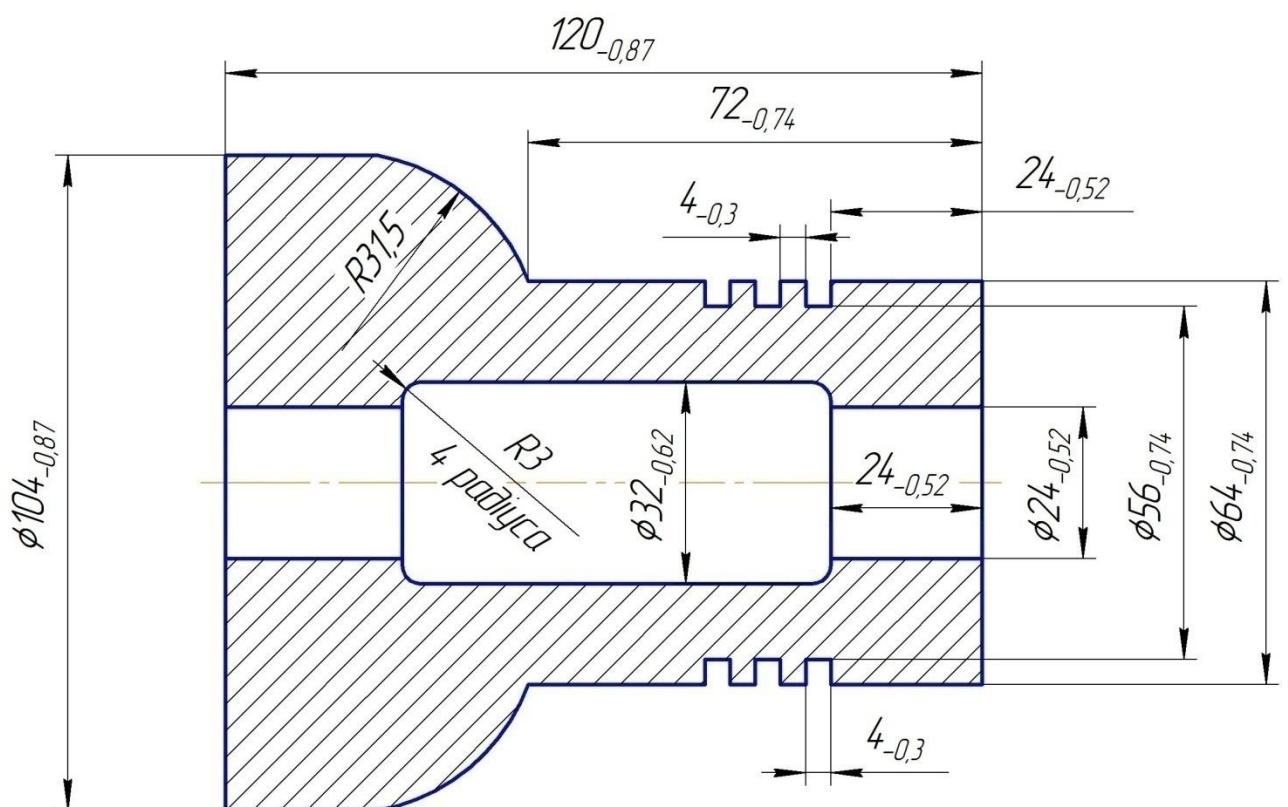
ДОДАТКИ

ДОДАТОК 1

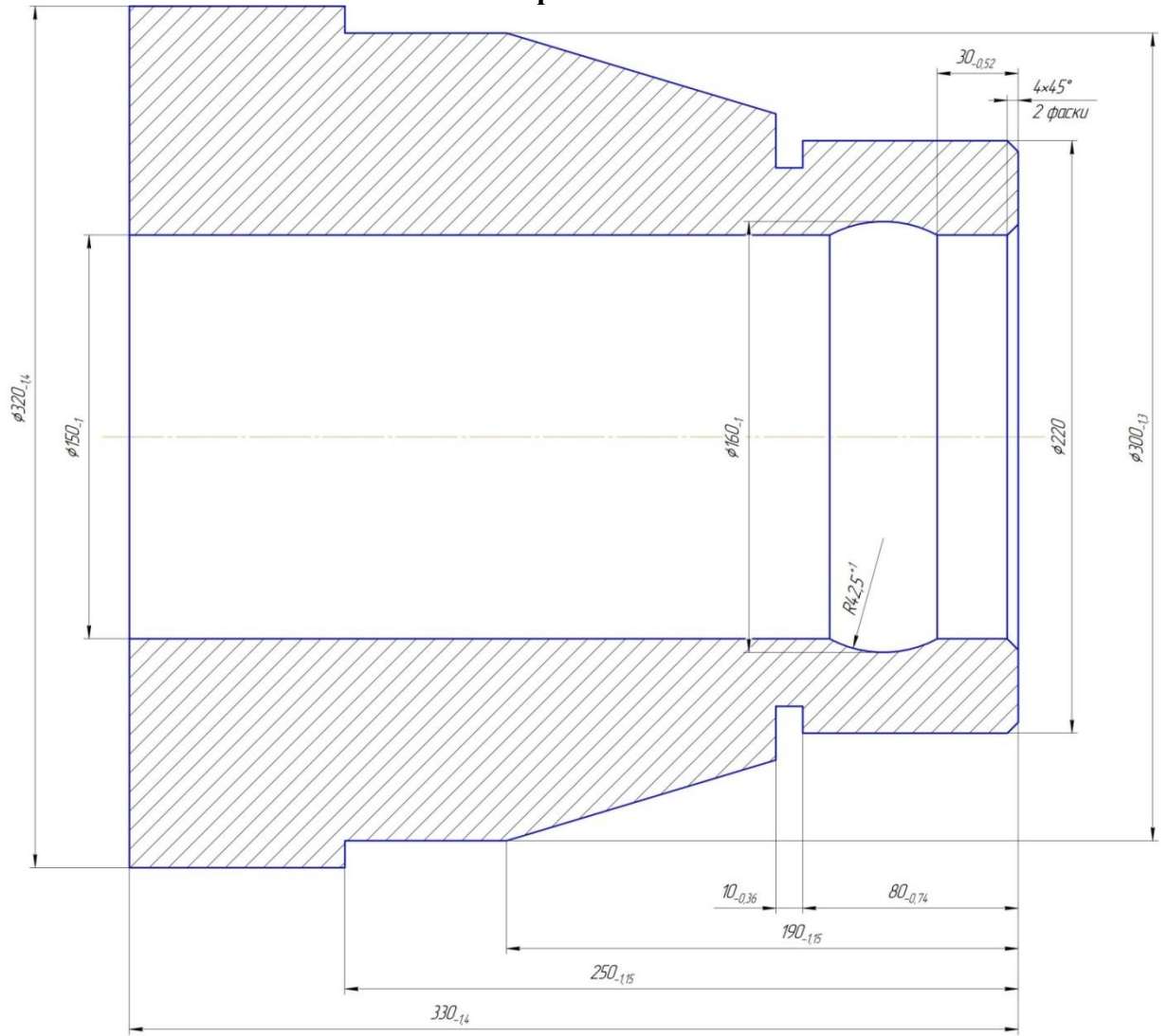
Варіант №1



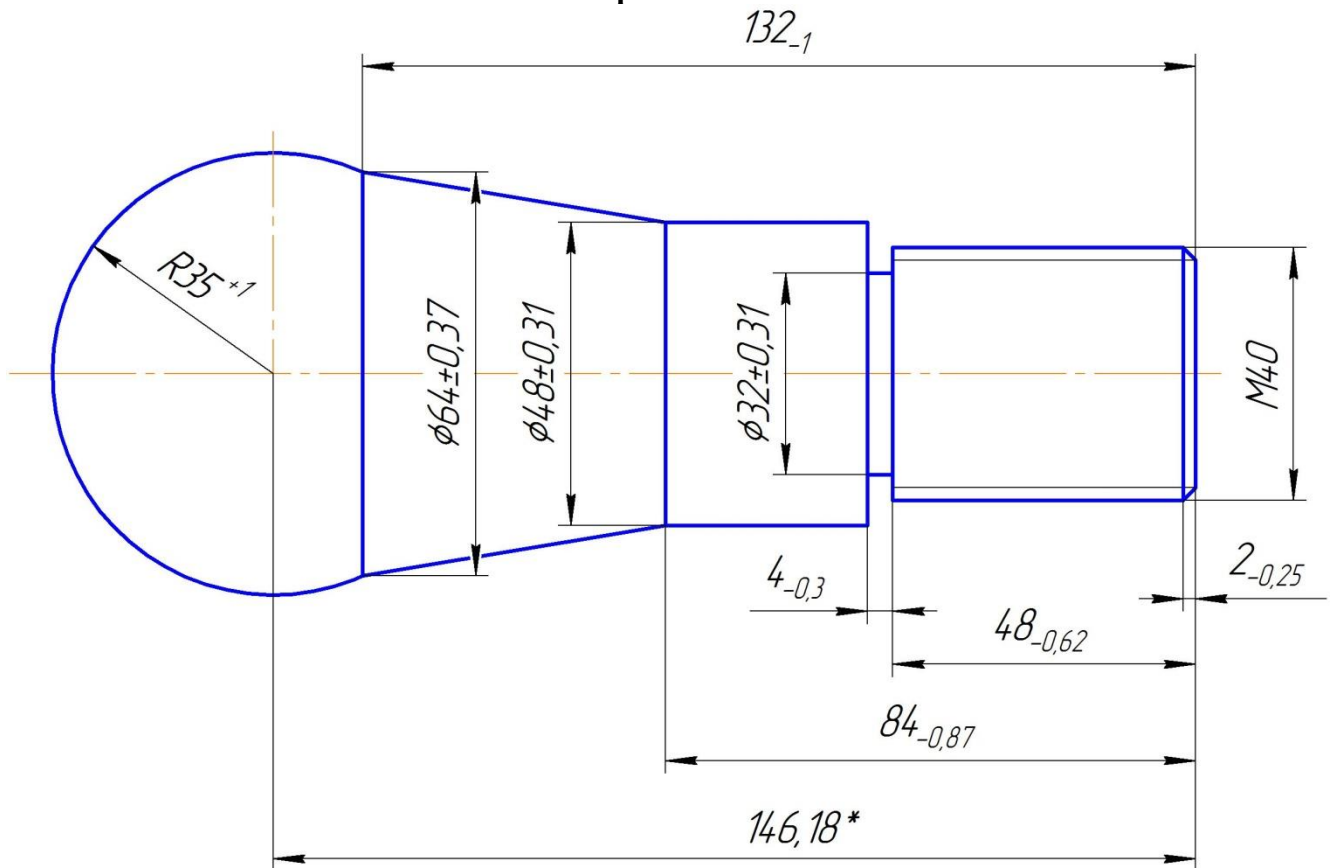
Варіант №2



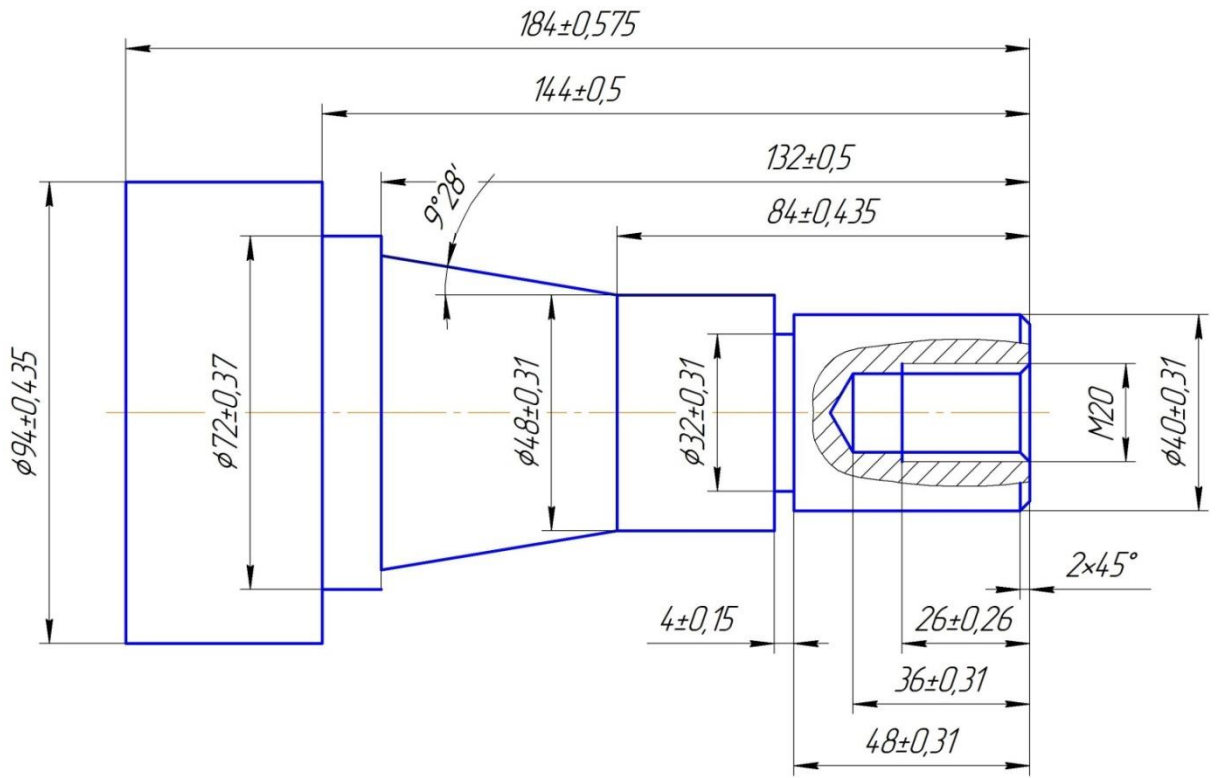
Вариант №3



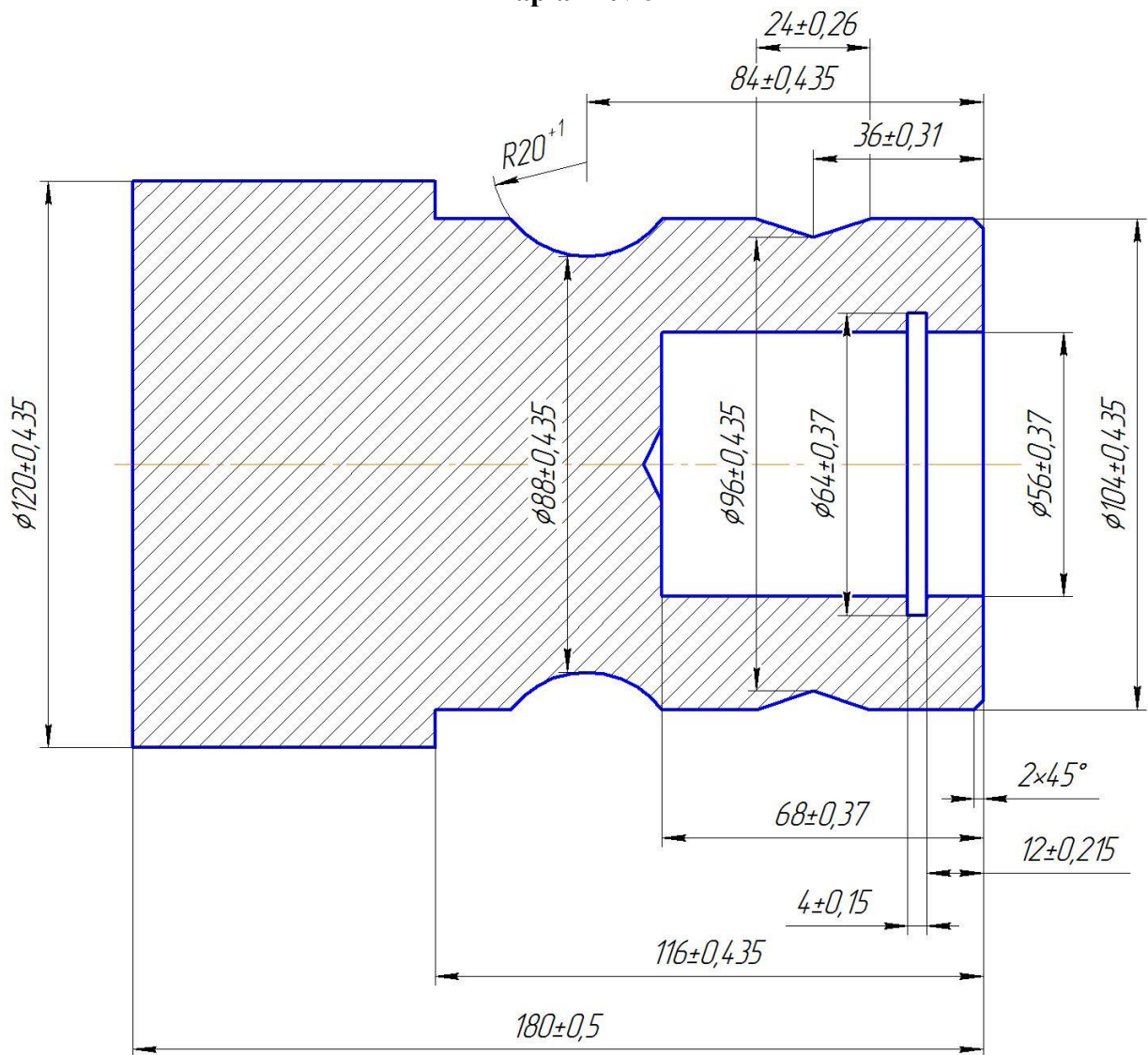
Вариант №4

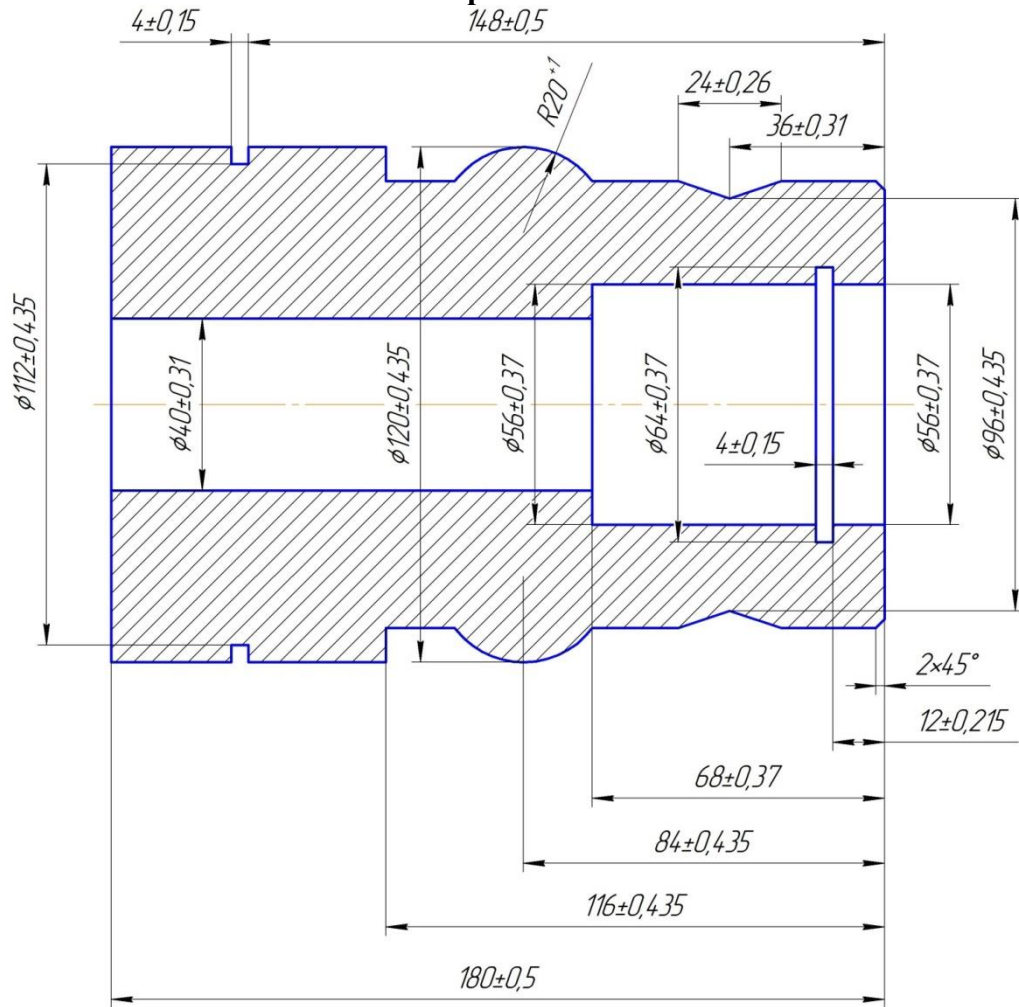
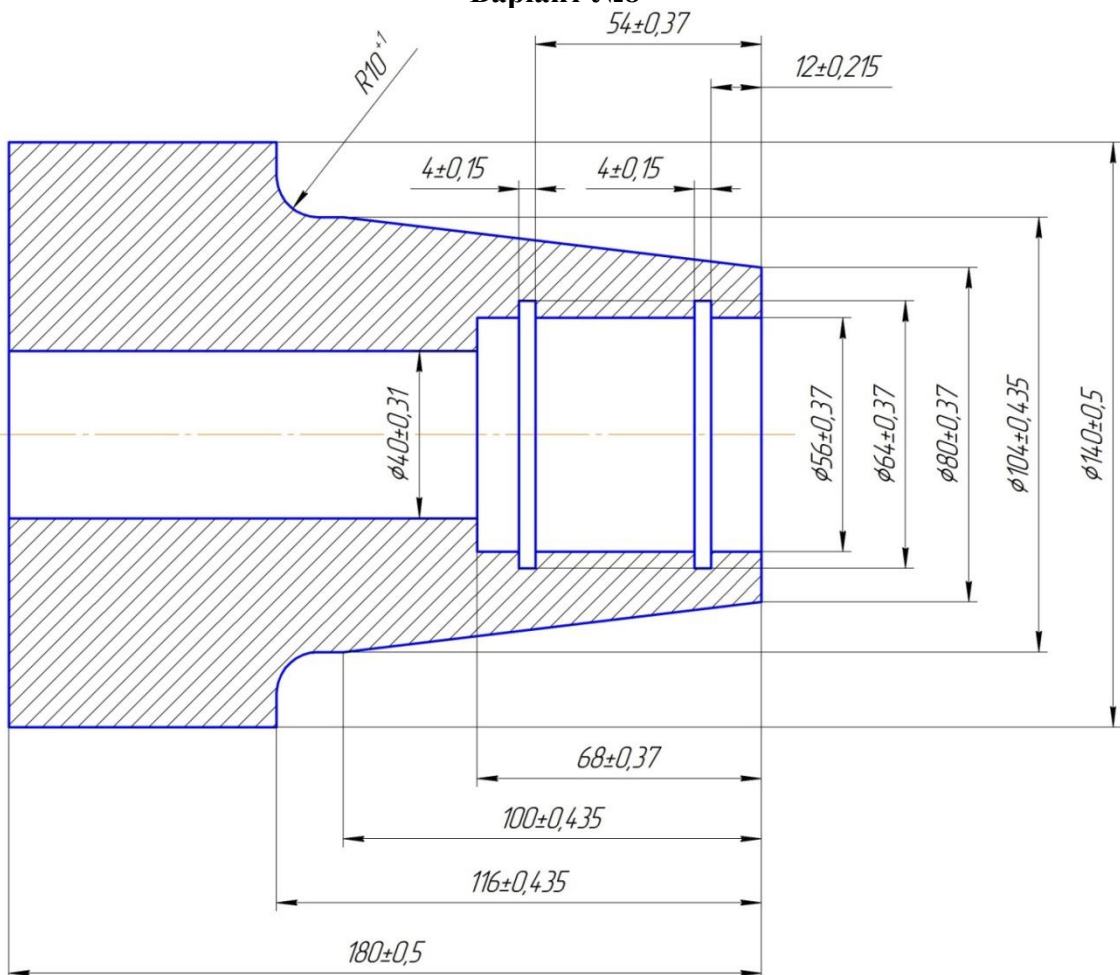


Вариант №5

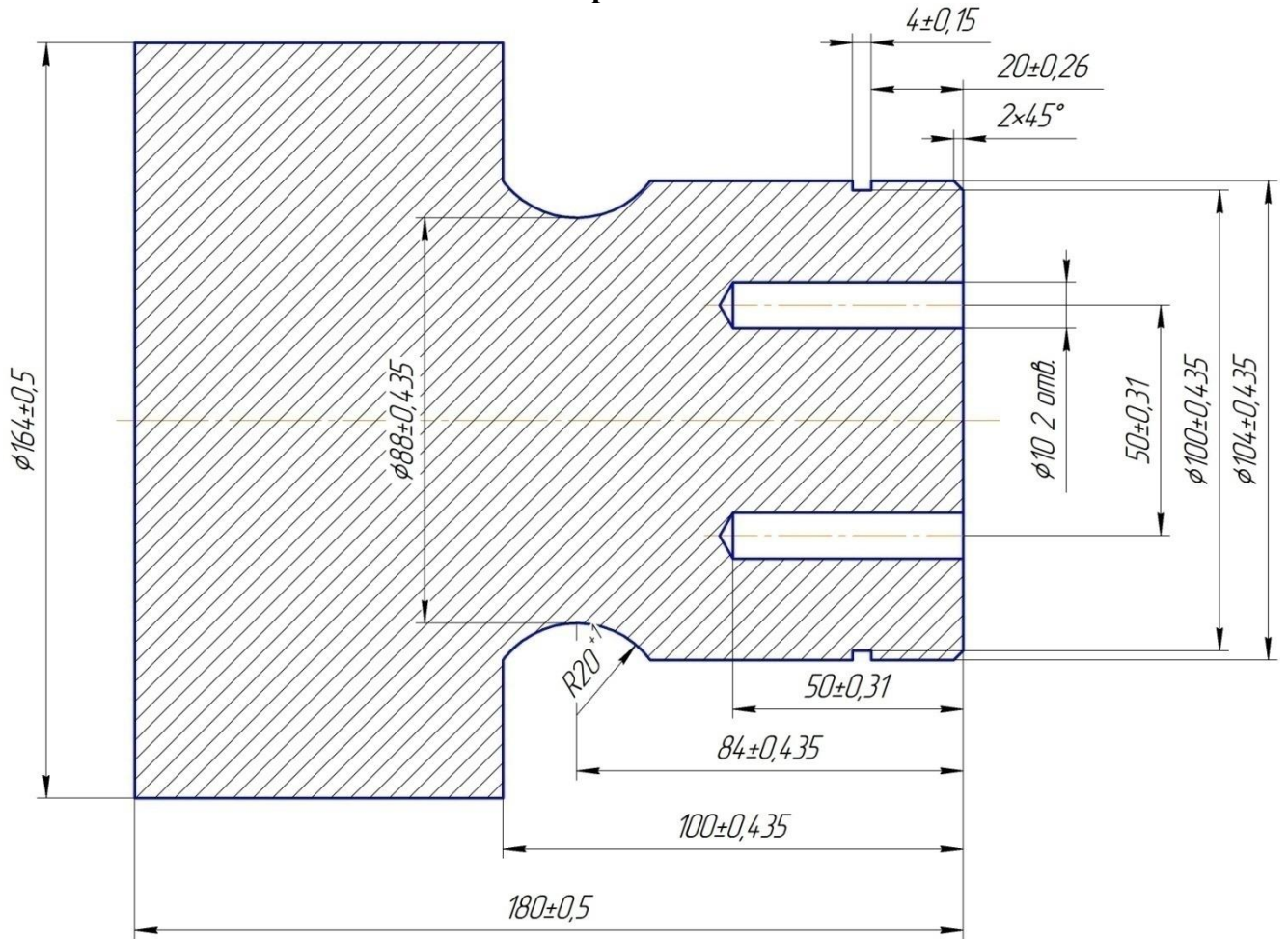


Вариант №6

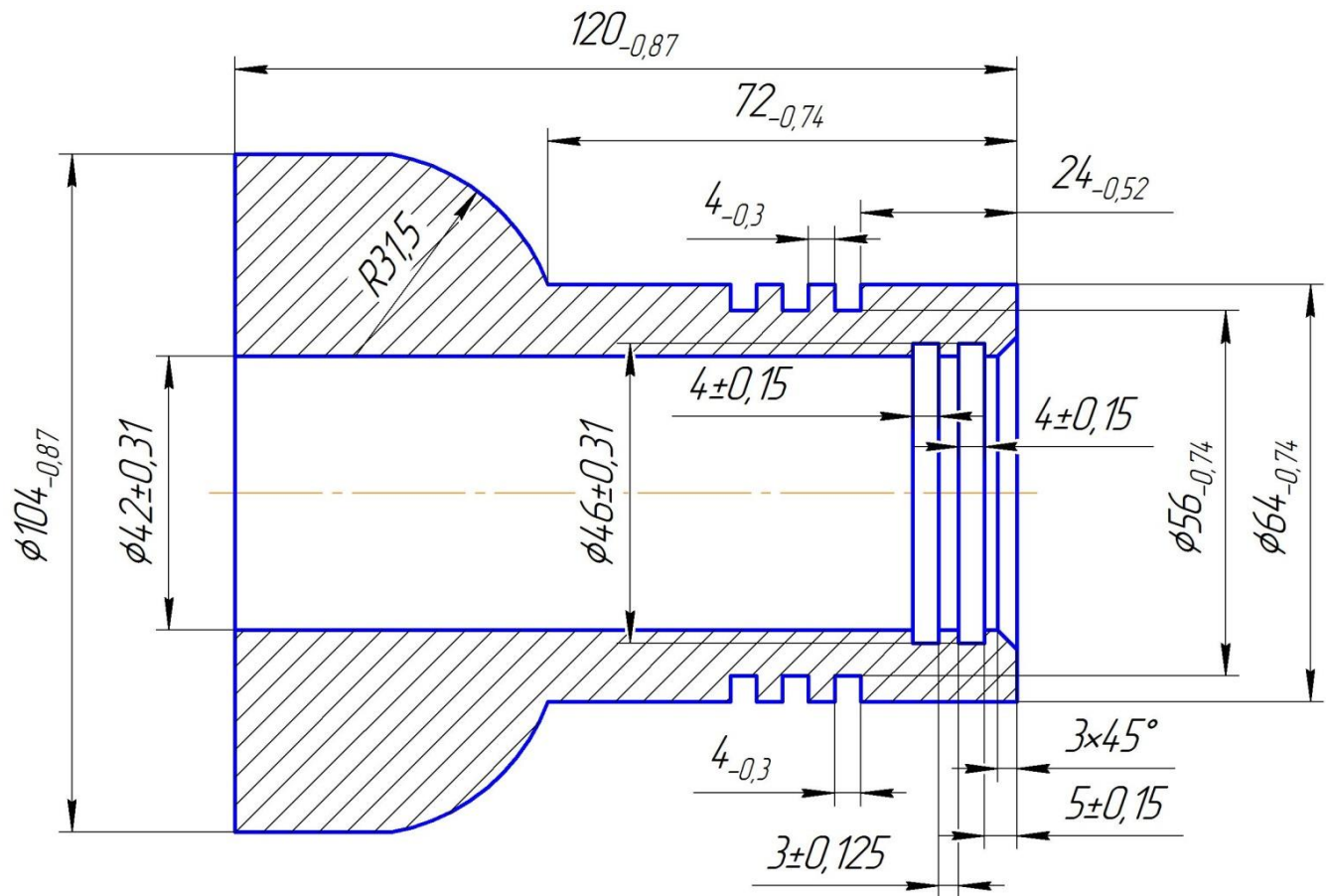


Вариант №7**Вариант №8**

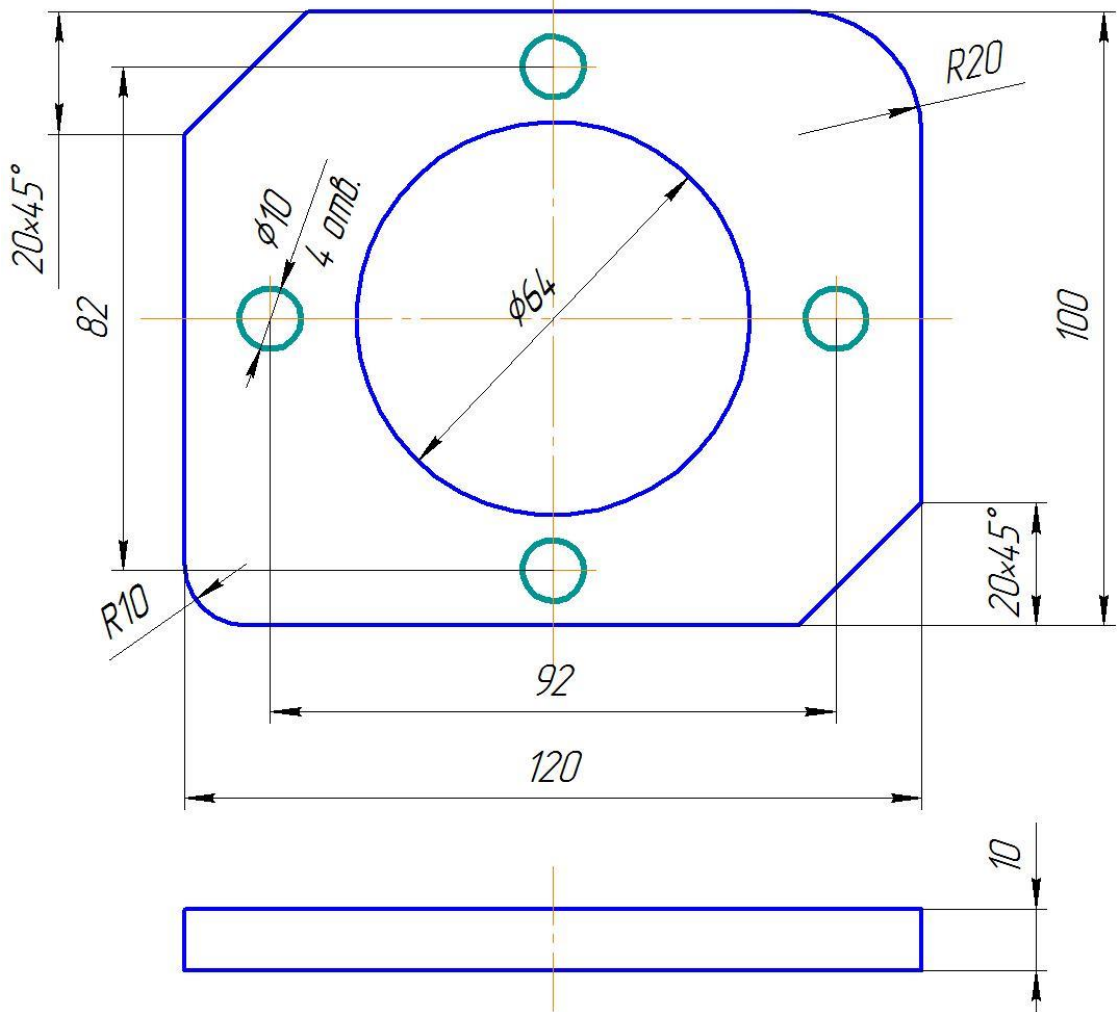
Вариант №9



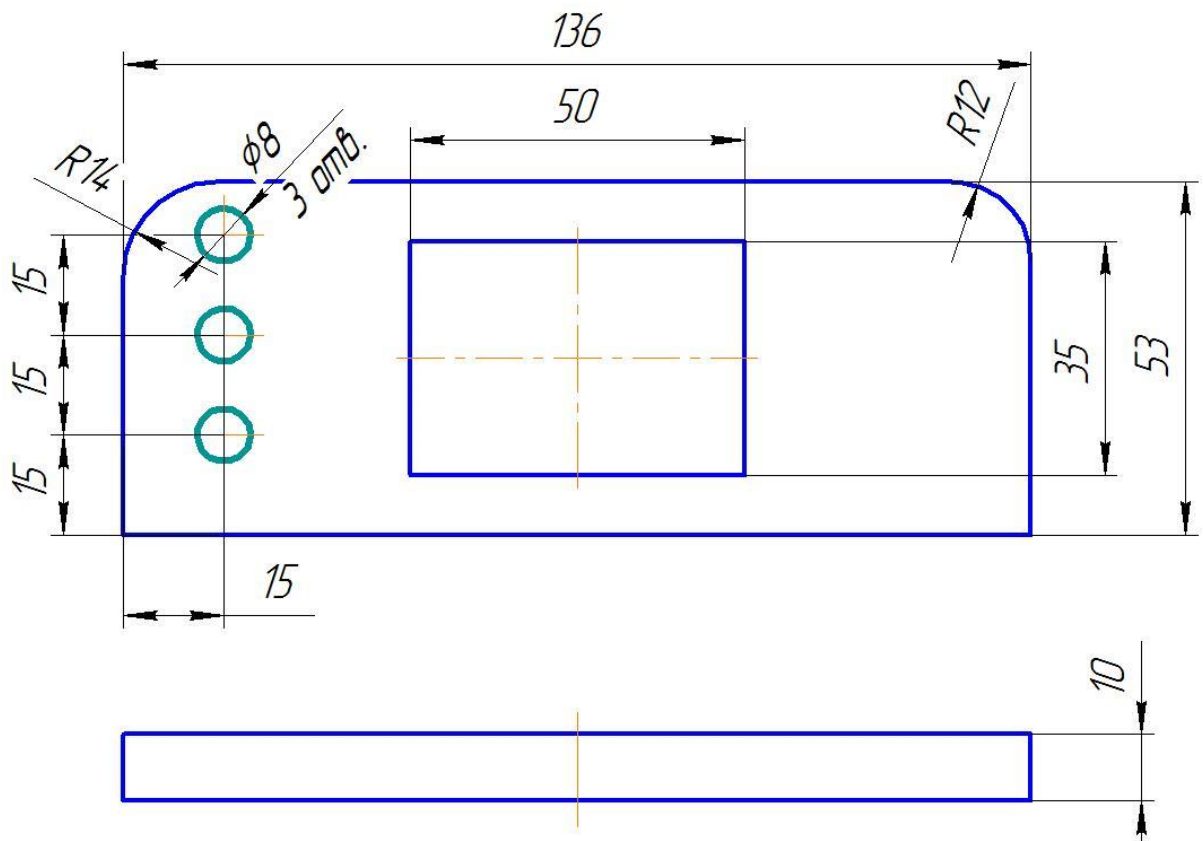
Вариант №10



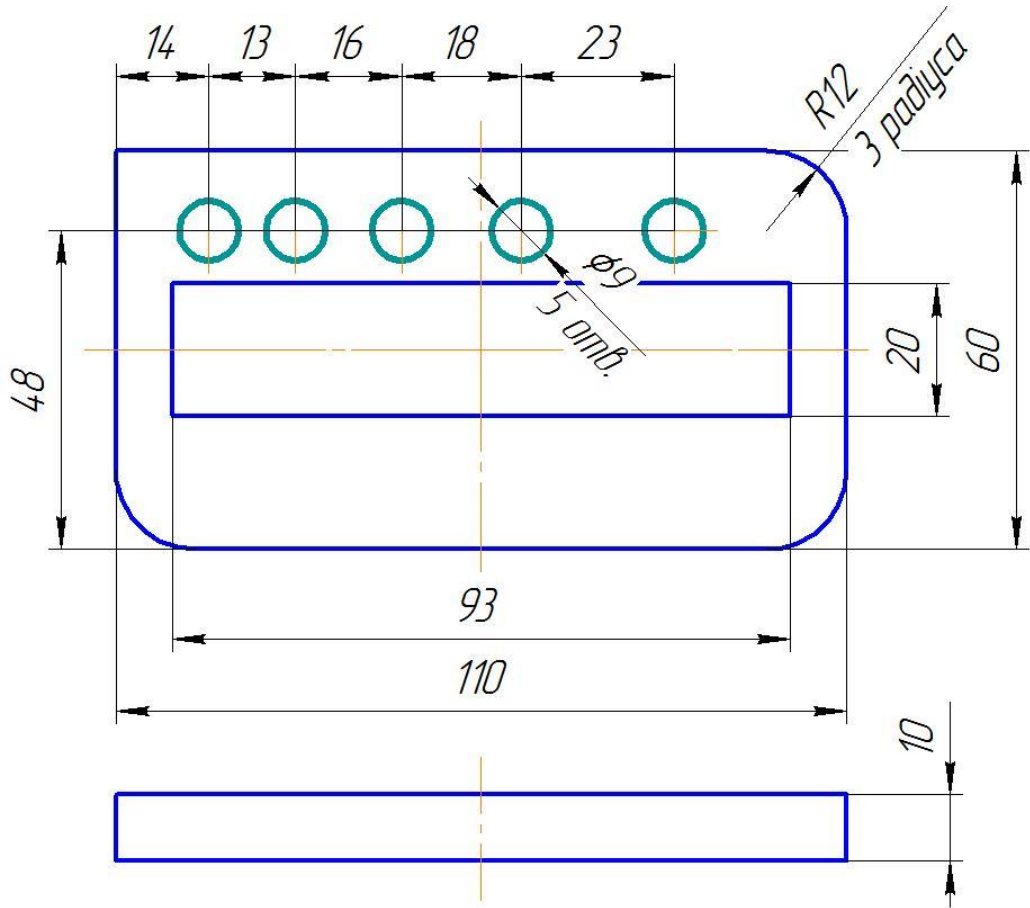
Варіант №1



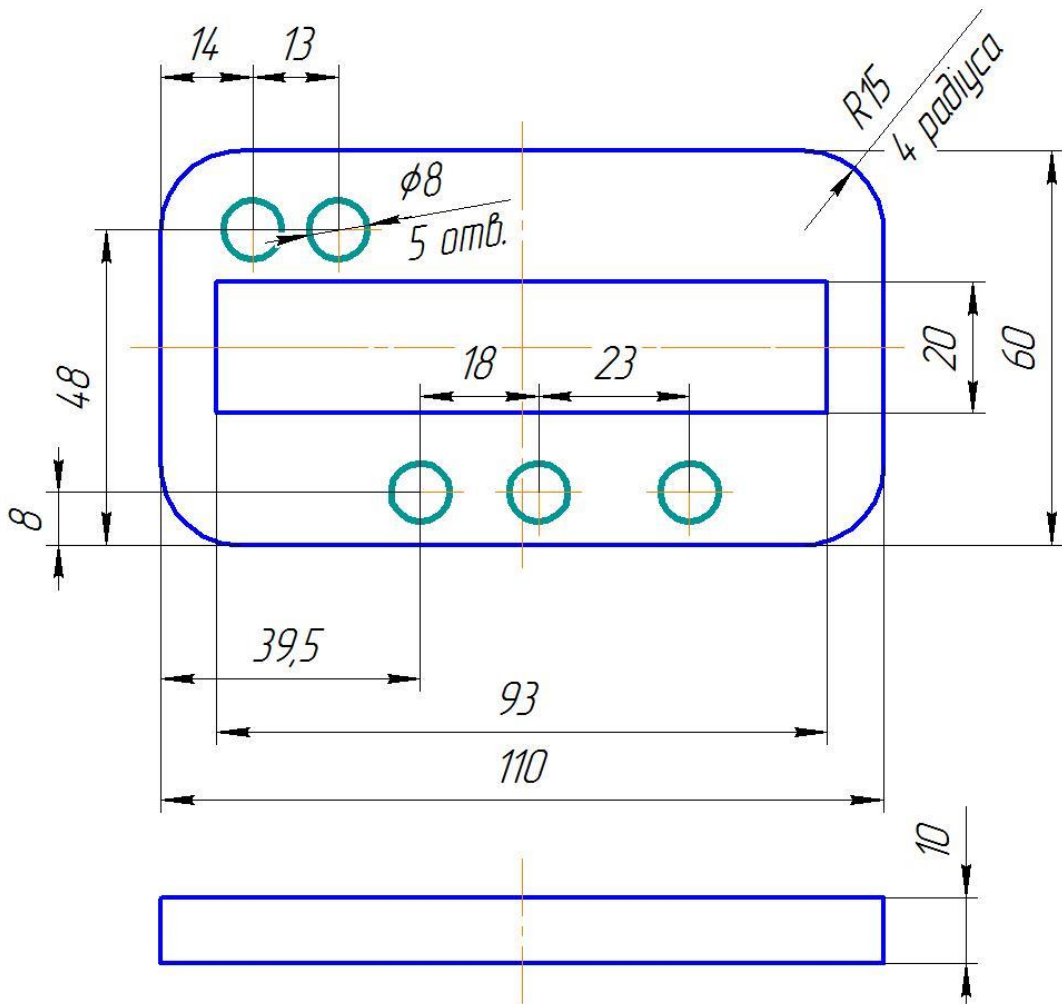
Варіант №2



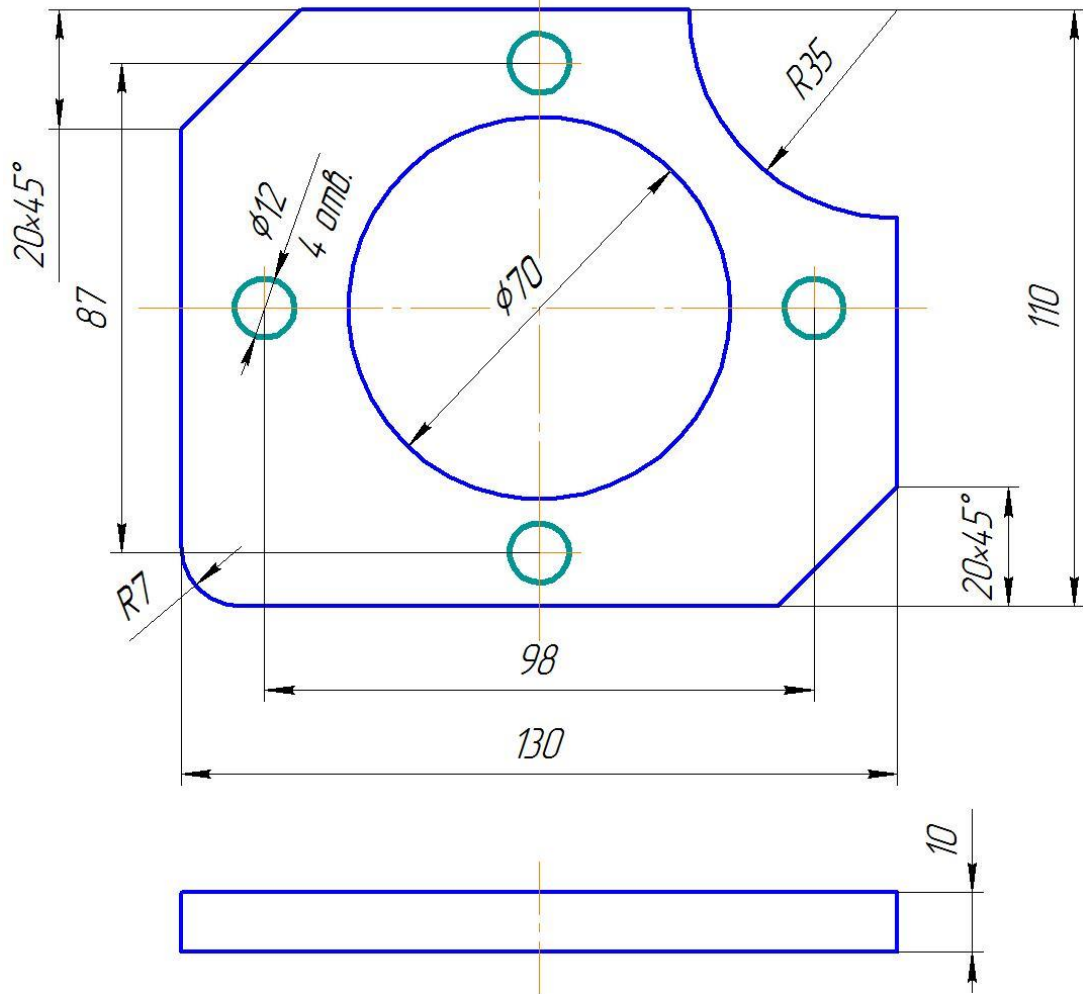
Варіант №3



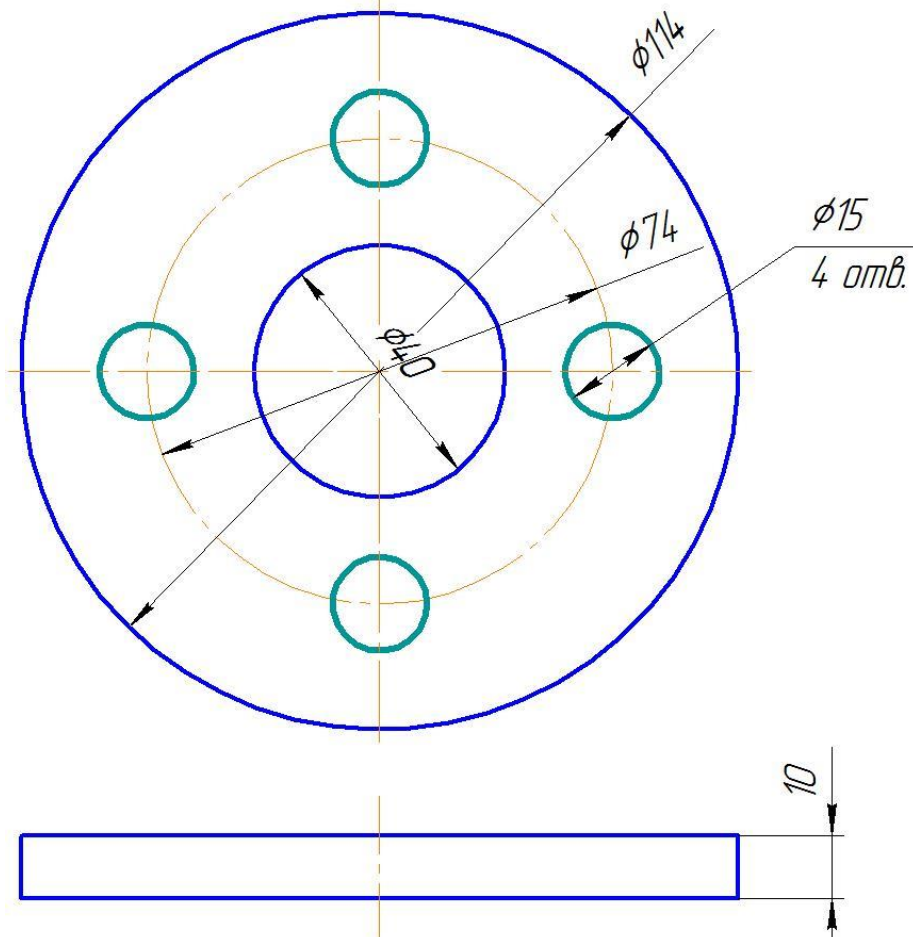
Варіант №4



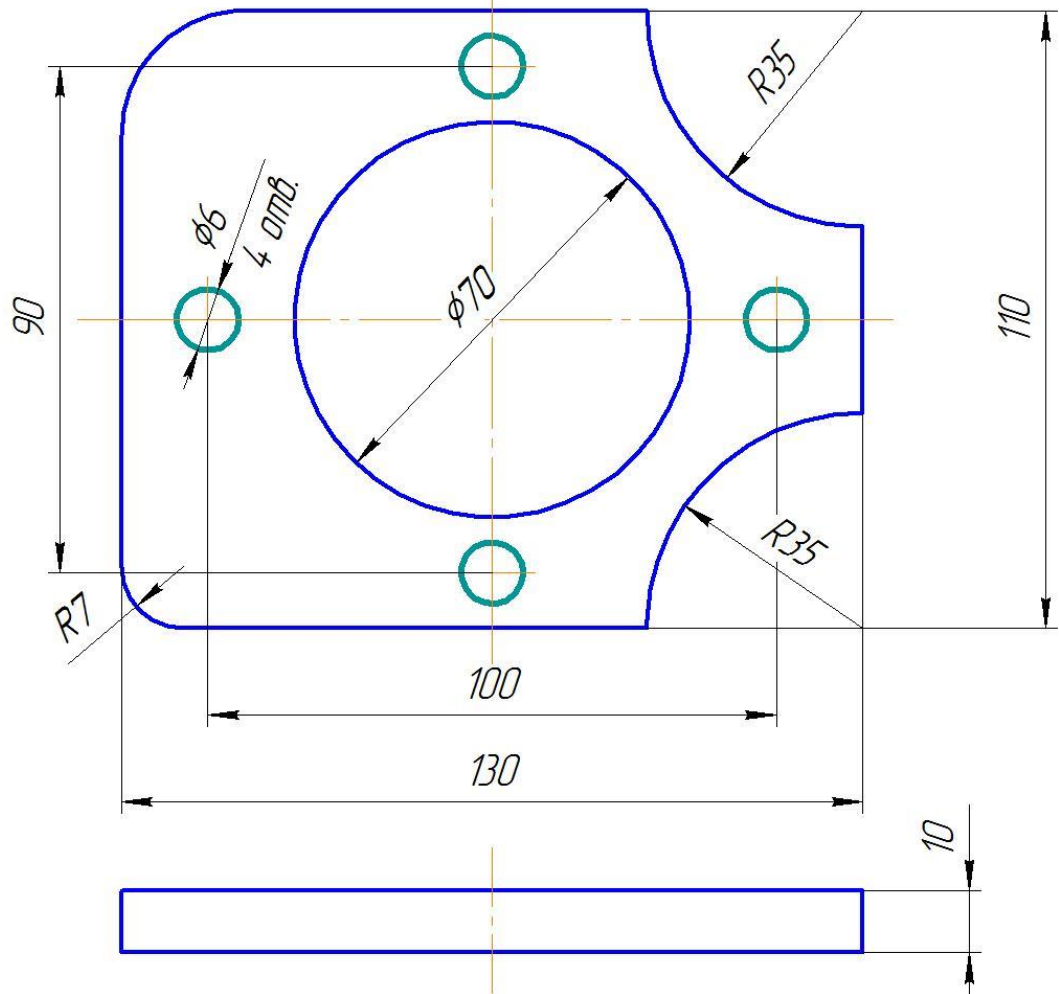
Вариант №5



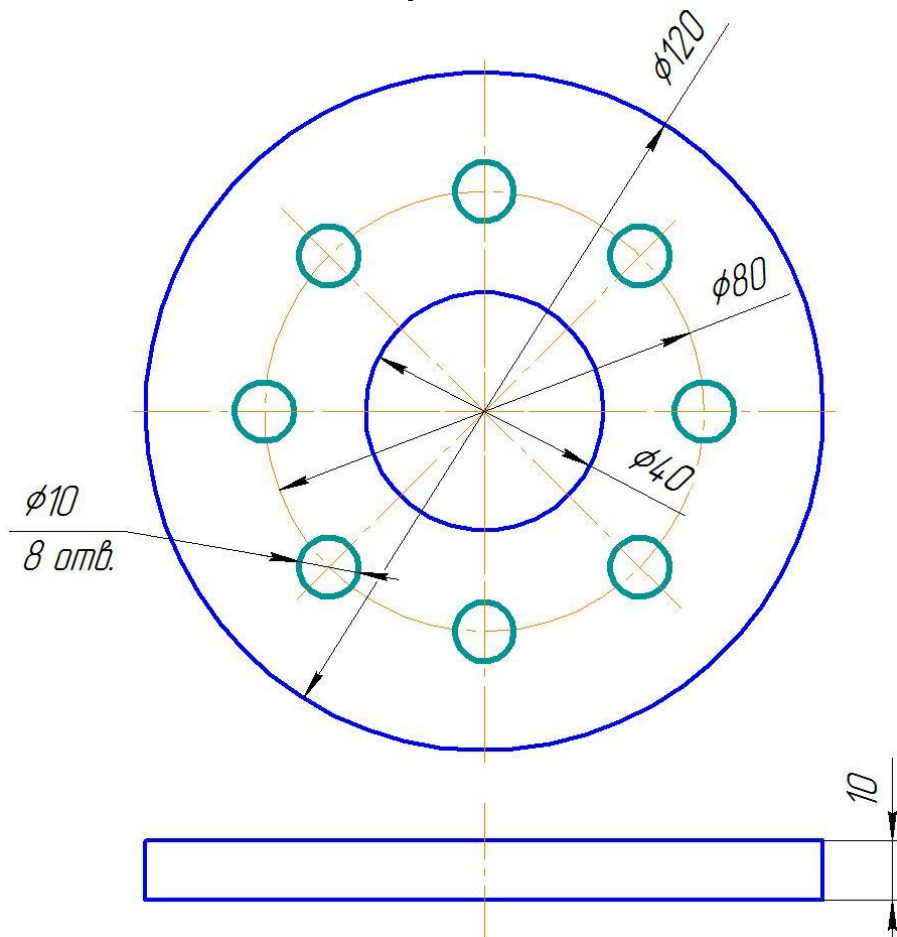
Вариант №6



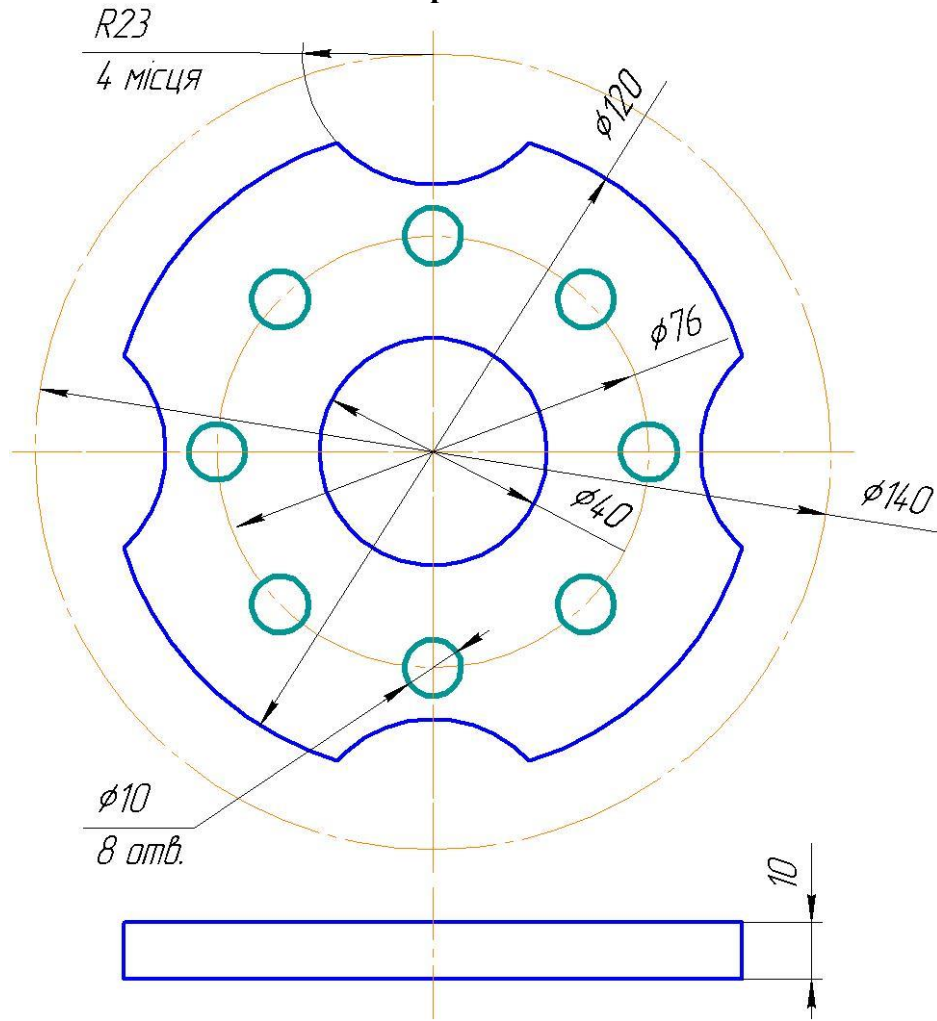
Вариант №7



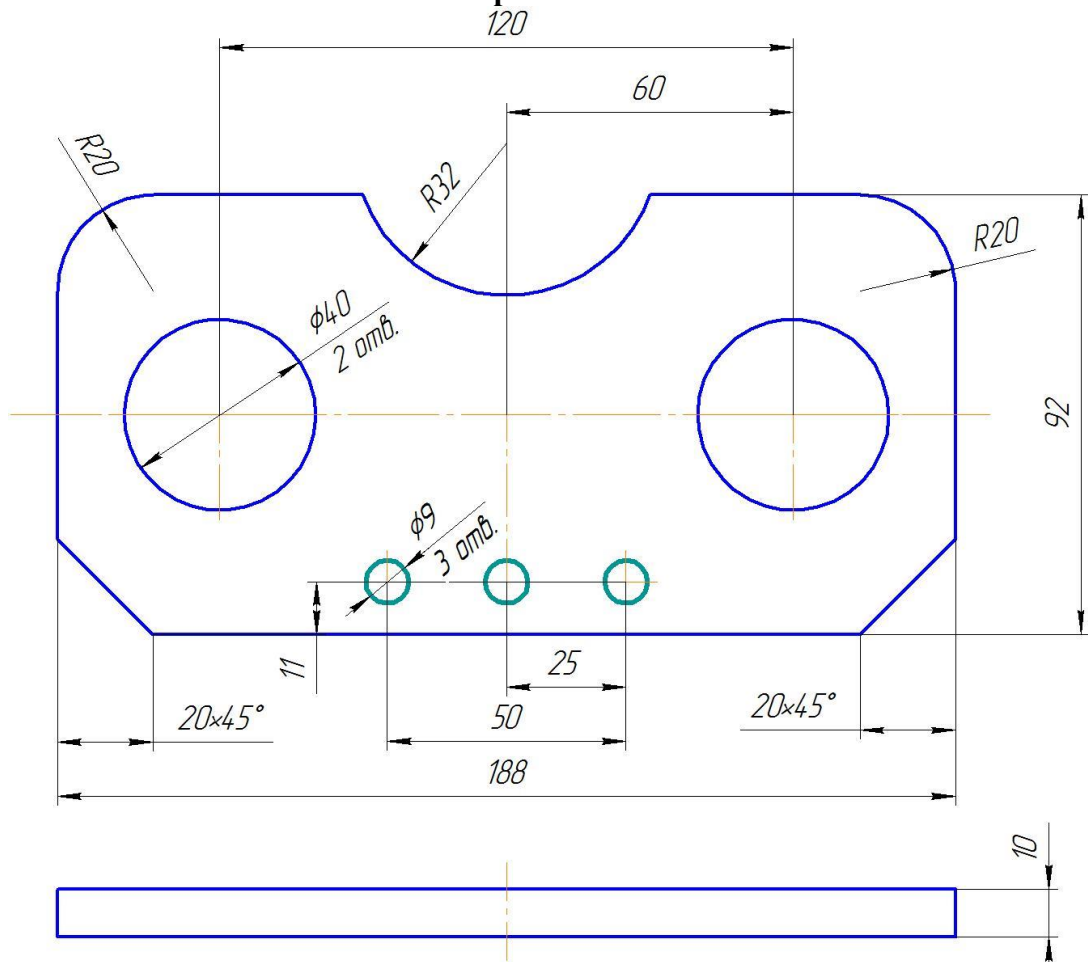
Вариант №8



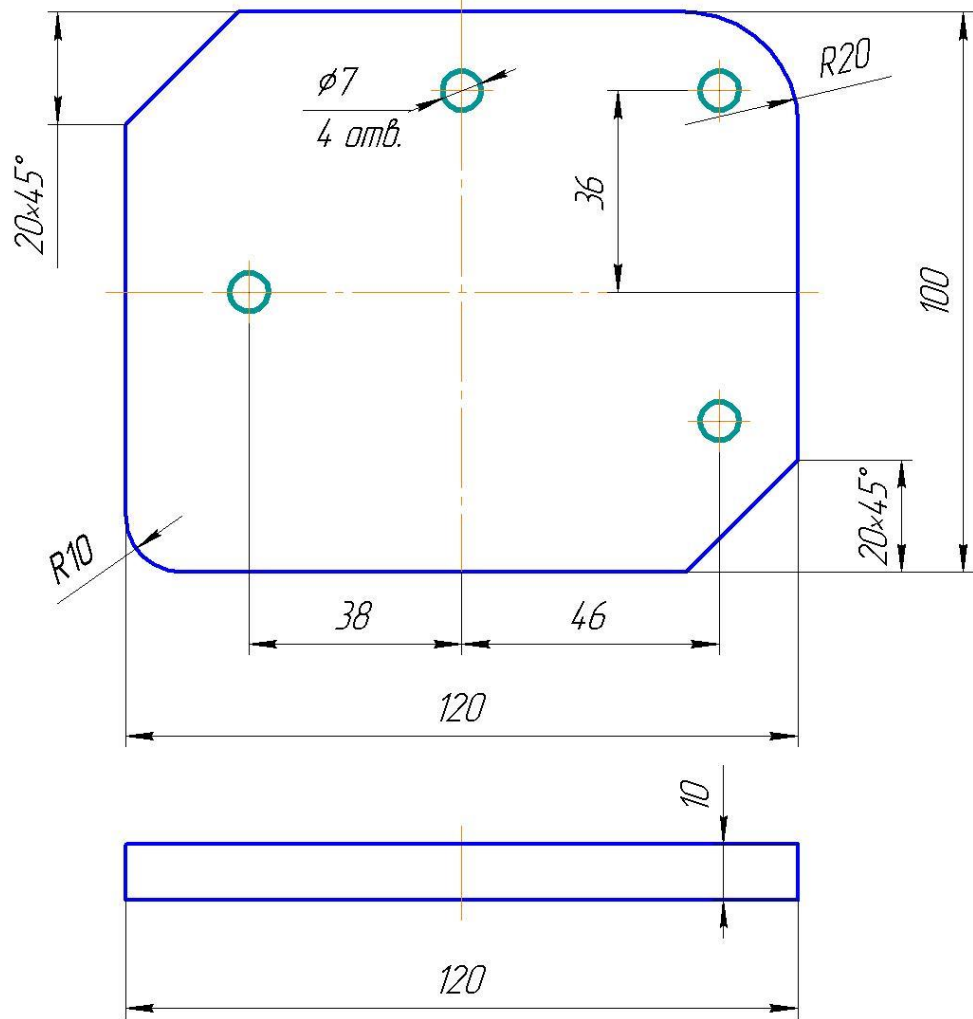
Варіант №9



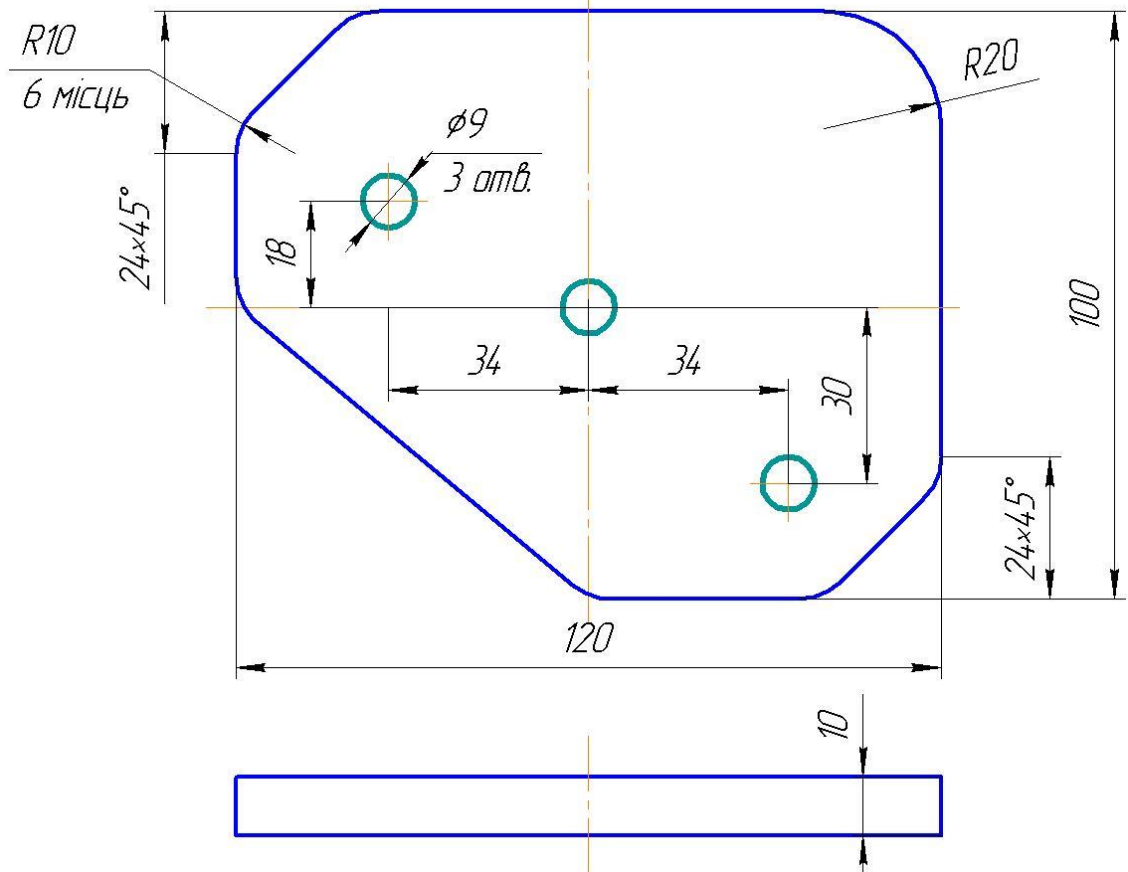
Варіант №10



Варіант №11



Варіант №12



Навчально-методичне видання
Методичні вказівки до виконання лабораторних робіт по курсу:
«Проектування технологічних процесів для верстатів з ЧПУ»
для студентів спеціальностей: 133 Галузеве машинобудування; 131 Прикладна механіка

Укладачі: Скібінський О.І., Селехова В.М.

м. Кропивницький

