

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Інститут післядипломної освіти

Кваліфікаційна робота бакалавра

на тему: Розробка PDM-модуля сполучення з інтегрованою САПР

Виконав студент групи К-5
спеціальність 122 Комп'ютерні науки
Бакулін Микола Станіславович

Керівник д. т. н., професор каф. АСМНСІ
Великодний Станіслав Сергійович

Консультант _____

Рецензент д. ф.-м. н., доцент каф. ВПМ
Буяджи Василь Володимирович

Одеса 2022

ЗМІСТ

Скорочення та умовні позначки	6
Вступ.....	8
1 Аналіз існуючих систем автоматизованого проектування і керування даними для механообробки.....	10
2 Проектування модуля постпроцесорної обробки	17
2.1 Концептуальна модель інтегрованої САПР ADEM 7.0	17
2.2 Етапи формування КП у системі ADEM 7.0	22
2.3 Завдання, які вирішуються адаптером	25
2.3.1 Визначення імені верстату	25
2.3.2 Визначення номера постпроцесору.....	26
2.3.3 Перетворення команд CLDATA в слова та кадри КП	26
2.3.4 Компонування кадрів КП	29
2.4 Етапи створення постпроцесору.....	30
2.4.1 Склад постпроцесору.....	30
2.4.2 Написання постпроцесора.....	32
2.4.3 Трансляція файлу алгоритму	36
2.4.4 Перегляд результатів роботи постпроцесора.....	37
2.4.5 Налагодження постпроцесора.....	37
3 Тестування сформованих структур і обробка результатів проектування	39
3.1 Формування файлу макрокоманд	39
3.2 Макет кадру та формат виводу	42
3.3 Формування вікон різних типів	43
3.4 Формування макету кадру.....	44
3.5 Формування файлу алгоритма	45
3.6 Арифметичні дії та функції в алгоритмах	46
3.7 Завантаження файлів алгоритмів.....	48
3.8 Приклад формування файлу алгоритмів.....	49
3.9 Налагодження алгоритмів	53

Висновки	56
Перелік джерел посилання	59
Додаток А Список систем і змінних користувача	62
Додаток Б Команди CLDATA.....	67
Додаток В Приклад паспорта верстата	68
Додаток Г Приклад файлу макрокоманд	71
Додаток Д Приклад макета кадру.....	72
Додаток Е Приклад файлу алгоритмів.....	74

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

БЦК – бінарний цифровий код.

ГІС – гнучка інтегрована система.

ДА – діаграма активності.

ДВВ – діаграма варіантів використання.

ДК – діаграма класів.

ДКМ – діаграма компонентів.

ДП – діаграма послідовності.

ДР – діаграма розміщення.

ДС – діаграма станів.

ЄСКД – єдина система конструкторської документації.

ЄСТД – єдина система технологічної документації.

ЗОР – змащувально-охолоджуючі рідини.

ІП – інваріантний постпроцесор.

ІСАПР – інтегрована система автоматизованого проектування.

ИНКАДР – ініціалізація номера кадру.

КП – керуюча програма.

НДІ – науково-дослідницький інститут.

ОР – охолоджуюча рідина.

ОФ – оціночна функція.

ПЕОМ – персональна електронна обчислювальна машина.

ПЗП – постійний запам'ятовуючий пристрій.

ПМ – програмний модуль.

ПНКАДР – придушення номеру кадру.

ПЩ – початку циклу.

СПВР – система програмного відтворення рухів.

СУБД – система управління базами даних

СЧПК – система числового програмного керування.

ЦДА – цифровий диференціальний аналізатор.

ЧПК – числове програмне керування.

2D – 2-dimensional.

3D – 3-dimensional.

ADEM – Automated Designing Engineered Manufacturing.

ASCII – American Standard Code for Information Interchange.

CAD – Computer-Aided Designing.

CAE – Computer-Aided Engineering.

CAM – Computer-Aided Manufacturing.

CAPP – Computer-Aided Process Planning.

CASE – Computer-Aided Software Engineering.

CLDATA – Cutter Location.

EIA – Energy Information Administration.

GPP – Generation of Process Planning.

ISO – International Organization for Standardization.

PDM – Product Data Management.

SQL – Structured Query Language.

ВСТУП

У складних реаліях існування сучасної промисловості України, важливим напрямом науково-технічного розвитку, є розробка нових технологій автоматизованого проектування з підвищеними вимогами щодо ступеня складності проєктованих конструкцій і скорочення термінів виконання замовлень.

Незважаючи на серйозний тиск кризових явищ в економіці та в умовах посилення конкурентної боротьби на ринку, ці складові змушують здійснювати роботи на найсучаснішому технічному рівні. Тому сьогодні конструювання, аналіз і технологічна підготовка виробництва виконується з використанням інтегрованих систем автоматизованого проектування (ІСАПР), вибір і організація роботи яких проходить в рамках єдиної концепції розвитку засобів проектування та технологічної підготовки виробництва, здатних вирішувати більш складні завдання, поставлені перед виконавцем. На можливості вирішення таких завдань і спрямована розглянута в дипломній роботі ІСАПР ADEM (Automated Designing Engineered Manufacturing), вигідно відрізняється від аналогів можливістю підключення нових (індивідуально розроблених під конкретні вимоги) модулів керування виробничими даними (PDM – Product Data Management) і бібліотек.

Мета роботи – створити PDM-модуль постпроцесорної обробки, що сполучується із ІСАПР ADEM.

Для досягнення встановленої мети, необхідно вирішити низку встановлених завдань:

- виконати огляд систем підготовки керуючих програм (КП), для встановлення їх недоліків та переваг, а також схожих рис, що можуть бути використані для створення нового модуля;
- сформулювати математичну модель механообробки, необхідну для виконання фізичних розрахунків, закладених в основу модуля;
- розібрати етапи створення постпроцесора, з обов'язковою процедурою трансляції файлів та переглядом результатів проєктування;

- створення паспортів верстатів, файлів макрокоманд, макетів кадрів, алгоритмів команд.

Об'єкт роботи – процес створення КП, за допомогою якої виконується механічна обробка на верстатах із ЧПК.

Предмет роботи – програмний модуль проектування КП, що сполучується із ICAPP ADEM.

Методи дослідження: методи аналітичної геометрії і обчислювальної математики для забезпечення розрахунків фізичних параметрів механообробки (різання і точіння) тіл обертання, методи і принципи автоматизованого проектування, методи організації графічного діалогу та побудови інтерфейсу користувача з використанням стандартів управління ANSI і ISO.

Заплановано практичне значення отриманих результатів. Результати дипломної роботи буде реалізовано у вигляді підсистеми, що створює основу для проектування КП для механічної обробки матеріалів на верстатах з ЧПК.

Буде створено графічний модуль взаємодії оператора верстата з генератором постпроцесорів, з можливістю відображення макрокоманд і переглядом результатів трансляції. Дана система дозволить оператору провести контроль кадрів КП без відпрацювання програми на верстаті. Також даний модуль дозволить оператору виконувати налагодження постпроцесора і формувати паспорт верстата і редагувати параметри КП.

Розвиток систем проектування КП в системах ЧПК, має важливе економічне значення в галузі обробки машинобудівної продукції, як засіб зменшення кількості браку, внаслідок виявлення «простих» помилок автоматичного програмування КП, на етапах налагодження і запуску перших зразків серійної продукції.

При практичному застосуванні програмного модуля (ПМ), вихідні дані можуть бути задані в результаті машинної обробки ADEM-CAM-файлу або сформовані інженером-технологом аналітично. Додатково в ПМ, передбачені необхідні коментарі щодо коректності формування макета кадру і формату виведення КП.

1 АНАЛІЗ ІСНУЮЧИХ СИСТЕМ АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ І КЕРУВАННЯ ДАНИМИ ДЛЯ МЕХАНООБРОБКИ

При написанні даного розділу, перш за все, були піддані аналізу методи та засоби автоматизованого проектування і підготовки КП для СПВД, що використовуються на профільних підприємствах. Це пакети і системи від провідних наукових інститутів країни і західних розробників.

Зупинімося на аналізі пакетів, функції яких найбільш схожі з функціями ADEM. Одним з перших проаналізованих аналогів стала система автоматизованого проектування управляючих програм для верстатів з ЧПК «САПР-ЧПК». Це Win32-додаток з простим і гнучким інтерфейсом, високою швидкістю розрахунку керуючих програм і розвиненими засобами верифікації траєкторії руху інструменту. Технологи високо оцінили Верифікатор G-кодів, відладчик постпроцесорів PostWizard, що входять до складу САПР-ЧПК, а також простоту її установки, навчання та використання [1]. Починаючи з 1989 р., система САПР-ЧПК різних версій впроваджена на більш ніж 300 підприємствах Росії і країн СНД.

Інтегрована оболонка позбавляє технолога від необхідності вивчати апаратне і програмне забезпечення ПЕОМ, тим самим прискорюючи процес введення САПР-ЧПК в експлуатацію і практично позбавляючи технологічні служби підприємств від послуг системних програмістів [2].

Інтегрована оболонка має розвинуті засоби для налаштувань численних параметрів САПР-ЧПК індивідуально для кожного користувача, відновлюючи значення при повторних сеансах [3]. Оболонка концентрує управління ви-кликами всіх модулів САПР-ЧПК від розрахунку контуру деталі та каркасного моделювання (рис. 1.1) до формування КП.

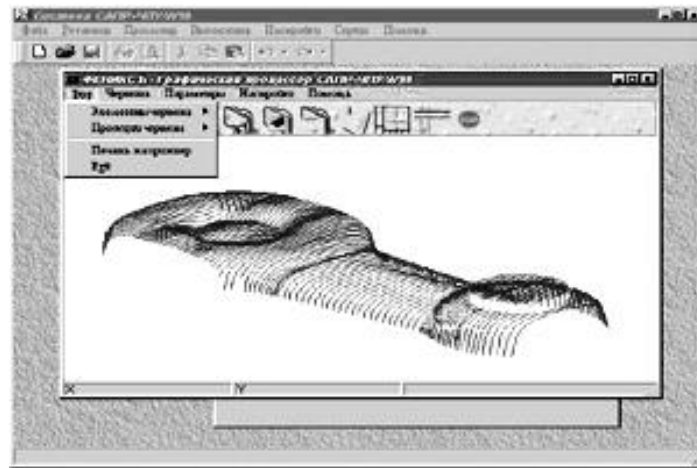


Рисунок 1.1 – Фрагмент каркасного моделирования 3D-обработки,
виконаного в «САПР-ЧПК»

В САПР-ЧПК при програмуванні обробки точково-заданих поверхонь, які задані перерізами, введений метод лінійно-кругової інтерполяції, що дозволяє автоматично розрахувати по масиву характерних точок (що задаються технологом) – гладкий контур, що складається з дуг окружностей і відрізків, що призводить до економії довжини перфострічки (в порівнянні з лінійною інтерполяцією), а для стійок управління верстатами типу Н-33, є необхідною умовою для використання коректорів на радіус інструменту [4].

У системі «САПР-ЧПК», є вбудований модуль препроцесора, призначений для трансляції вихідної програми у так званий файл контуру деталі і технологічних команд. У процесі трансляції, в першу чергу, обчислюються канонічні параметри всіх геометричних елементів і ідентифікаторів, а потім обробляються оператори, складові тіло процедури (рядки обходу). Крім основних операторів (рух за елементами геометрії, підйом/опускання по Z), препроцесор інтерпретує управляючі оператори (обробка ділянок, оператори циклів), а також макропроцедури, виклики автоциклов, CNC-підпрограми і технологічні команди [5].

В САПР-ЧПК є спеціальний модуль, призначений для спрощення створення КП обробки для токарних верстатів з ЧПК «Електроніка НЦ-31». З його допомогою КП роздруковується у вигляді лістингу, зручного для подаль-

шого ручного введення верстатником в пам'ять стійки з ЧПК. Крім того, модуль дозволяє технологу використовувати всі оригінальні можливості ручного програмування: зміна типу розрахунку переміщення, зсув контуру, організація і виклик підпрограм, звернення до постійних циклам [6].

Модуль «Графічний Процесор» «Фенікс» включений до складу базового комплексу системи САПР-ЧПК. Модуль «Фенікс», використовується для графічної верифікації управляючих програм, тобто для графічного представлення контуру деталі, елементів роз'єднаної геометрії, а також імітації переміщень інструменту в процесі обробки деталі [7].

У інтегровану оболонку системи САПР-ЧПК/2005 вбудований «Коректор керуючих програм». Його призначення полягає у введенні з носія раніше підготовленої КП, її перекодування з інформаційного коду верстата (ISO, EIA) у текстовий вигляд, подальша коректування за допомогою екранного редактора і, нарешті, зворотна операція виведення на носій в необхідному коді системи з ЧПК [8].

Коректор має велике число настройок. При введенні з перфострічки можлива фільтрація символів протягання, видалення міжкадрових протяжок, а при виведенні на неї можна вибрати один з чотирьох інформаційних кодів верстата, визначити величини старт/стопних протяжок, задати число протягань між кадрами керуючої програми [9].

ARTIPP (універсальний постпроцесор) – дозволяє використовувати єдину бібліотеку з 300 постпроцесорів для будь-якої із систем – ADEM, CATIA, UG, Pro/Engineer, Duct, PowerMill, ProCAM, CADDs, Euclid, Anvill, КРЕДО, ГЕММА-3D, САП-УФА, АРТ, БАПТ, MODART [10]. ARTIPP генерує керуючу програму на основі CL-файлів, сформованих CAD/CAM-системами [11]. За допомогою ARTIPP технолог може розробляти постпроцесори для будь-якого комплексу «верстат-система з ЧПК» [12].

«Інваріантний постпроцесор» (ІП) – унікальний програмний продукт, розроблений в 1975 р. і який пройшов довгий шлях еволюції, до цих пір не має повних аналогів у СНД і за кордоном. У складі системи САП-СМ4 він

впроваджений на 500 підприємствах колишнього СРСР, а в складі САПР-ЧПК – на 220 підприємствах Росії і СНД [13]. Модуль ІІІ призначений для перетворення файлу траєкторії руху інструменту і техкоманд в файл керуючої програми, адаптованої до конкретного комплексу «верстат – система ЧПК».

На відміну від традиційних методів постпроцесування, ІІІ може бути налаштований на будь-який з мов системи ЧПК. Налаштування модуля полягає у створенні нової або використанні готової анкети (паспорта), що описує характеристики конкретного комплексу «верстат-система ЧПК» [14].

Вхідною інформацією для модуля ІІІ є:

- файл траєкторії руху інструменту і техкоманд;
- файл паспорта;
- файл макропроцедури.

Вихідною інформацією є текстовий файл КП, який може бути виведений на перфострічку, записаний на магнітний диск, флеш-карту або переданий у пам'ять стійки через лінію прямого управління.

ІІІ є гарним доповненням до будь-якої імпоротної і вітчизняної CAD/CAM-системи, так як вже має готові паспорти практично на будь-ЧПК-обладнання. Однак, якщо ж обладнання унікальне, то використання ІІІ не дає значний виграш у вартості, терміни створення і зручність модифікації паспортів у порівнянні з технологіями індивідуального постпроцесування або генерування постпроцесорів.

Відмітною принципом GrafCAM є автоматичне формування тексту програм на вхідній мові своєї базової системи, тобто САПР-ЧПК. Крім того, GrafCAM є 100% графічним інтерпретатором і відладчиком текстів програм, створених за допомогою систем САП-32, САП-ЄС, САП-СМ4, САП-ПК, ACAD-SAP, що дозволяє забезпечувати режим наскрізного проектування за допомогою простого макромови.

Пакет макропроцедури «евольвенти» призначений для розрахунку КП для верстатів з ЧПК токарної, фрезерної (з використанням корекції на радіус

інструменту) і електроерозійної груп, верстатах з лазерною різкою для виготовлення прямозубих циліндричних зубчастих коліс зовнішнього і внутрішнього зачеплення з евольвентним профілем зуба [15].

На верстатах токарної групи можлива обробка кінцевого фасонного ріжучого інструменту з евольвентним профілем для подальшого виготовлення конкретного зубчастого колеса на універсальному обладнанні [16]. У токарній обробці передбачена можливість автоматичної вибірки металу за схемою «Петля» з програмованим припуском під чистову обробку [17].

Профіль розглянутого контуру зуба являє собою масив точок, що належать евольвенті, і може бути оброблений сплайнами системи САПР-ЧПК з лінійною або лінійно-круговою інтерполяцією (рис. 1.2).

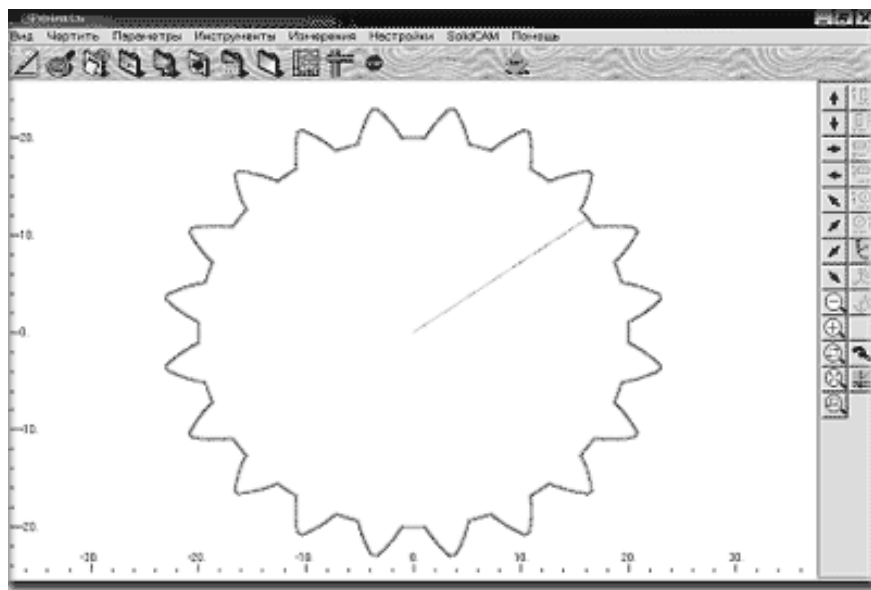


Рисунок 1.2 – Профіль зубчастого колеса з евольвентним профілем зуба, що інтерполюється за допомогою пакету «Евольвента»

Сплайн з лінійною інтерполяцією апроксимує масив точок ламаним лініями з задається ступенем точності [18]. При необхідності, постпроцесор може плавно з'єднувати отримані відрізки дугами кіл для забезпечення роботи з корекцією радіусу інструменту [19]. Сплайн з лінійно-круговою інтерпо-

ляцією з масиву точок евольвенти формує кадри КП (з плавно сполучених дуг окружностей), що на порядок знижує обсяг КП в порівнянні з лінійною інтерполяцією [20].

У процесі формування КП на етапі постпроцесування, передбачена можливість розрахунку розміру між роликами для контролю евольвентного профілю зуба, виготовленого за отриманою КП, а також передбачена діагностика помилкових ситуацій у виборі діаметра ролика.

Використання пакету «Евольвента» на заводі спеціалізованого інструменту і технологічної оснастки «Гомсельмаш» (Білорусія), при виготовленні кілець-калібрів (для контролю зубчастих коліс), показало, що відхилення профілю зуба від аналогічних (виготовлених за КП, розрахованих із застосуванням систем PEPS-2 і PRO/Engineer) не перевищує 0,001 мм, а розбіжність розрахованого розміру між роликами не перевищує 0,003 мм в порівнянні з фактичним.

З проведеного аналізу видно, що сучасні ІСАПР дозволяють реалізувати будь-які закони управління, однак, властивий цифровим обчислювальним пристроям послідовний характер реалізації алгоритмів, вимагає розробки проблемно-орієнтованого математичного забезпечення, заснованого на ефективних обчислювальних методах, при яких забезпечуються висока швидкодія і найвища точність опису траєкторій, тому нижче необхідно привести огляд обчислювальних методів формування КП для верстатів з ЧПК.

Питання аналізу методів і засобів вирішення траєкторних задач виникають, з одного боку, у зв'язку з широким вибором тих і інших, а з іншого – через розмаїття технічних вимог і специфіки вирішуваних завдань.

В якості засобів реалізації, в даний час, широко використовується сучасна мікропроцесорна техніка з програмної та мікропрограмного реалізацією алгоритмів. Що ж стосується методів обчислень, то тут головним чином використовуються традиційні, розроблені на перших стадіях розвитку систем ЧПК [21].

Не вдаючись у подробиці оптимального взаємини методів, а отже, алгоритмів і засобів вирішення траєкторних задач, зазначу лише, що кожному вибраному виду засобів відповідає свій метод обчислень, а також – для будь-якого методу можна вибрати оптимальні засоби його реалізації [22], [23]. Природно, що при цьому, в першу чергу, необхідно враховувати набір технічних вимог, що пред'являються до PDM-системи. Перерахую основні з них: точність інтерполяції $h = 0,01 \div 0,001$ мм; максимальні геометричні розміри, що задаються в кадрі програми, 10 м, максимальна швидкість обробки (прискорений хід) від 10 до 12 м/хв; частота слідкуючого приводу не перевищує 500 Гц.

Базові можливості модуля плоского й об'ємного моделювання ADEM CAD наведені нижче.

2D-кооперація:

- можливість плоского моделювання з використанням, як булевих операцій, так і аплікативного, розвинені функції роботи з шарами, штрихування багатозв'язних областей із збереженням асоціативності контура, динамічна модифікація моделі зі збереженням її геометричної цілісності;
- евристична параметризація моделі з автоматичним розпізнаванням топологічних обмежень параметризація з явним завданням параметричної моделі;
- редагування сканованих креслень, розробка векторної моделі на базі сканованого зображення;
- створення параметричних бібліотек фрагментів, позначень, елементів, деталей і складальних вузлів, підтримка стандартів ЄСКД, ANSI та ін.

3D-кооперація:

- автоматичне побудова просторових моделей на основі розрахованих перерізів, використання плоских побудов для формування просторових тіл без додаткових перетворень, використання в якості основи просторових побудов плоских контурів, ребер і граней створених або імпортованих тіл і поверхонь;
- повноцінне твердотільне моделювання з використанням як булевих, так і базових операцій, єдині методи роботи з твердими тілами, поверхнями і відкритими оболонками, здійснення взаємодії між твердими тілами і поверхнями (наприклад, обрізка твердого тіла поверхнею або поверхні твердим тілом);
- локальні операції редагування (власних та імпортованих моделей), побудова заокруглень заданим радіусом (постійним або змінним), а

також фасок (рівносторонніх або різнобічних) на ребрах твердих тіл і відкритих оболонок;

- унікальні операції видалення окремих граней тіла і автоматичне добудовування його до замикання; побудова сполучення, що має сингулярна точку переходу з увігнутою поверхні на опуклу і побудови скруглення на вершині тіла;
- робота зі збірками включає в себе функції за завданням взаємного розташування деталей, аналізу перетинів і визначення величин зазорів;
- автоматична генерація креслярських видів на базі просторової моделі;
- отримання вирізів, розрізів і перерізів;
- імпорт об'ємної геометричної інформації через формати IGES, SAT, STL, STEP.

Обробка ЧПК за допомогою модуля ADEM CAM:

- підготовка керуючих програм (КП) з використанням будь-яких видів геометричних даних, плоских ескізів, креслень, поверхонь, твердих тіл і їх комбінацій;
- повна асоціативність геометричній і технологічній моделей, автоматичне відстеження в УП змін, внесених конструктором;
- динамічне моделювання процесу обробки з можливістю завдання складної заготовки (штампування, лиття) і порівняння результату обробки з математичною моделлю;
- лістопробівка з параметрами вібровисечки, контролем залишкового припуску, нахлеста, виходу перед та після;
- токарна обробка за схемами: чорнове, попереднє, зміщене, прорізка, контурне, чорнова прорізка; формування переходів точити, розточити, відрізати, підрізати, нарізати різьбу; завдання токарних переходів з моделюванням об'ємів видалається припуску для кожного перехо-

- ду, створення власної бібліотеки токарних різців з точним моделюванням ріжучої кромки і завданням настроювальної точки;
- електроерозія 2х і 4х координатна, корекція подачі в кутах і контроль величини недорізів з подальшим доопрацюванням після перезачоплення;
 - 2х, 2.5х, 3х, 5х координатне фрезерування, автоматичний підбір необроблених зон для 2х, 2.5х і 3х координатного фрезерування;
 - розрахунок траєкторії руху інструменту, використовуючи схеми: еквідистанта, зворотній еквідистанта, петля еквідистантних, зигзаг еквідистантних, спіраль, петля, зигзаг, петля UV, зигзаг UV, петля контурна, зигзаг контурний;
 - контроль залишкового припуску окремо для зовнішніх і внутрішніх кордонів конструктивного елемента;
 - багатопрохідних обробка по Z координаті ("Z – level") на основі плоскої геометричній моделі, параметричне завдання ливарних кутів, галтелей і профілю стінки без тривимірного моделювання;
 - контроль ширини розриву поверхні при обробці тривимірної моделі;
 - моделювання елементів пристосувань з контролем на заріз;
 - автоматичний розрахунок точки врізання врізання по нормалі, лінійне з нахилом, радіусне з нахилом, попереднє засвердлювання; формування підходу і відходу по нормалі, лінійного (з контролем довжини і кута) або радіусного (з контролем радіусу і кута розвороту);
 - формування траєкторії з урахуванням або глибини різання, або числа проходів; обробка з постійною висотою гребінця;
 - реалізація програмного управління шпинделем, змащувально-охолоджуючі рідини (ЗОР); використання верстатних циклів робота з викликом підпрограм, формування декількох варіантів маршруту обробки в рамках одного проекту, контроль обробки сполучень, контроль апроксимації траєкторії руху інструменту, як для всього проекту, так і для конкретного переходу;

- збереження петель еквідистанти в процесі налагодження КП, робота з радіусною корекцією, як еквідистантно, так і контурної з контролем довжин відрізків дотичного або перпендикулярного;
- використання інструменту всіх типів: фрези кінцеві, конічні, кутові, дискові, із закругленими або сферичні, створення бібліотеки інструментів;
- параметрична зв'язок між переходами в маршруті обробки, використання користувацьких команд для формування спеціальних кадрів;
- апроксимація траєкторії обробки тривимірної моделі дугами.

Відкритий інтерфейс модуля ADEM TDM:

- ієрархічна, деревоподібна структура побудови конструкторських і технологічних об'єктів (технологічні процеси, операції, переходи, інструмент, оснащення тощо);
- оптимізація використання інформаційних баз даних підприємства відповідно до вимог користувача;
- спеціалізований мова макропрограмування, що дозволяє реалізувати будь-які прикладні алгоритми управління графічними та технологічними параметрами, заповнення вихідних форм, організації запитів баз даних, ведення інженерних розрахунків (параметри різання, нормування тощо). Мова має як стандартні оператори (оператори присвоювання, порівняння, організації циклів тощо), так і спеціалізовані (оператори роботи із запитом баз даних, форматування, завантаження графічної інформації тощо);
- доповнення користувацькими параметрами діалогів технологічних команд в модулі ADEM CAM при побудові маршруту обробки виробу;
- організація зберігання нормативно-довідкової інформації в таблицях бази даних. Структура таблиць не обмежується ADEM TDM, тому можна використовувати як новостворені таблиці бази даних, так і вже існуючі на підприємстві, розроблені за допомогою СУБД різного

- типу: FoxPro, MS Access, MS SQL Server, Paradox та ін. Для отримання інформації з баз даних використовуються стандартні SQL запити;
- можливість обміну інформацією з іншими системами через узгоджені канали даних;
 - вихідні форми і карти будь-якої конфігурації: відповідно до ЄСКД, ЄСТД, стандартом підприємства та ін.;
 - основні варіанти проектування технологічної та конструкторської документації з можливістю діалогового зміни основних параметрів сценарію проектування і записів бази даних.

2.2 Етапи формування КП у системі ADEM 7.0

Покажемо послідовність етапів формування керуючої програми на спроектованій діаграмі послідовності (ДП), що наведено на рис. 2.2.

Частина модуля, що проектує, ADEM CAM (процесор) готує послідовність команд обробки в універсального вигляді (CLDATA). Програма, що перекладає цю послідовність команд з формату CLDATA в формат конкретної стійки ЧПК, називається процесором адаптації або адаптером (рис. 2.3).

У своїй роботі адаптер використовує постпроцесор на верстаті і формує на робочому диску файл з ім'ям PLENT.TAP, що містить текст керуючої програми (КП) у форматі ASCII.

Стойки ЧПК працюють з різними системами кодування символів, що відрізняються від формату ASCII, наприклад: ISO, БЦК та ін. Щоб підготувати КП для завантаження в стійку ЧПК, необхідно перекодувати кожен символ файлу PLENT.TAP з формату ASCII у формат цієї стійки. Це робить перекодувальник, який запускається автоматично після відпрацювання адаптера. Сформований перекодувальником файл PROG.TAP містить КП в форматі стійки ЧПК.

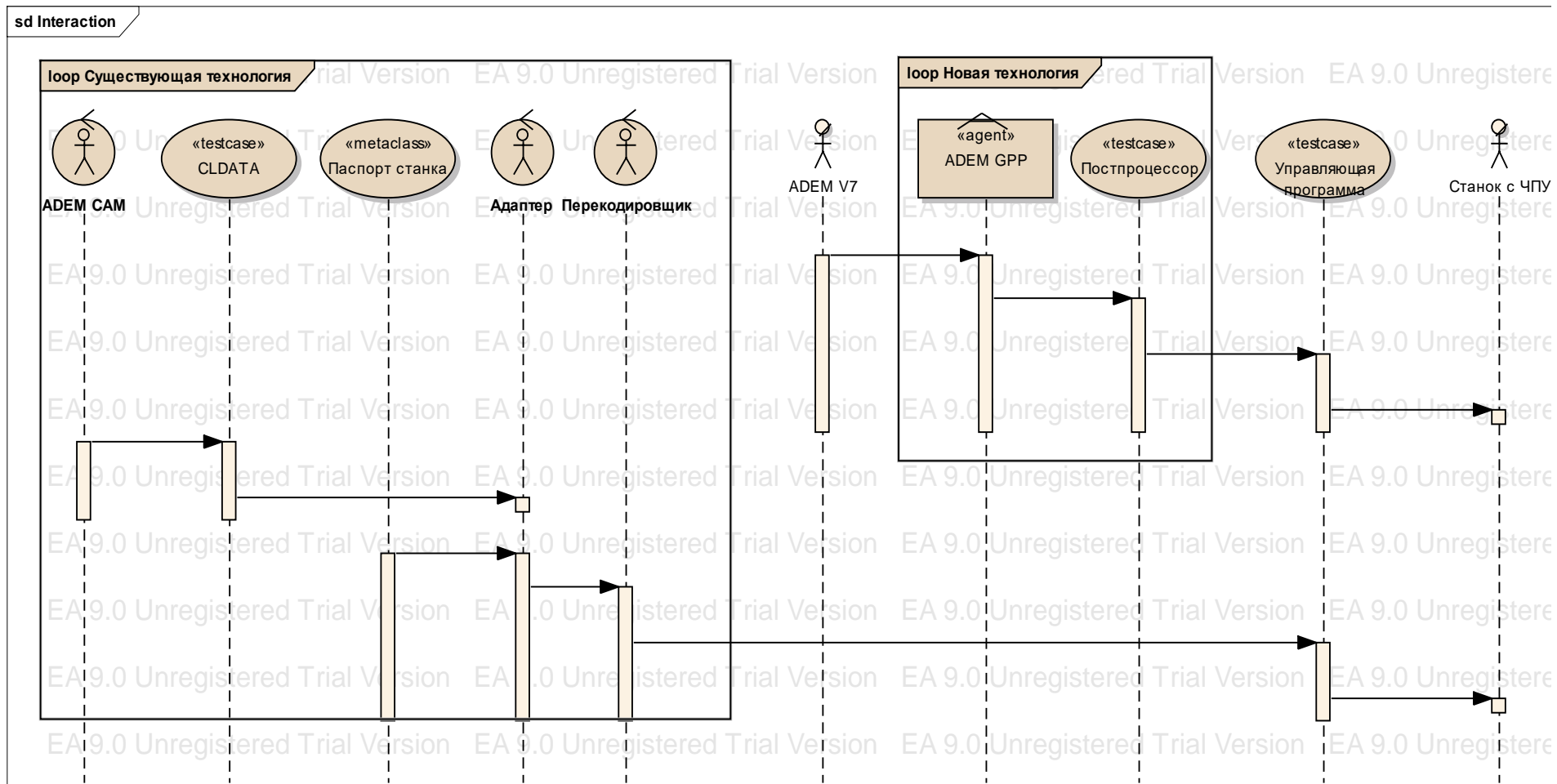


Рисунок 2.2 – ДП існуючої технології та підсистеми, що проектується

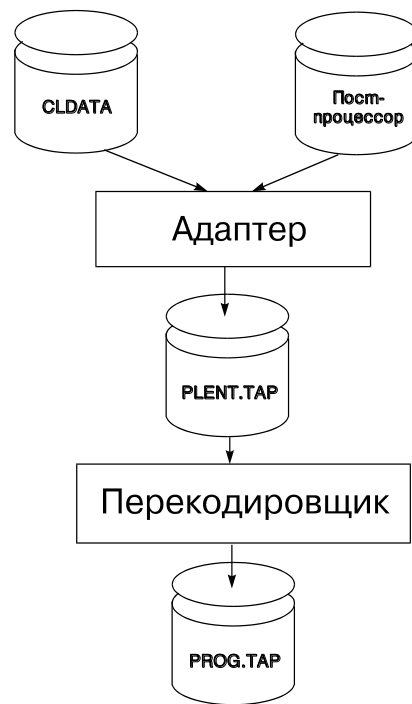


Рисунок 2.3 – Загальний алгоритм роботи адаптера

Файли CLDATA, PLENT.TAP і PROG.TAP є тимчасовими файлами, при виході системи вони знищуються. КП повинна бути записана на диск командою Зберегти КП як. Створюються два файли (основне ім'я задається користувачем):

<Ім'я>. TAP – КП в текстовому вигляді (ASCII-формат);

<Ім'я>. TNC – КП для передачі на верстат у форматі, визначеному в постпроцесорі.

CLDATA в системі ADEM називається проміжна (процесор – адаптер) інформація про траєкторії інструмента та технологічні параметри обробки. CLDATA складається з послідовності команд.

Кожна команда CLDATA позначає певну дію, має свій код і може мати свої параметри (табл. 2.1).

Щоб отримати КП, необхідно представити послідовність дій, що містяться у файлі CLDATA, у вигляді кадрів КП на конкретний верстат.

Два терміна, відомі всім програмістам для верстатів з ЧПК.

Таблиця 2.1 – Приклад команд CLDATA

Команда	Код	Параметри	Дія
Включити прискорене переміщення	25	Без параметрів	Включення прискореного переміщення
Лінійне переміщення	181	Координати X, Y, Z	Переміщення інструменту в точку з коорд. X, Y, Z
Включити робочу подачу	23	Величина подачі	Включення заданої подачі

Слово КП (слово) – складова частина кадру КП, що містить дані про параметр процесу обробки або інші керуючі дані.

Адреса ЧПК (адреса) – частина слова КП, що визначає призначення наступних за ним даних цього слова.

2.3 Завдання, які вирішуються адаптером

Адаптер вирішує наступні завдання:

- визначає ім'я верстата, на який потрібно отримати КП.
- визначає на ім'я верстата, номер постпроцесора, який буде використовуватися при формуванні КП.
- перетворює команди CLDATA в слова і кадри КП.
- компонує інформацію у вигляді єдиної КП.

2.3.1 Визначення імені верстату

Ім'я верстату задається командою ВЕРСТАТ в модулі ADEM CAM. Це ім'я є параметром команди CLDATA ВЕРСТАТ (код 3).

Треба мати на увазі, що після зміни імені верстата необхідно повторно згенерувати CLDATA, тобто виконати команду ПРОЦЕСОР.

2.3.2 Визначення номера постпроцесору

Визначення номера постпроцесора виконується за ім'ям верстата, який шукається в каталозі верстатів. Каталог верстатів – текстовий файл з ім'ям STANKI.SKR, що знаходиться в директорії \ ADEM \ NCM \ POSTPR разом з постпроцесором для верстатів.

Каталог містить записи наступного формату:

| <Ім'я верстата> | <коментар> | <номер постпроцесору> |

Ім'я верстата, задане командою ВЕРСТАТ, має точно відповідати імені, записаному в каталозі верстатів у графі <ім'я верстата>. Якщо адаптер не виявить в каталозі заданого імені, він видасть повідомлення "Верстат не включений до каталогу", адаптер завершить роботу, КП сформована не буде.

2.3.3 Перетворення команд CLDATA в слова та кадри КП

Кожній команді CLDATA поставлений у відповідність алгоритм представлення її дії у форматі КП. Перетворення команди CLDATA в частину КП здійснюється в два етапи:

- пошук за кодом команди CLDATA її алгоритму;
- реалізація знайденого алгоритму.

Пошук алгоритму за кодом команди CLDATA.

Алгоритми відображення дії команд CLDATA містяться у файлі алгоритмів. Цей файл є частиною постпроцесора на верстат і має ім'я, наприклад, для постпроцесора з номером 222, FTRP0222.ANK. Без цього файлу КП формуватися не буде, адаптер видасть повідомлення "Немає файлу алгоритмічного заповнювача".

Якщо на будь-яку команду CLDATA алгоритм не буде знайдений, вона ніяк не відіб'ється в КП.

Реалізація алгоритму.

Алгоритм представляє з себе послідовність рядків наступного формату:

[IF] <умова виконання> [ELSE] <команда алгоритму>;

де IF вказує, що команда повинна бути виконана тільки при дотриманні умови, наступної за IF. Тільки рядки з IF можуть мати альтернативні рядки; ELSE вказує, що даний рядок є альтернативною для рядка з IF, що розташована вище.

Алгоритм може виглядати наступним чином:

```
Z -> E;
X-> XT;
Y-> YT;
IF E! = 1 ПНКАДР;
КАДР;
```

Введемо деякі нові поняття.

Системні та змінні користувача.

Як було зазначено вище, команда CLDATA може мати параметри. Значення параметрів присвоюються відповідним системним змінним при реалізації її алгоритму. Наприклад, при відпрацюванні алгоритму команди Лінійне переміщення (код 181) значення координат точки поточного положення інструмента X, Y і Z присвоюються системним змінним з іменами XT, YT і ZT.

Крім системних змінних існують змінні користувача. Їх імена так само фіксовані (див. додаток А), але значення для користувача змінних визначає лише розробник КП.

При складанні алгоритму можна користуватися лише системними і призначеними для користувача змінними. У наведеному вище прикладі XT і YT – системні змінні, а E, X і Y – призначені для користувача.

Вікно кадру.

Вікно кадру описує слово кадру КП і складається з двох частин:

– символічна частина адреси (містить послідовність символів).

- формат виводу визначає вид виведеної числової інформації (наприклад, максимальна кількість символів, що виводяться, кількість позицій після десяткового дробу тощо).

Приклад: G [],

де G – символна частина вікна;

[] – умовне позначення формату виводу.

Макет кадру.

Вікна кадру у тій послідовності, в якій вони повинні розташовуватися у кадри, містяться у файлі макета кадру. Таким чином, макет кадру – це структура кадру КП: взаємне розташування всіх можливих слів кадру і опис кожного з них.

Цей файл є частиною анкети на верстат і має ім'я, наприклад для анкети з номером 222, KADR0222.ANK. Без цього файлу КП формуватися не буде, адаптер видасть повідомлення "Немає макета кадру".

Виведення інформації у вікно макету кадру.

Алгоритм формує КП, виводячи інформацію у вікна макета кадру. Цю інформацію він отримує з CLDATA за допомогою системних змінних.

Команда алгоритму, що виводить інформацію у вікно макету кадру має наступний формат:

<Номер вікна в макеті кадру> -> <виведена інформація>;

як видно з контексту команди, для виведення інформації у вікно необхідно вказати його порядковий номер у макеті кадру.

Приклад. Інструмент знаходиться в точці з координатами:

x = 10, y = 20, z = 30.

Макет кадру:

N [] G [] G [] X [] Y [] Z [] M []....

Відпрацьовані команди алгоритму:

```
4 -> XT;
5 -> YT;
6 -> ZT;
2 -> 1;
```

Сформована частина кадру:

G1X10.Y20.Z30.

2.3.4 Компонування кадрів КП

Як було сказано вище, КП формується виведенням інформації у вікна макета кадру. При цьому інформація, що виводиться надходить спочатку в буфер формованого кадру. Потім вміст цього буфера передається до КП і утворює там окремий кадр. Передача ця відбувається, наприклад, при відпрацюванні команди алгоритму КАДР.

При формуванні КП виконуються наступні правила.

1) Автоматична нумерація кадрів здійснюється через вікно типу НОМЕР КАДРУ. Формат виводу цього вікна містить інтервал нумерації.

2) Автоматично формується вікно типу КІНЕЦЬ КАДРУ.

3) Збереження інформації. Кадр КП умовно ділиться на частини, кожна з яких визначає включення / вимикання будь-якої функції або її параметри. Ці функції можуть бути альтернативними, коли функція скасовує дію попередньої (наприклад, функції G0, G1, G2, G3 в стійці FANUC), а можуть бути спільно працюючими, тобто можуть розміщуватися в одному кадрі (наприклад, G1, F, M). Для контролю розміщення цих функцій у кадрах введені два поняття:

– тип вікна – це тип функції верстата, реалізований через це вікно (допоміжна функція, підготовча функція і т. д.);

- номер групи вікна – це номер групи альтернативних функцій відповідного типу.

Розглянемо ситуацію. У будь-яке вікно заноситься інформація (наприклад, у вікно $G []$ заноситься 1). При цьому можливі два випадки:

а) кадр, що формується не містить альтернативної функції, тоді він доповнюється заноситься у вікно інформацією ($G1$).

б) кадр, що формується вже містить альтернативну функцію (наприклад, $G2$) – у цьому випадку інформація, що міститься в сформованому кадрі, виведеться в КП окремим кадром, почнеться формування нового кадру.

4) Підтримка модальності. Функції в кадрах КП бувають двох типів: модальні та локальні. Модальні діють до їх скасування альтернативною функцією, локальні діють в межах одного кадру. Для обліку цієї особливості складання КП введено поняття модальність вікна.

5) Модальність вікна. Якщо зазначено, що вікно діє модально, адаптер запам'ятовує останнє виведене в це вікно значення. Коли за алгоритмом знову приходить команда вивести інформацію у це вікно, адаптер порівнює останнє виведене та значення, що виводиться. Якщо вони однакові, інформація не виводиться.

6) Гасіння порожніх кадрів. Адаптер здійснює контроль за наявністю інформації при виведенні кадру. Якщо за алгоритмом відпрацьовується команда КАДР (Кінець кадру), а в сформованому кадрі немає інформації, команда КАДР ігнорується.

2.4 Етапи створення постпроцесору

2.4.1 Склад постпроцесору

Склад підсистеми постпроцесорної обробки наведемо на прикладі спроектованої діаграми компонентів (ДКМ) (рис. 2.4).

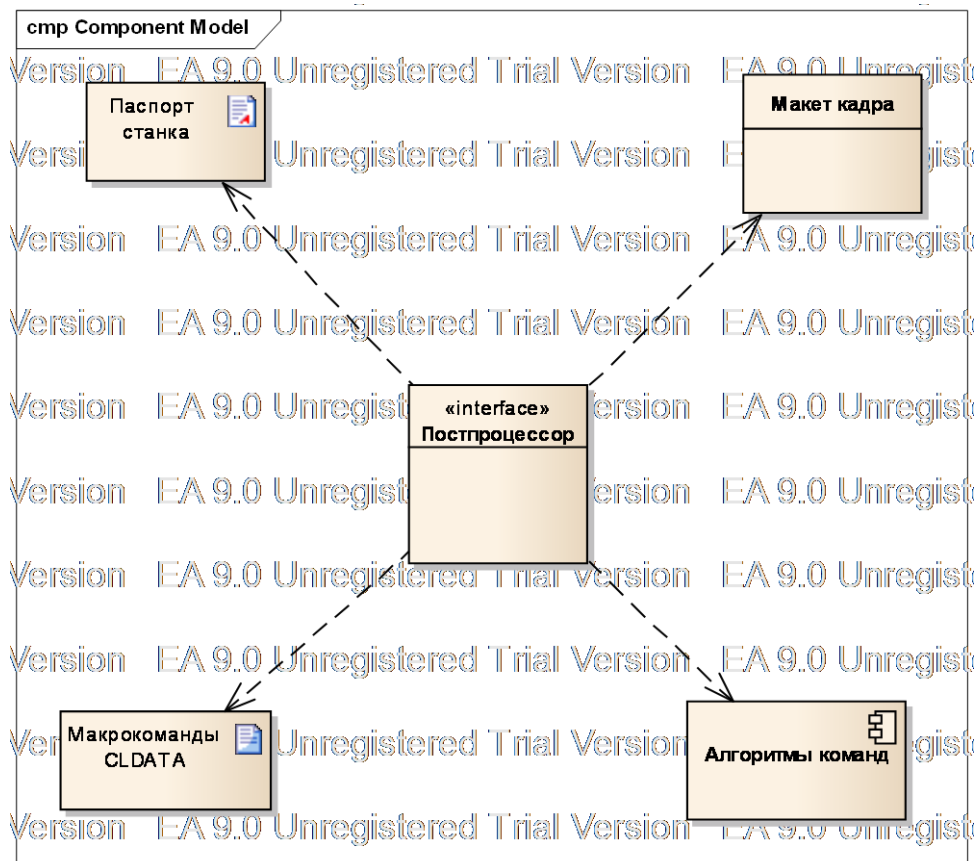


Рисунок 2.4 – ДКМ підсистеми ADEM GPP

Постпроцесор складається з чотирьох частин:

- 1) Паспорт верстата – загальні дані по верстату і правилам програмування.
- 2) Макрокоманди – інформація про обробку адаптером таких команд CLDATA, для реалізації яких необхідно виконати декілька команд.
- 3) Макет кадра – структура кадру КП, тобто взаємне розташування всіх можливих вікон кадру і опис кожного з них.
- 4) Алгоритми представлення команд CLDATA у вигляді кадрів і слів КП.

Паспорт верстата – це набір питань і можливі варіанти відповідей про верстат і правилах програмування для нього.

2.4.2 Написання постпроцесора

1) Запустити програму ADEM GPP – генератор постпроцесорів, головне вікно якої наведено на рис. 2.5.

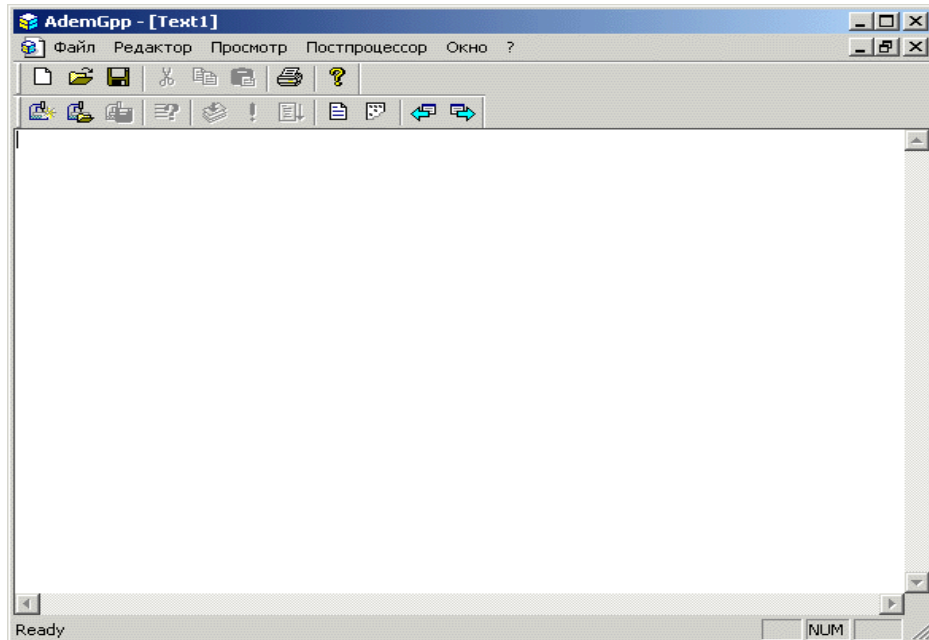



Рисунок 2.5 – Головне вікно ADEM GPP

Генератор постпроцесорів дозволяє виконати наступні дії:

2) Створити новий постпроцесор для цього необхідно натиснути кнопку Створити постпроцесор  після цього відкриється вікно «Новий постпроцесор».

У полі «Номер» треба ввести номер постпроцесора. У полі «Модель» треба ввести модель верстату. У полі «Коментар» треба ввести коментар.

Для створення постпроцесора можна використовувати аналогічний постпроцесор розроблений до нього. Для цього необхідно встановити прапорець «Використовувати постпроцесор-аналог» та в полі «Номер» – ввести номер аналогічного постпроцесора. Можна вибрати постпроцесор зі списку, якщо натиснути кнопку біля поля «Номер». Після заповнення діалогу – натиснути клавішу ОК.

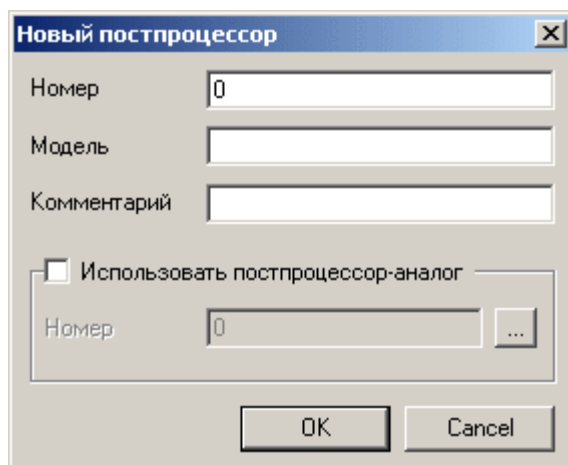



Рисунок 2.6 – Вікно створення нового постпроцесора

3) Відкрити постпроцесор – для цього треба натиснути кнопку Відкрити постпроцесор  – активізується «Відкрити постпроцесор» (рис. 2.7).

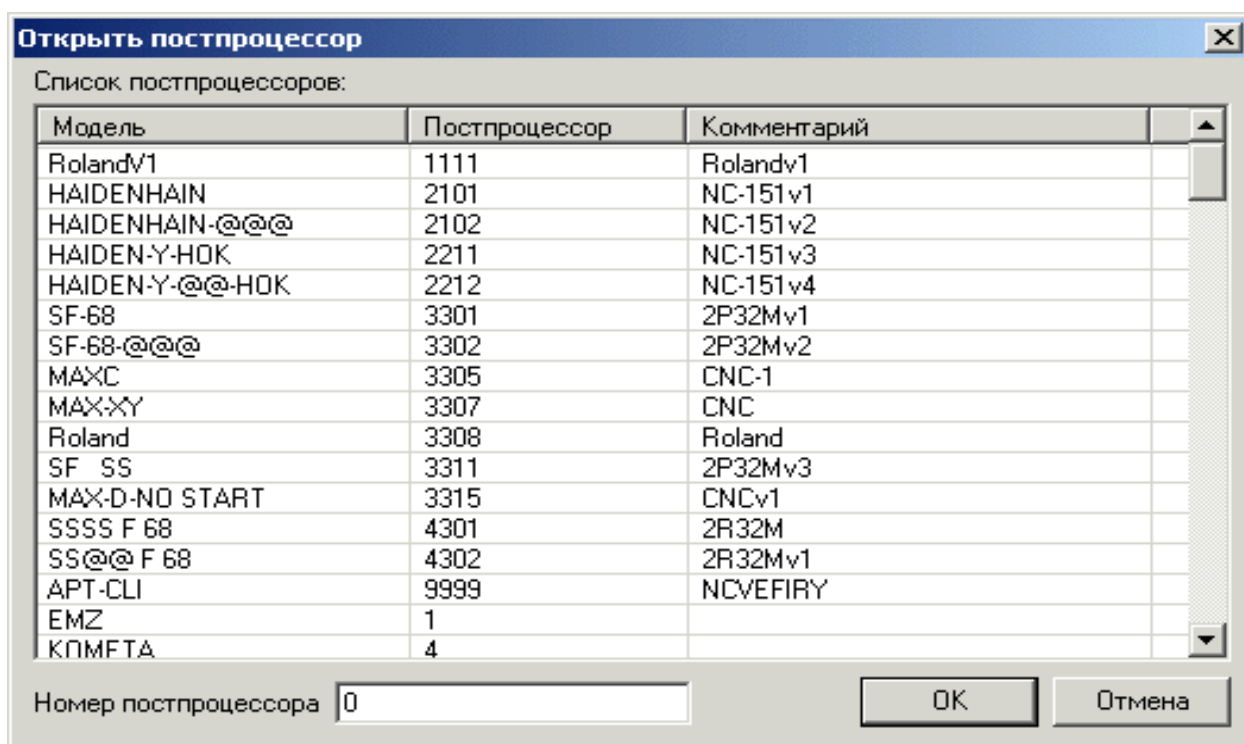




Рисунок 2.7 – Вигляд вікна «Відкрити постпроцесор»

Відкриється діалог вибору постпроцесорів. Зі списку треба вибрати потрібний постпроцесор та натиснути ОК. Якщо відомий номер постпроцесора – ввести його номер у поле «Номер постпроцесора».

- 4) Зберегти постпроцесор – натиснути кнопку  – система збереже активний постпроцесор.
- 5) Заповнити паспорт верстата – натиснути кнопку Параметри  – з'явиться діалог «Постпроцесор» (рис. 2.8).
- 6) Сформуванати макрокоманду – натиснути: Параметри – «Постпроцесор». Вибрати вкладку Макрокоманди (рис. 2.9).

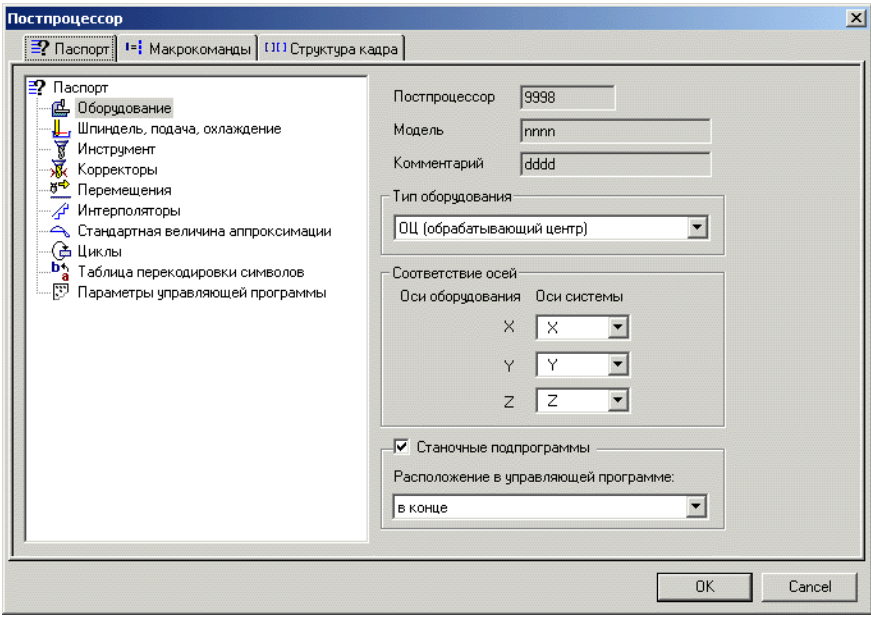


Рисунок 2.8 – Заповнення паспорта

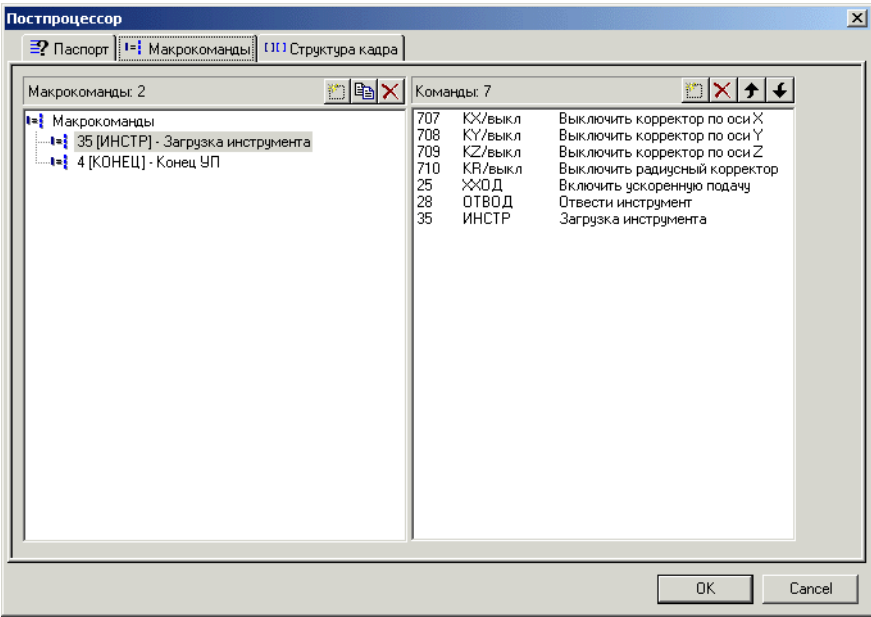


Рисунок 2.9 – Формування макрокоманди

- 7) Сформуванати макет кадру – натиснути: Параметри – «Постпроцесор» – Структура кадру (рис. 2.10).

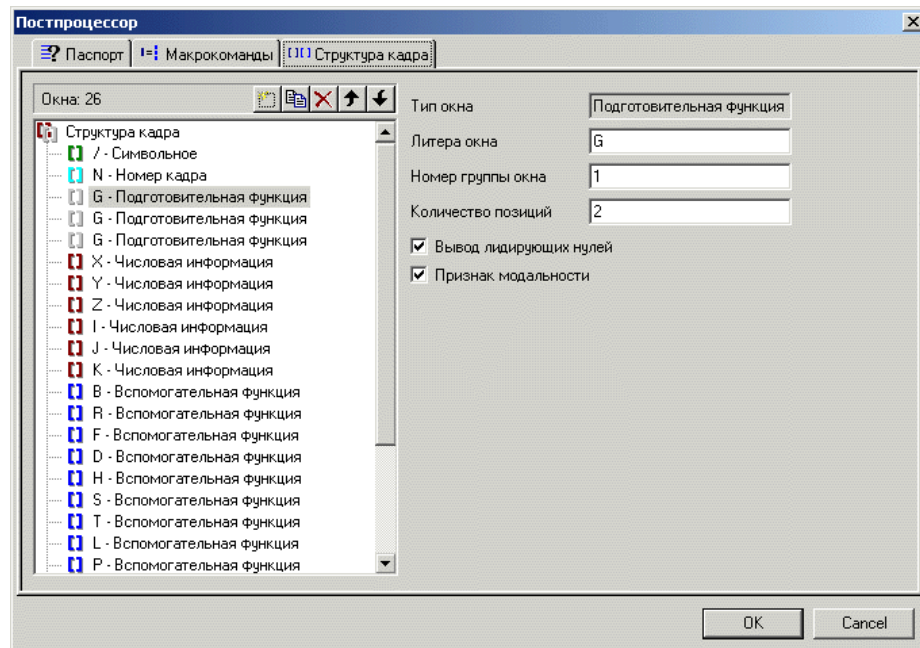



Рисунок 2.10 – Формування макета кадру

- 8) Сформуванати новий файл алгоритму – натиснути кнопку Створити  – відкриється вікно з порожнім текстовим файлом.

- 9) Відкрити алгоритм – натиснути кнопку Відкрити та вибрати файл алгоритму – система відкриє вибраний файл (рис. 2.11).

Наступні дії аналогічні для схожих дій будь-якого текстового редактору, тому в роботі наведемо їх без описів та коментарів.

- 10) Зберігання змін в алгоритмі.
- 11) Робота з буфером обміну.
- 12) Роздруківка файлів постпроцесора (виконується з меню «Файл»).

```

Post9998.pst
P2=1;
G11=BKJKOPR;
END ;

181 ;
IF DZ=0 DO;
  IF P2=1 DO;
    IF ПЛМ=СЛВ G2->42;
    ELSE G2->41;
    D->G11;
    P2=0;
  ENDDO;
  G1->P4;
  IF DX!=0 X->XT;
  IF DY!=0 Y->YT;
  КАДР;
  ENDDO;
END ;

1 ;
25->ИПРОГ;
ПНКАДР;
КАДР;
END ;


END ;

```

Рисунок 2.11 – Текст алгоритму

2.4.3 Трансляція файлу алгоритму

За допомогою трансляції текстовий файл алгоритму переводиться в коди зрозумілі пристрою із ЧПК. Для виконання трансляції необхідно:

- створити новий або відкрити існуючий постпроцесор. У текстовому редакторі з'явиться текст алгоритму, в який можна вносити зміни.
- натиснути кнопку Трансляція .

Якщо в тексті не виявлено помилок – з'явиться повідомлення Трансляція виконана успішно, в іншому випадку – з'явиться повідомлення про помилку у наступному форматі:

<Номер рядка>: Команда <код команди> Помилка у рядку <текст рядка>,

де <Номер рядка> – номер рядка алгоритму, що містить помилку;

<Код команди> – код команди CLDATA, в алгоритмі якій виявлена помилка;



<Текст рядка> – текст рядка, що містить помилку.

Крім того, у вікні текстового редактора система підсвітить рядок з помилкою – необхідно виправити її та повторити трансляцію.

2.4.4 Перегляд результатів роботи постпроцесора

Для того, щоб перевірити постпроцесор – треба увійти в модуль ADEM CAM, командою ВЕРСТАТ встановити верстат, для якого створили постпроцесор, запустити (можливо повторно) Процесор, потім Адаптер і переглянути спроектовану КП. Якщо в постпроцесорі виявилися помилки – знову повернутися в ADEM GPP та змінити його.


Модуль ADEM GPP дозволяє створювати КП, переглядати текст КП та текст CLDATA за допомогою наступних команд:

- виконати Адаптер – кнопка Адаптер . Система переведе CLDTA в КП за допомогою поточного постпроцесора;
- переглянути CLDATA – кнопка CLDATA . Система відкриє вікно з текстом CLDATA;
- переглянути КП – кнопка Керуюча програма. Система відкриє вікно з текстом КП;
- переглянути попереднє вікно – кнопка Попереднє вікно;
- переглянути наступне вікно – кнопку Наступне вікно.

2.4.5 Налаштування постпроцесора

Існує можливість порядкового налаштування алгоритмів команд CLDATA, що також є способом перевірки постпроцесора. Відпрацьовуються команда за командою, при цьому є можливість стежити за значеннями всіх змінних, які формуються кадром і сформованою КП, оперативно змінювати алгоритм. Для налаштування необхідно:

- у модулі ADEM CAM встановити назву верстата, відповідну постпроцесору, що налаштовується та підготувати CLDATA;

- увійти в ADEM GPP і відкрити постпроцесор, що налагоджується;
- натиснути кнопку Налагодження .
- запустити налагоджувач алгоритму (рис. 2.12)

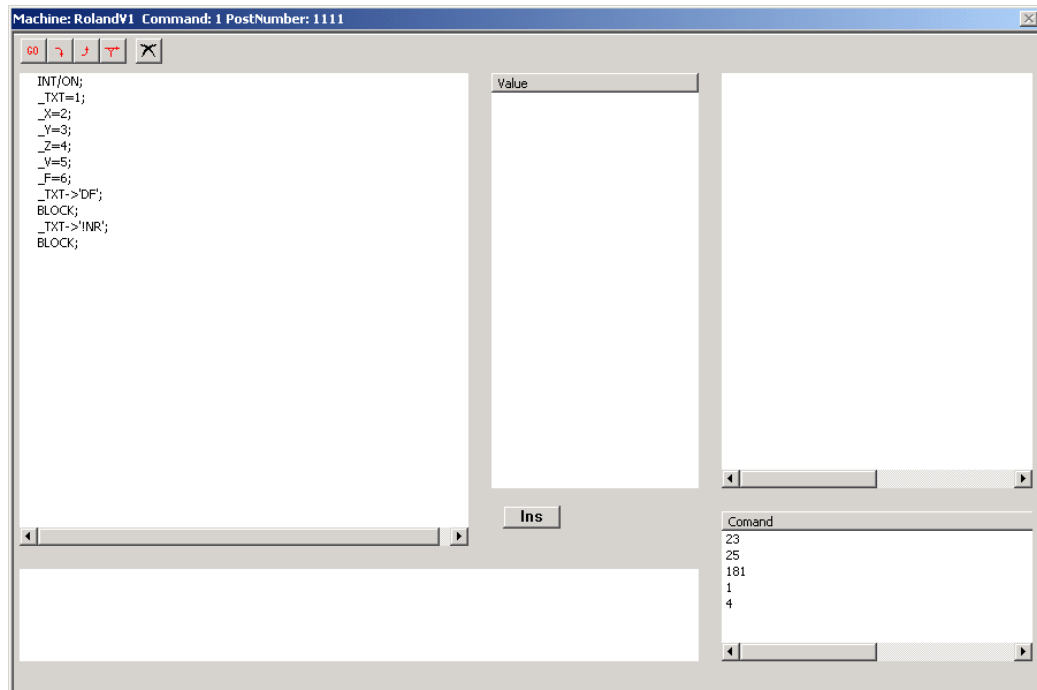


Рисунок 2.12 – Налагоджувач алгоритму.

Треба мати на увазі, що після зміни імені верстату – необхідно перерахувати CLDATA, тобто виконати команду «Процесор» у ADEM CAM.

3 ТЕСТУВАННЯ СФОРМОВАНИХ СТРУКТУР І ОБРОБКА РЕЗУЛЬТАТІВ ПРОЕКТУВАННЯ


3.1 Формування файлу макрокоманд

Нагадаємо, що файл макрокоманд містить інформацію про обробку адаптером тих команд CLDATA, для реалізації яких необхідно виконати декілька команд CLDATA.

Наприклад, команді завантажити інструмент буває потрібно поставити у відповідність команди:




- включити прискорене переміщення (код 25);
- вимкнути коректор за віссю Z (код 709);
- відвести інструмент (код 28);
- вимкнути охолодження (код 700);
- вимкнути шпиндель (код 701);
- завантажити інструмент (код 35).

Файл макрокоманд є обов'язковою частиною постпроцесора і має ім'я, наприклад, для постпроцесора з номером 222: MCOM0222.ANK.

При формуванні файлу макрокоманд необхідно запустити модуль підготовки і налагодження постпроцесорів (ADEM GPP), далі натиснути кнопку Параметри  – з'явиться вікно Постпроцесор. Треба активізувати закладку Макрокоманди (рис. 3.1).

Закладка має два вікна. У лівому вікні знаходяться сформовані макрокоманди, а у правому створюються підлеглі елементи. Для формування макрокоманди необхідно вибрати зі списку в лівому вікні макрокоманду, яка буде включати в себе допоміжні команди. Потім у правому вікні створюються допоміжні команди. Для роботи використовуються кнопки правого і лівого вікна. Якщо кнопка належить лівому вікну, то вона працює з макрокомандами. Якщо кнопка належить правому вікну, то вона працює з підлеглими елементами.

Розглянемо детальніше отриману структуру вікна:

- кнопка Створити  додає новий елемент зі списку команд CLDATA;
- кнопка Копіювати копіює виділений елемент;
- кнопка Видалити видаляє виділений елемент;
- кнопка Перемістити вгору  переміщує виділений елемент вгору;
- кнопка Перемістити вниз  переміщує виділений елемент вниз.

Повний список команд CLDATA з кодами наведено у додатку Б.

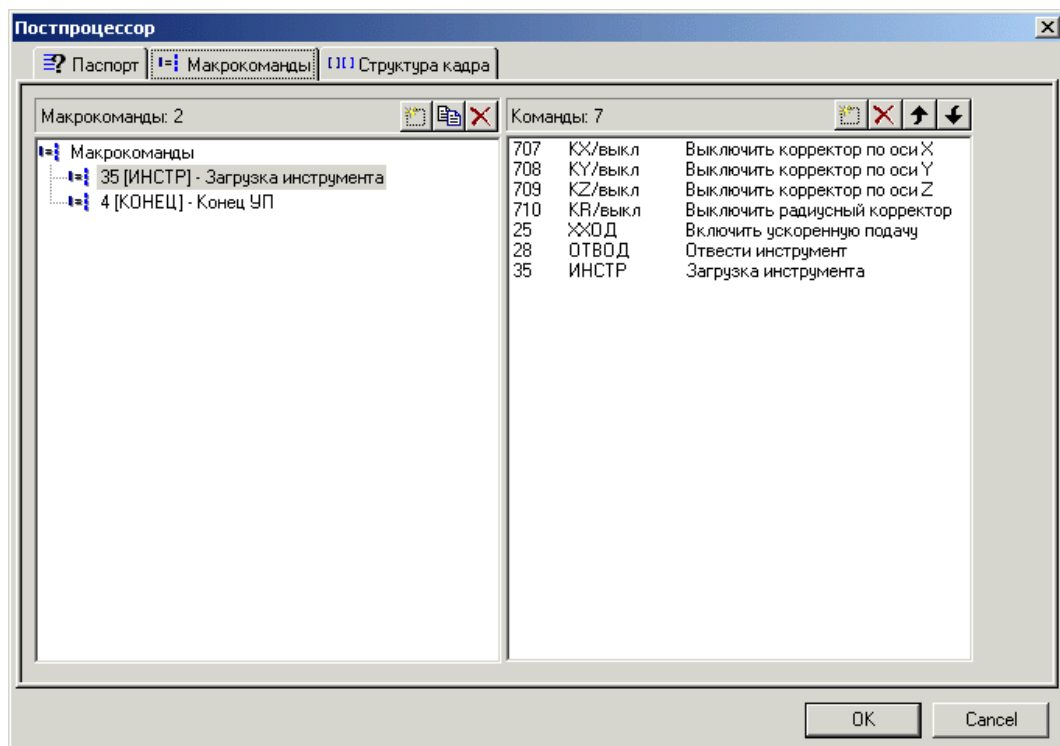


Рисунок 3.1 – Вікно макрокоманд

Приведемо приклад формування файлів макрокоманд. Необхідно сформувати файл макрокоманд постпроцесора з номером 1 і занести в нього плани одпрацювання двох команд Завантаження інструменту і Кінець керуючої програми (рис. 3.2).

Для лівого вікна за допомогою кнопки Створити додається:

- завантаження інструменту;

Для правого вікна за допомогою кнопки Створити додаються:

- вимкнути коректор X;
- вимкнути коректор Y;
- вимкнути коректор Z;
- включити прискорену подачу;
- відвести інструмент;
- вимкнути шпиндель;
- завантажити інструмент.

Для лівого вікна за допомогою кнопки Створити додається:

- кінець керуючої програми.

Для правого вікна за допомогою кнопки Створити додається:

- вимкнути коректор X;
- вимкнути коректор Y
- вимкнути коректор Z;
- включити прискорену подачу;

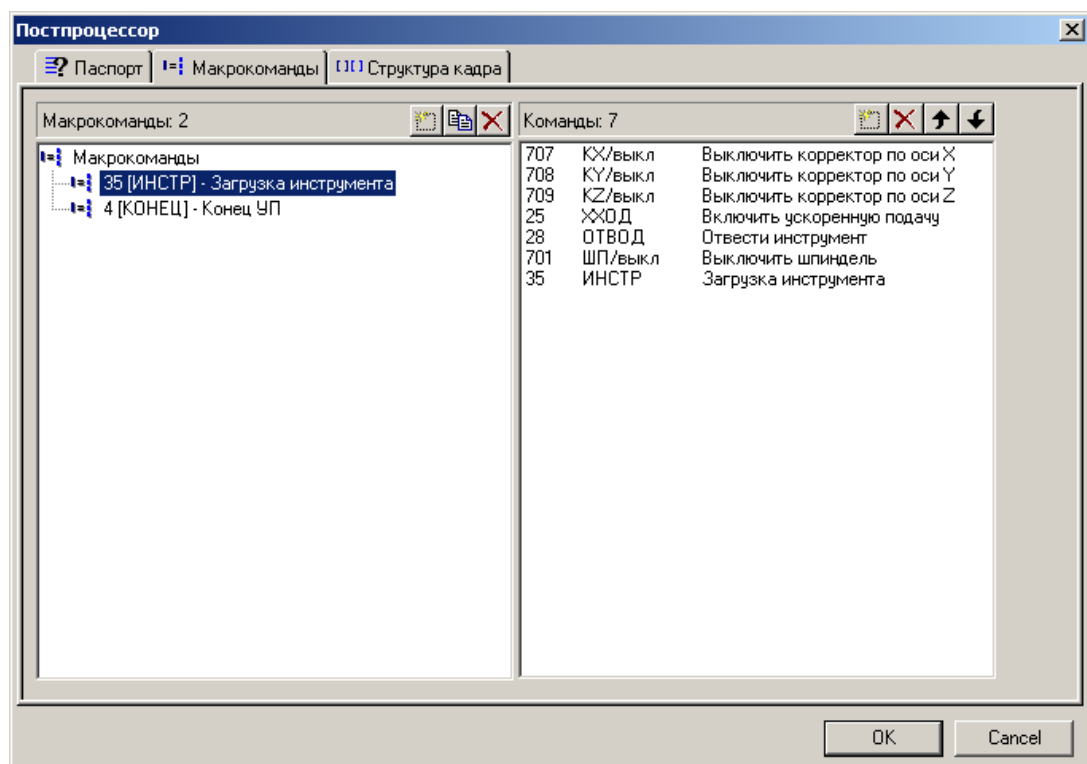


Рисунок 3.2 – Вікна макрокоманд Завантаження інструменту і Кінець КП

- відвести інструмент;
- вимкнути ОР;
- вимкнути шпиндель;
- кінець програми.

3.2 Макет кадру та формат виводу

Макет кадру – це структура кадру КП: взаємне розташування всіх можливих вікон кадру і опис кожного з них.

Вікно кадру описує слово кадру КП і складається з двох частин:

Символьна частина – адреса ЧПК (може містити кілька символів);

Формат виводу – визначає вид виведеної числової інформації.

Приклад вікна кадру:

G [],

де G – символьна частина вікна;

[] – умовне позначення формату виводу.

Макет кадру:

```
N [ ] G [ ] G [ ] X [ ] Y [ ] Z [ ] I [ ] J [ ] K [ ] F [ ]
M [ ] L [ ]
```

Файл макета кадру має ім'я, наприклад, для постпроцесору з номером 222, KADR0222.ANK. Без цього файлу КП формуватися не буде, адаптер видасть повідомлення "Немає макета кадру".

Приклад файлу макета кадру дивіться у Додатку Д.

3.3 Формування вікон різних типів

Наведемо запити системи, які можуть виникнути при формуванні різних типів вікон.

НОМЕР КАДРУ:

- літера вікна;
- кількість позицій;
- інтервал нумерації кадрів;
- вивід лідируючих нулів.

ПІДГОТОВЧА і ДОПОМІЖНА ФУНКЦІЇ:

- літера вікна;
- номер групи вікна;
- кількість позицій;
- вивід лідируючих нулів;
- ознака модальності.

ЧИСЛОВА ІНФОРМАЦІЯ:

- літера вікна;
- номер групи вікна;
- кількість позицій;
- вісь дії вікна;
- придушення лівих нулів;
- ознака модальності.
- висновок знаку "+".

Інші параметри залежать від того, у якому вигляді виводиться число в кадр: у вигляді дробу або у вигляді цілої кількості дискрет.

При виборі в списку «Ціле число дискрет»:

- величина дискретності; при виборі в списку «Дробова частина»:
- виведення останніх нулів;
- виведення нульових значень;
- виведення десяткового дробу;

- придушення правих нулів в дрібних числах;
- вісь дії вікна.

СИНХРОДОРІЖКА:

- літера вікна.

ТЕКСТ:

- літера вікна.

СИМВОЛЬНЕ ВІКНО:

- літера вікна.

КІНЕЦЬ КАДРУ:

- літера вікна.

3.4 Формування макету кадру

Для формування макету кадру – необхідно натиснути кнопку Параметри – з'явиться діалог «Постпроцесор», далі обрати закладку Структура кадру (рис. 3.3).

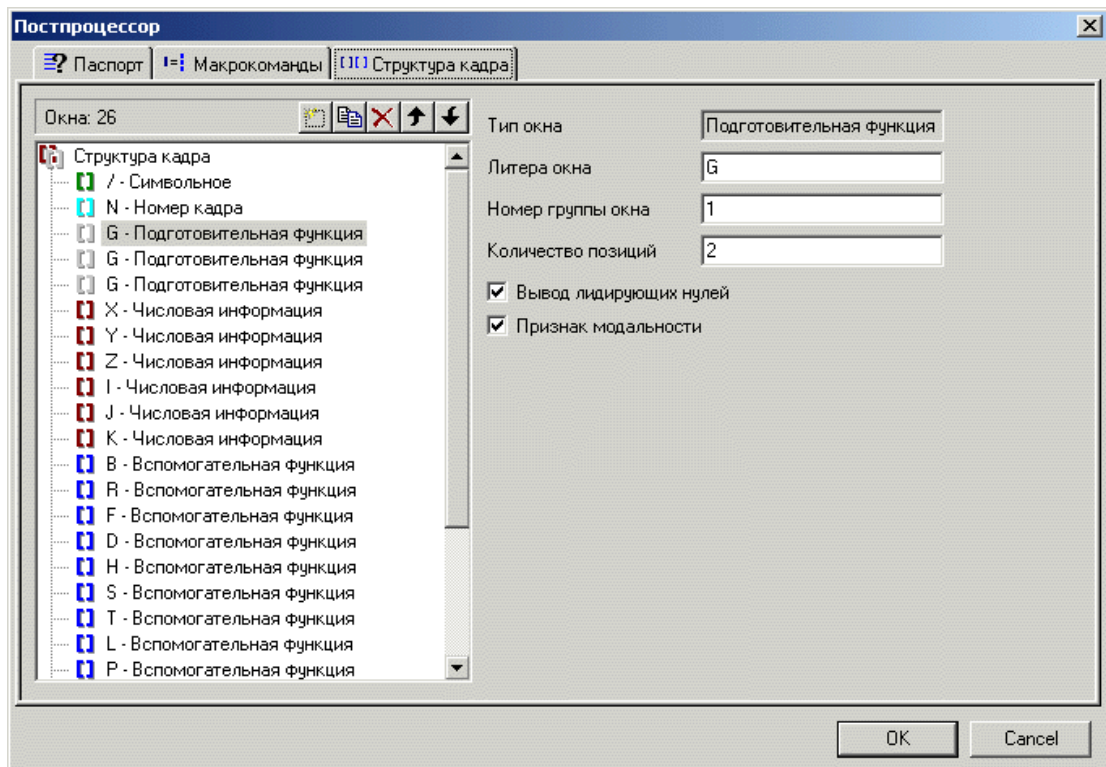



Рисунок 3.3 – Формування макета кадру

Формування вікна кадру відбувається після натиснення кнопки Створити  – відкриється додаткове меню – необхідно обрати тип вікна і визначити параметри формату виводу вибраного вікна:

- тип вікна;
- номер кадру;
- підготовча функція;
- допоміжна функція;
- числова інформація;
- синхродоріжка;
- текст;
- символічне вікно;
- кінець кадру;

Після цього можна описати нове вікно, яке займе своє місце після поточного і буде мати черговий порядковий номер.

Для копіювання видалення і переміщення необхідно використовувати присутні стандартні кнопки.

3.5 Формування файлу алгоритма

Алгоритми відображення дії команд CLDATA містяться у файлі алгоритмів, що є частиною постпроцесора на верстат і має ім'я, наприклад, для анкети з номером 222: FTRP0222.ANK. Без цього файлу КП формуватися не буде, адаптер видасть повідомлення "Немає файлу алгоритмічного заповнювача".

Алгоритм представляє собою послідовність рядків наступного формату:

[<Мітка>:] [ELSE] [IF <умова виконання>] <команда алгоритму>;

де IF – вказує, що команда повинна бути виконана тільки при дотриманні умови, що йде після IF – тільки рядки з IF можуть мати альтернативні рядки;

ELSE – вказує, що даний рядок є альтернативним;

<Мітка> – ідентифікатор рядків при посиланнях (ціле позитивне число);

<Команда алгоритму> – визначає дію з формування КП або зміни значення системної змінної або змінної користувача.

Приклад файлу алгоритмів наведено у додатку Е.

3.6 Арифметичні дії та функції в алгоритмах

Символьні вирази, які можуть міститися у КП та відрізняються своїм поданням від розповсюджених мов програмування, вони означають виконання деяких дій, що зведені у табл. 3.1.

Таблиця 3.1 – Символьні вирази та відповідні їм дії, які можуть міститися у КП

Символьний вираз	Інтерпретація
:	Ділення
**	Піднесення до ступеню
&	Логічне множення (і)
	Логічне додавання (або)
!=	Нерівність


Функції та їх позначення, які можуть використовуватися у алгоритмах та відрізняються своїм поданням від розповсюджених мов програмування, зведені у табл. 3.2.


Для визначення пріоритету виконання дій – використовуються дужки.


Таблиця 3.2 – Функції та відповідні їм позначення, які можуть міститися у алгоритмах КП

Позначення функції	Функція
LOGD	Десятковий логарифм
LOGE	Натуральний логарифм
СOKH	Величина, що виводиться у вікно
NTC	Номер точки циклу в таблиці
NNC	Номер початку циклу в таблиці
DEV	Обчислення цілої частини з округленням
PARV1	Зсув нуля поточної системи координат по осі X базової системи координат
PARV2	Зсув нуля поточної системи координат по осі Y базової системи координат
PARV3	Зсув нуля поточної системи координат по осі Z базової системи координат
PARV4	Кут, на який потрібно повернути поточну систему координат для того, щоб проекція обраної осі поточної системи координат на площину XY базової системи координат сполучилася з віссю X базової системи координат.
PARV5	Кут, на який потрібно повернути поточну систему координат для того, щоб проекція обраної осі поточної системи координат на площину XZ базової системи координат сполучилася з віссю X базової системи координат
PARV6	Кут, на який потрібно повернути поточну систему координат для того, щоб проекція обраної осі поточної системи координат на площину XY базової системи координат сполучилася з віссю Y базової системи координат
PARV7	Кут, на який потрібно повернути поточну систему координат для того, щоб проекція обраної осі поточної системи координат на площину ZY базової системи координат сполучилася з віссю Y базової системи координат.
PARV8	Кут, на який потрібно повернути поточну систему координат для того, щоб проекція обраної осі поточної системи координат на площину XZ базової системи координат сполучилася з віссю Z базової системи координат.
PARV9	Кут, на який потрібно повернути поточну систему координат для того, щоб проекція обраної осі поточної системи координат на площину ZY базової системи координат сполучилася з віссю Z базової системи координат.
XYZ	Базова система координат
X'Y'Z'	Поточна система координат

3.7 Завантаження файлів алгоритмів

Для формування нового файлу алгоритму – необхідно натиснути кнопку Створити  – відкриється вікно із порожнім текстовим файлом. При створенні нового постпроцесора вікно алгоритму створюється автоматично.

Відкрити алгоритм – натиснути кнопку Відкрити  – вибрати файл алгоритму. Система відкриє обраний файл. При відкритті постпроцесора вікно алгоритму відкривається автоматично.

Зберегти зміни в алгоритмі – натиснути кнопку Зберегти  – система збереже зміни в поточному алгоритмі. При збереженні постпроцесора та виконанні трансляції – алгоритм зберігається.

Алгоритм формується та редагується у текстовому вигляді (рис. 3.4). Перед трансляцією алгоритму у файл – система перевіряє його на наявність помилок. Рядок не включається до алгоритму, якщо у ньому допущені помилки. Наприклад: наприкінці немає символу «;».

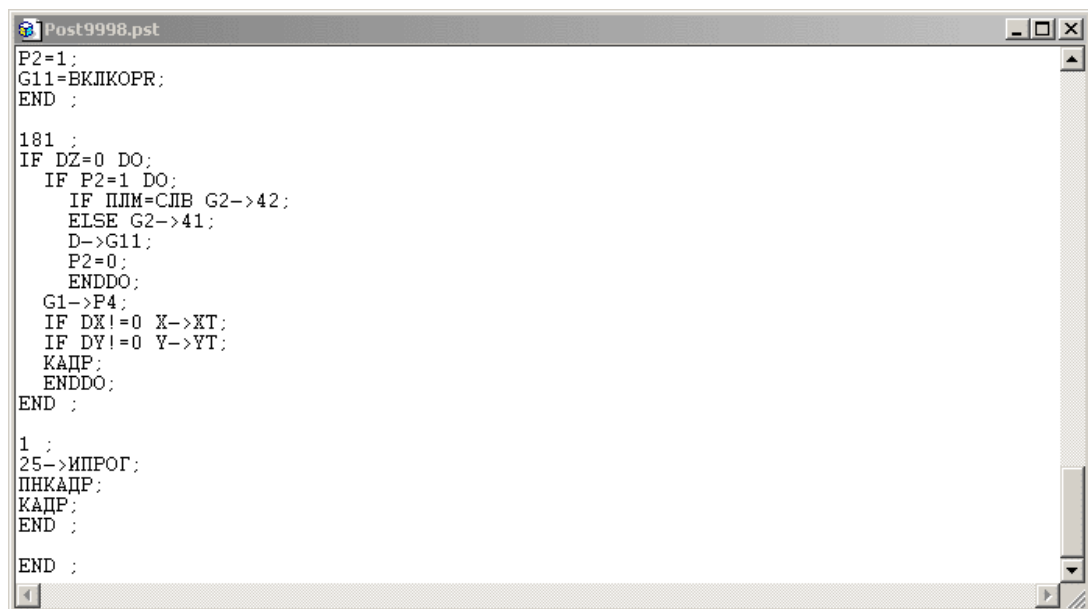


Рисунок 3.4 – Редагування тексту алгоритму

3.8 Приклад формування файлу алгоритмів

На визначеному етапі необхідно сформувати файл алгоритмів постпроцесора з номером 1 для токарного верстату, тобто занести до нього алгоритми наступних команд:

- лінійна інтерполяція;
- кругова інтерполяція;
- включити подачу;
- включити прискорене переміщення;
- включити шпиндель;
- включити охолодження;
- змінити інструмент;
- виключити шпиндель;
- виключити охолодження;
- кінець КП.

Викликаємо вікно диспетчера постпроцесорів і встановлюємо його номер – 1. Можна почати зі складання алгоритма на кожну з CLDATA команд. Почнемо з ініціалізації номерів вікон макету кадра в алгоритмі команди Програма, її код дорівнює 1 (повний список команд із їхніми кодами наведено у Додатку Б).

Припустимо: макет кадру постпроцесора 1 має такий вигляд:

$N[]G[]X[]Z[]I[]K[]F[]T[]S[]M[]L[]*[]$.

Користувальницькій змінній G дамо номер вікна $G[]$ у макеті кадра. Для цього введемо в алгоритм рядок: $G=2$; таким чином ініціалізуємо всі номери вікон макета кадра. Алгоритм після закінчення набору прийме наступний вигляд:

```
1;
G=2;
X=3;
```

```

Z=4;
I=5;
K=6;
F=7;
T=8;
SK=9;
M=10;
L=11;
END.

```

Необхідно звернути увагу на рядок 8 алгоритму. Номер вікна $S[]$ позначений через користувальницьку змінну SK тому, що змінна S – є системою та буде містити значення подачі.

Виходячи з того, що всі номери вікон макету ініціалізовано, закінчимо на цьому складання алгоритма команди з кодом 1.

Уводимо код команди Лінійна інтерполяція, її код дорівнює 181.

Припустимо, що КП формується в збільшеннях, збільшення за віссю X записується у кадрі з адресою X, збільшення за віссю Z записується у кадрі з адресою Z. Нульові збільшення за осями не формуються.

Сформуємо алгоритм обробки команди. Щоб сформувати ненульове переміщення за віссю X, занесемо у вікно $X[]$ його величину. Для цього введемо рядок алгоритму:

```
IF DX!=0 X->DX;
```

Аналогічно сформуємо ненульове переміщення за віссю Z:

```
IF DZ!=0 Z->DZ;
```

Оформимо інформацію у вигляді кадру. Для цього введемо команду алгоритму КАДР:

```

181;
IF DX!=0 X->DX;
IF DZ!=0 Z->DZ;
КАД;
END.

```


Уводимо код команди Кругова інтерполяція, її код дорівнює 182. Припустимо, що кругова інтерполяція формується в збільшеннях. Збільшення за віссю X записується у кадрі за адресою X, збільшення за віссю Z записується у кадрі за адресою Z, ненульова абсолютна різниця координат за віссю X початкової точки дуги й центра дуги записуються за адресою I, ненульова абсолютна різниця координат за віссю Z початкової точки дуги й центра дуги записуються за адресою K. Рух за годинною стрілкою задається функцією G2, проти годинної стрілки – G2.

Сформуємо алгоритм:

```
183;
X->DX;
Z->DZ;
IF XЦОК!=XC I->ABS(XЦОК-XC);
IF ZЦОК!=ZC K->ABS(ZЦОК-ZC);
IF НАПРОК=ЧС G->2;
ELSE G->3;
КАДР;
END.
```

Наступна команда: Включити подачу, код 22.

Припустимо, що величина робочої подачі формується за адресою F.

Сформуємо алгоритм відпрацювання команди:

```
23;
F->S;
END.
```

Необхідно звернути увагу на те, що у алгоритмі немає команди алгоритма КАДР. Це значить, що інформація до кадру буде виведена у наступних командах.

Наступна команда: Включити прискорене переміщення, код 25. Припустимо, що величина прискореного переміщення дорівнює 70000 та формується вона за адресою F. Сформуємо алгоритм обробки команди:

```
25;
F->70000;
```

END.

Наступна команда: Включити шпиндель, код 23. Припустимо, що величина обертів шпинделя задається за адресою S. Напрямок обертання за годинною стрілкою визначає функція M3 – проти годинної – M3. Сформуємо алгоритм обробки цієї команди:

```
24;
SK->N;
IF НВШП=ЧС M->3;
ELSE M->4;
END.
```

Наступна команда: Включити охолодження, код 26. Припустимо, що охолодження включається функцією M8. Сформуємо алгоритм обробки цієї команди:

```
26;
M->8;
END.
```

Наступна команда: Змінити інструмент, код 35. Припустимо, що номер позиції інструменту встановлюється за адресою T, а команда на завантаження інструмента задається функцією M6. Все наведене повинно бути оформлено окремим кадром. Сформуємо алгоритм обробки цієї команди:

```
35;
КАДР;
T->ТИНСТ;
M->6;
КАДР;
END.
```

Наступна команда: Виключити шпиндель, код 701. Припустимо, що вимикання шпинделя задається функцією M5. Сформуємо алгоритм обробки цієї команди:

```
701;
```

```
КАДР;  
М->5;  
КАДР;  
END.
```

Наступна команда: Виключити охолодження, код 700. Припустимо, що вимикання охолодження задається функцією М9. Сформуємо алгоритм обробки цієї команди:

```
700;  
КАДР;  
М->9;  
КАДР;  
END.
```

Остання команда: Кінець керуючої програми, код 3. Припустимо, що кінець КП задається функцією М2. Сформуємо алгоритм обробки цієї команди:

```
4;  
КАДР;  
М->2;  
КАДР;  
END.
```

Таким чином, у цьому підрозділі було сформовано приклади всіх фрагментів необхідних алгоритмів.

Після завершення процесу формування файлу алгоритмів, збережемо файл постпроцесора, що створено, під номером: 1.

3.9 Налаштування алгоритмів

Всі етапи формування адаптером КП можна спостерігати при запуску процесора адаптації у режимі налаштування (рис. 3.5).

Для цього необхідно:

- у системі ADEM CAM встановити ім'я верстата (одержати CLDATA);

- увійти до модуля ADEM GPP;
- запустити відладник (кнопка Налагодження )

У заголовку вікна (рис. 3.5) перебуває наступна інформація:

- ім'я верстата;
- код команди, що відпрацьовується, CLDATA;
- номер постпроцесора.

У лівому верхньому розділі вікна:

- алгоритм на поточну команду CLDATA (підсвітлюється поточний рядок алгоритму).

У лівому нижньому розділі вікна:

- сформований кадр КП.

У центральному розділі вікна:

- область завдання системних і користувальницьких змінних, що використовується в алгоритмах для контролю їхніх значень;
- клавіша INS – дозволяє ввести в поле ім'я змінної для контролю її значення.

У правому верхньому розділі вікна:


- сформовані кадри КП.

У правому нижньому розділі вікна:

- список алгоритмів на команди CLDATA, що задіяні у постпроцесорі;
- подвійний щиглик миші на номері алгоритму – виводить його вміст на екран.

Відладник може працювати у двох режимах:

- повне налагодження до точки зупинки (якщо задана);
- покрокове налагодження.

Повне налагодження виконується при натисканні клавіші  (або клавіші – F8) до точки зупинки (Breakpoint), що відзначають подвійним щигликом миші на першому символі рядка алгоритму (або клавіша – F9).

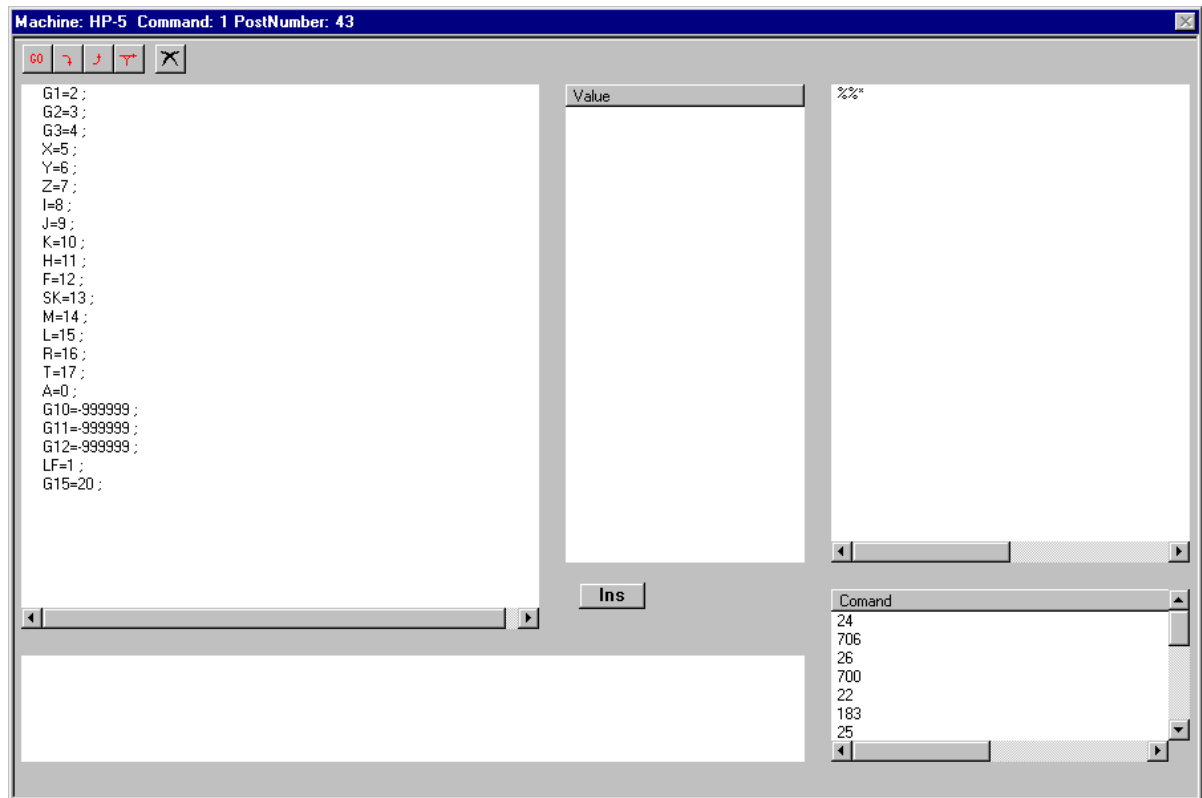




Рисунок 3.5 – Вікно відладника алгоритмів

Покрокове налагодження виконується при натисканні клавіші : (алгоритми усіх команд CLDATA виконуються покроково, точки зупинок ігноруються).

Додаткові режими налагодження виконуються при натисканні клавіш :

- перша клавіша дає можливість вийти з поточного алгоритму із його виконанням;
- друга клавіша – відпрацьовує виклики алгоритмів (CALL) не входячи до них.

ВИСНОВКИ

В процесі виконання кваліфікаційної роботи, була досягнута мета роботи, а саме – створений PDM-модуль постпроцесорної обробки, що сполучається із ІСАПР «ADEM».

Для досягнення встановленої мети, в роботі було вирішено ряд наступних поставлених задач:

- виконано огляд систем підготовки КП, для встановлення їх недоліків та переваг, а також схожих рис, що можуть бути використані для створення нового модуля;
- розібрано етапи створення постпроцесора, з обов'язковою процедурою трансляції файлів та переглядом результатів проектування;
- створено паспорти верстатів, файлів макрокоманд, макетів кадрів, алгоритмів команд.

Об'єктом роботи став процес створення КП, за допомогою якої виконується механічна обробка на верстатах із ЧПК.

Предметом роботи – програмний модуль проектування КП, що сполучується із ІСАПР ADEM.

Отримані результати, полягають у наступному:

- забезпечено перетворення команд CLDATA в кадри КП, за допомогою пошуку відповідного алгоритму подання;
- удосконалено автоматичне компонування кадрів КП, яке відрізняється проміжною передачею до буфера сформованого кадру;
- отримав подальший розвиток процес проектування графічного інтерфейсу користувача генератора постпроцесорів із можливістю відображення макрокоманд та переглядом результатів роботи.

У роботі використані наступні методи дослідження: методи аналітичної геометрії і обчислювальної математики для забезпечення розрахунків фізичних параметрів механообробки (різання і точіння) тіл обертання, методи і принципи автоматизованого проектування, методи організації графічного ді-

алогу та побудови інтерфейсу користувача з використанням стандартів управління ANSI та ISO.

Бакалаврська кваліфікаційна робота включила у себе виконання наступних основних етапів:

- аналіз існуючих систем автоматизованого проектування і підготовки даних для механообробки;
- огляд загальних принципів побудови керуючих програм;
- формування КП в системі ADEM 7.0;
- представлення форматів подання даних;
- завдання, які вирішуються адаптером;
- створення постпроцесору;
- формування паспорта верстату;
- тестування сформованих структур і обробка результатів моделювання.

Практичне значення отриманих результатів полягає в наступному: результати реалізовані у вигляді програмного модуля, що створює основу для проектування КП для механічної обробки матеріалів на верстатах з ЧПК.

Моделі рішення задач проектування реалізовані програмно. У процесі тестування і випробувань вони показали свою працездатність та ефективність на конкретних прикладах завдань механічної обробки матеріалів.

Створений графічний модуль взаємодії оператора верстата із генератором постпроцесорів, з можливістю відображення макрокоманд і переглядом результатів трансляції. Дана система дозволяє оператору провести контроль кадрів КП без відпрацювання програми на верстаті. Також даний модуль дозволяє оператору виконувати налагодження постпроцесора і формувати паспорт верстата і редагувати параметри КП.

Загалом, після проведених досліджень, можна зробити висновки, що розвиток систем проектування КП в системах ЧПК, має важливе економічне значення в галузі обробки машинобудівної продукції, як засіб зменшення кількості браку, внаслідок виявлення «простих» помилок автоматичного про-

формування КП, на етапах налагодження і запуску перших зразків серійної продукції.

При практичному застосуванні програмного модуля (ПМ), вихідні дані можуть бути задані в результаті машинної обробки ADEM-CAM-файлу або сформовані інженером-технологом аналітично. Додатково в ПМ, передбачені необхідні коментарі щодо коректності формування макета кадру і формату виведення КП.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Филиппович В. И. Инвариантное постпроцессирование. Дискуссия об итогах тридцатилетней эволюции метода. Пермь: ООО «Евразия Лимитед», 2003. 48 с.
2. Великодний С. С. Методологические основы реинжиниринга систем автоматизированного проектирования. *Управляющие системы и машины*. 2014. № 2. С. 39–43.
3. Филиппович К. В. Идеология постпроцессирования в современных CAD/CAM-системах. Пермь: ООО «Евразия Лимитед», 2000. – 60 с.
4. Великодний С. С. Модели и методы интерполяции сложных геометрических контуров для систем автоматизированного проектирования формообразования деталей. Монографія. Харків: ФОП Панов А.Н., 2017. 232 с.
5. Werner J. The Case For Verifying And Optimizing Tool Paths. Irvine: CGTech, 2003. 58 p.
6. Великодний С. С. Проблема реинжиниринга видов обеспечения систем автоматизированного проектирования. *Управляющие системы и машины*. 2014. № 1. С. 57–61, 76.
7. Филиппович К. В. ToolStore – среда для ведения библиотеки инструментов в верификаторе CNC-Verify системы САПР-ЧПУ/2005. Пермь: ООО «Евразия Лимитед», 2005. 36 с.
8. Невлюдов И. Ш., Великодний С. С., Омаров М. А. Использование CAD/CAM/CAE/CAPP при формировании управляющих программ для станков с ЧПУ. *Восточно-Европейский журнал передовых технологий*. 2010. № 2/2 (44). С. 37–44.
9. Грунина Е. В. Методика проектирования в CAD/CAM Cimatron УП для гравирования профиля на юбилейной медали. Ярославль: ЯГТУ, 2008. 15 с.

10. Великодний С. С. Моделі та методи проактивного управління проектами із розвитку програмних систем і продуктів. Монографія. Одеса: Одеський державний екологічний університет, 2021. 322 с. ISBN 978-966-186-182-3. (URL: <http://eprints.library.odeku.edu.ua/id/eprint/9595/>)
11. Velykodniy S. Reengineering of open software system of 3D modeling BRL-CAD. *Innovative Technologies and Scientific Solutions for Industries*. 2019. No 3 (9), P. 62–71. (кат. «Б») DOI: <https://doi.org/10.30837/2522-9818.2019.9.062>.
12. Великодний С. С., Тимофєєва О. С., Зайцева-Великодна С. С. Метод розрахунку показників оцінки проекту при виконанні реінжинірингу програмних систем. *Радіoeлектроніка, інформатика, управління*. 2018. № 4. С. 135–142. DOI: 10.15588/1607-3274-2018-4-13.
13. Zelinski P. A Better Process From Better Posts. *Serving the Metalworking Industries*. 2011. №2. P. 28–31.
14. Schultz D.F. What's Wrong With Postprocessors? *Numerical Control Computer Sciences*. 1998. №1. P. 11–15.
15. Константинов М. П. Расчёт программ фрезерования на станках с ЧПУ. Москва: Машиностроение, 1985. 160 с.
16. Уралов В. И. Технологическая подготовка многооперационных станков. Москва: Машиностроение, 1985. 88 с.
17. Подготовка исходных данных и работа с универсальной системой подготовки управляющих программ для токарных станков с ЧПУ (СПС-ТАУ): инструкция. Москва: ЭНИИМС, 1977. 94 с.
18. Великодний С. С. Моделювання складних процесів та систем (Частина 1): конспект лекцій. Одеса: Одеський державний екологічний університет, 2021. 92 с. ISBN 978-966-186-181-6. (URL: <http://eprints.library.odeku.edu.ua/id/eprint/9494/>)
19. Velykodniy S. S. Analysis and synthesis of the results of complex experimental research on reengineering of open CAD systems. *Applied Aspects of*

Information Technology. 2019. Vol. 2. No 3. P. 186–205. (кат. «Б») DOI: 10.15276/aait.03.2019.2.

20. Кохан Д. Проектирование технологических процессов и переработка информации: пер. с нем. Москва: Машиностроение, 1981. 312 с.
21. Великодный С. С. Модели и методы интерполяции сложных геометрических контуров. Для систем автоматизированного проектирования формообразования деталей. Монография. 2-е изд. Эрфурт: LAMBERT Academic Publishing, 2019. 236 с.
22. Вашкевич С. Н. Алгоритмы для решения задач формообразования в микропроцессорных системах ЧПУ. Станки и инструмент. 1986. №11. С. 29–30.
23. Сосонкин В. Л. Микропроцессорные системы числового программного управления станками. Москва: Машиностроение, 1985. 288 с.

ДОДАТОК А

Список систем і змінних користувача

Даний додаток містить повний перелік систем і змінних користувача, які можуть бути використані у файлах алгоритмів.

Таблиця А.1 – Координати інструмента

XT XCURPOS	Координата X формована.
YT YCURPOS	Координата Y формована.
ZT ZCURPOS	Координата Z формована.
XC XBEG	Координата X попередня.
YC YBEG	Координата Y попередня.
ZC ZBEG	Координата Z попередня.
DX	Приріст по осі X.
DY	Приріст по осі Y.
DZ	Приріст по осі Z.
AT ACURPOS	Формоване кутове положення А.
BT BCURPOS	Формоване кутове положення В.
CT CCURPOS	Формоване кутове положення С.
AC ABEG	Попереднє кутове положення А.
BC BBEG	Попереднє кутове положення В.
CC CBEG	Попереднє кутове положення С.
DA	Кутовий приріст А.
DB	Кутовий приріст В.
DC	Кутовий приріст С.
ДЛПЕР LENG/MOV	Довжина переміщення.

Таблиця А.2 – Кругова інтерполяція

ХЦОКР X/CENTER	Координата центра окружности X.
УЦОКР Y/CENTER	Координата центра окружности Y.
ЗЦОКР Z/CENTER	Координата центра окружности Z.
РОКР R/CIRC	Радіус окружности.
НАПРОКР DIR/CIRC	Напрямок руху по дузі окружности.
ЧС CW	По годинниковій стрілці [1].
ПЧС CWW	Проти вартовий стрілки [0].
ВОКР TYPECIRC	Вид окружности.
ВЫП CONVEX	Опукла [0].
ВОГН CONCAVE	Увігнута [1].
ЦУГОЛ ANG/CIRC	Центральний кут дуги.
КПЛОК ПЛОКР C/PLACE	Площина дуги окружности.
XY, XZ, YZ	Площина XY, XZ, YZ.

Таблиця А.3 – Наступні переміщення інструмента

ВХСЛ V/XNEXT	Вектор наступного переміщення по осі X.
ВУСЛ V/YNEXT	Вектор наступного переміщення по осі Y.
ВЗСЛ V/ZNEXT	Вектор наступного переміщення по осі Z.
УГНАП ANG/M	Кут перелому траєкторії.
К/ПЕРЕЛ C/BREAK	Код перелому траєкторії.

Таблиця А.3 – Геометрія й номер позиції інструментів

ТИНСТР N/TOOL	Номер інструмента, що завантажує.
ИНСТР1 FIRSTOOL	Перший інструмент програми.
СЛИНСТР NEXTOOL	Наступний інструмент.
КОЛИНСТ NOF/TOOL	Кількість інструментів у таблиці.
ПНИНСТ IND/TOOL	Порядковий номер, що завантажує інструмент у таблиці.
РИНСТР RAD/TOOL	Радіус завантажуваного інструменту.
РСЛИНСТР NEXT/RTL	Радіус наступного завантажуваного інструменту.
ВЫЛЕТХ XOVERH	Виліт інструмента по осі Х.
ВЫЛЕТУ YOVERH	Виліт інструмента по осі У.
ВЫЛЕТZ ZOVERH	Виліт інструмента по осі Z.

Таблиця А.4 – Вмикання / вимикання коректорів

ВКЛКОРХ XCOMPON	Коректор, що включає, по Х.
ВКЛКОРУ YCOMPON	Коректор, що включає, по У.
ВКЛКОРZ ZCOMPON	Коректор, що включає, по Z.
ВКЛКОРR RCOMPON	Коректор, що включає, по R.
ВЫКЛКОРХ XCOMPOFF	Коректор, що виключає, по Х.
ВЫКЛКОРУ YCOMPOFF	Коректор, що виключає, по У.
ВЫКЛКОРZ ZCOMPOFF	Коректор, що виключає, по Z.
ВЫКЛКОРR RCOMPOFF	Коректор, що виключає, по R.

Таблиця А.5 – Керування шпинделем

N NOB SPIN	Обороти шпинделя.
V VPEЗ VC	Швидкість різання.
NДИАП N/RANGE	Номер діапазону шпинделя.
НВШП DIR/SPIN	Напрямок обертання шпинделя.
КОДШПИНД COD/SPIN	Код у ряді шпинделя.
СЛНОБ NEXTSPIN	Наступні обороти шпинделя.
СЛVPEЗ NEXTVC	Наступна швидкість різання.
СЛНДИАП NEXT/RNG	Наступний номер діапазону шпинделя.
СЛНВШП NX/DIRSP	Наступне направлення обертання шпинделя.
СЛКОДШП NEXTC/SP	Наступний код у ряді шпинделя.
УГОРШП ANG/SPIN	Кут орієнтації шпинделя.

Таблиця А.6 – Керування подачею

S ПОДМИН FEED	Подача в мм/хв.
SOБ ПОДОБ FEEDS/T	Подача в мм/об.
СЛПОДМИН NEXT/F/T	Наступна подача в мм/хв.
СЛПОДОБ NEXT/FD	Наступна подача в мм/про.

Таблиця А.7 – Різьблення

П/РЕЗЬБЫ PR/THRD	Профіль різьблення.
МЕТРИЧ METRIC	Метрична [0].
ТРАПЕЦ TRAPEZ	Трапецеїдальна [1].
УПОРНАЯ BUTTRESS	Упорна [2].
ТРУБНАЯ PIPE	Трубна [3].
ПРЯМОУГ SQUARE	Прямокутна [4].
В/РЕЗЬБЫ KINDTHRD	Вид різьблення.
НАРУЖ EXTERN	Зовнішня [0].
ВНУТ INTERN	Внутрішня [1].
Т/РЕЗЬБЫ TYPETHRD	Тип різьблення.
ПРАВАЯ RIGHT	Права [0].
ЛЕВАЯ LEFT	Ліва [1].
СБЕГ RUN/OUT	Збіг різи.
Ф/РЕЗЬБЫ FORMTHRD	Форма різьблення.
ЦИЛ STRAIGHT	Циліндрична [0].
КОНИЧ TAPER	Конічна [1].
УГР ANG/THRD	Кут різьблення.

ДОДАТОК Б

Команди CLDATA

Найменування	Позначення	Код
Програма	ПРОГРАМ	1
Деталь	ДЕТАЛЬ	2
Верстат	СТАНОК	3
Кінець КП	КОНЕЦ	4
Стоп програми	СТОП	22
Включити робочу подачу	ПОДАЧА	23
Включити шпиндель	ШПИНД	24
Включити прискорену подачу	ХХОД	25
Включити СОЖ	СОЖ	26
Включити вистій	ВЫСТОЙ	27
Відвести інструмент	ОТВОД	28
Перезахоплення (лістоштампівка)	ПЕРЕХВАТ	29
Допоміжна функція	ФУНВ	30
Підготовча функція	ФУНП	31
Пропуск кадрів	ПРОПУСК	32
Умовна зупинка програми	ОСТАНОВ	33
Зміна столу	СТОЛ	34
Завантаження інструменту	ИНСТР	35
Цикл	ЦИКЛ	36
Поворот стола	ПОВОРОТ	40
Сполучене переміщення	МУЛЬТИ	41
Різьба	РЕЗЬБА	90
Лінійна інтерполяція	ИДИТОЧ	181
Кругова інтерполяція	ИДИОКР	183
Виклик підпрограми	ВЫЗОВ	223
Початок підпрограми	ПОДПРОГ	252
Фіксація початку циклу	НЦ	401
Позиція зміни інструменту	БЕЗПОЗ	451
Вставка в КП	ВСТАВКА	458
Перевірка умови	ЕСЛИ	526
Передача керування на мітку	ИДИМЕТ	527
Мітка	МЕТКА	528
Коментарій програми	КОММЕНТ	582
Виключити СОЖ	СОЖ/ВЫКЛ	700
Виключити шпиндель	ШП/ВЫКЛ	701
Завершити пропуск кадрів	ПР/ВЫКЛ	702
Включити коректор по осі X	KX/ВКЛ	703
Включити коректор по осі Y	KY/ВКЛ	704
Включити коректор по осі Z	KZ/ВКЛ	705
Включити радіусний коректор	KR/ВКЛ	706
Виключити коректор по осі X	KX/ВЫКЛ	707
Виключити коректор по осі Y	KY/ВЫКЛ	708
Виключити коректор по осі Z	KZ/ВЫКЛ	709
Виключити радіусний коректор	KR/ВЫКЛ	710
Орієнтований останов шпинделя	ШП/ОР	711
Кінець підпрограми		713
Зарезервована команда (її алгоритм ніколи не спрацьовується)		799

ДОДАТОК В

Приклад паспорта верстата

Постпроцесор, що приведено тут як приклад, поставляється разом із системою й має номер 222. Він підготовлений для верстата моделі 6313Ф3, стійка НЗЗ.

1. Верстат: Фрезерний.

2. Відповідність осей:

XCT->XDET;
YCT->YDET;
ZCT->ZDET.

3. Максимальне переміщення:

по осі X - 0.0 мм;
по осі Y - 0.0 мм;
по осі Z - 0.0 мм.

4. Довжина заправної частини стрічки - 0.0 м.

5. Кількість пропусків між кадрами: 3.

6. СОЖ.

Включення - вручну;

7. Підпрограми - немає.

8. Інтерполяція.

Лінійна – немає;

Кругова – є;

Максимальний радіус інтерполяції 99999.0 м;

Переміщення по дузі розбиваються на октанти.

9. Апроксимація за замовчуванням - 0.009 м.

10. Інструмент.

Завантаження інструмента - ручна;

Завантаження першого інструмента наприкінці програми - завантажувати;

Координати БЕЗПОЗ - не обумовлено;

Площина зміни інструмента - у точці НЦ;

Стандартна площина холостого ходу - не формувати;

Площини переміщення на x/ходу за декількома координатами - XY, YZ, XZ, XYZ.

11. Подача.

Встановлюється - програмно;

Діапазони - безступінчасте регулювання;

Місце вмикання подачі - в одному кадрі з переміщенням.

12. Шпиндель.

Частота обертання шпинделя встановлюється - вручну;

Діапазони - безступінчасте регулювання;

Місце вмикання шпинделя - негайно після одержання команди.

13. Корекція по осях включається.

X – негайно після одержання команди;

Y – негайно після одержання команди;

Z – при першому переміщенні по осі Z;

R – в одному кадрі з переміщенням.

14. Корекція по осях вимикається

X – негайно після одержання команди;

Y – негайно після одержання команди;
Z – при першому переміщенні по осі Z;
R – в одному кадрі з переміщенням.

15. Початковий блок – %*

16. Кінцевий блок – *

17. Стандартний код формування програми – ISO парний.

18. Верстатні цикли – відсутні.

19. Змінювані символи:

* -> 10;
W -> 136.

ДОДАТОК Г

Приклад файлу макрокоманд

Постпроцесор, що наведений у даному додатку у якості приклада має номер 222.

Макрокоманда = 35	(Завантаження інструменту)
KZ/ВЫКЛ	709 (Виключити коректор по осі Z)
ХХОД 25	(Включити прискорену подачу)
ОТВОД 28	(Відвести інструмент)
ШП/ВЫКЛ	701 (Виключити шпиндель)
СОЖ/ВЫКЛ	700 (Виключити СОЖ)
ИНСТ 35	(Завантаження інструмента)

Макрокоманда = 4	(Кінець КП)
KZ/ВЫКЛ	709 (Виключити коректор по осі Z)
ХХОД 25	(Включити прискорену подачу)
ОТВОД	28 (Відвести інструмент)
ШП/ВЫКЛ	701 (Виключити шпиндель)
СОЖ/ВЫКЛ	700 (Виключити СОЖ)
КІНЕЦЬ	4 (Кінець КП)

ДОДАТОК Д

Приклад макета кадру

N[]G[]X[]Y[]Z[]I[]J[]K[]F[]M[]L[]W____%[]([]

N[]	НОМЕР КАДРУ
	Макс. кількість позицій під число – 3
	Придушення лівих – немає
	Інтервал нумерації – 1
G[]	ПІДГОТОВЧА ФУНКЦІЯ
	Номер групи – 1
	Макс. кількість позицій під число – 2
	Придушення лівих – немає
	Ознака модальності – у межах кадру
G[]	ПІДГОТОВЧА ФУНКЦІЯ
	Номер групи – 2
	Макс. кількість позицій під число – 2
	Придушення лівих – немає
	Ознака модальності – у межах кадру
X[]	ЧИСЛОВА ІНФОРМАЦІЯ
	Номер групи – 1
	Макс. кількість позицій під число – 6
	Придушення лівих – немає
	Висновок "+" для позитивних чисел – є
	Ознака модальності – у межах кадру
	Вісь дії – X
Y[]	ЧИСЛОВА ІНФОРМАЦІЯ
	Номер групи – 1
	Макс. кількість позицій під число – 6
	Придушення лівих – немає
	Висновок "+" для позитивних чисел – є
	Ознака модальності – у межах кадру
	Вісь дії – Y
Z[]	ЧИСЛОВА ІНФОРМАЦІЯ
	Номер групи – 1
	Макс. кількість позицій під число – 6
	Придушення лівих – немає
	Висновок "+" для позитивних чисел – є
	Ознака модальності – у межах кадру
	Вісь дії – Z

I[]	ЧИСЛОВА ІНФОРМАЦІЯ
	Номер групи – 1
	Макс. кількість позицій під число – 6
	Придушення лівих – немає
	Висновок "+" для позитивних чисел – є
	Ознака модальності – у межах кадру
	Вісь дії – X
J[]	ЧИСЛОВА ІНФОРМАЦІЯ
	Номер групи – 1
	Макс. кількість позицій під число – 6
	Придушення лівих – немає
	Висновок "+" для позитивних чисел – є
	Ознака модальності – у межах кадру
	Вісь дії – Y
K[]	ЧИСЛОВА ІНФОРМАЦІЯ
	Номер групи – 1
	Макс. кількість позицій під число – 6
	Придушення лівих – немає
	Висновок "+" для позитивних чисел – немає
	Ознака модальності – у межах кадру
	Вісь дії – ?
F[]	ЧИСЛОВА ІНФОРМАЦІЯ
	Номер групи – 2
	Макс. кількість позицій під число – 4
	Придушення лівих – немає
	Висновок "+" для позитивних чисел – немає
	Ознака модальності – у межах кадру
	Вісь дії – ?
M[]	ДОПОМІЖНА ФУНКЦІЯ
	Номер групи – 1
	Макс. кількість позицій під число – 2
	Придушення лівих – немає
	Ознака модальності – у межах кадру
L[]	ПІДГОТОВЧА ФУНКЦІЯ
	Номер групи – 2
	Макс. кількість позицій під число – 3
	Придушення лівих – немає
	Ознака модальності – у межах кадру
W__%[]	СИВОЛЬНЕ ВІКНО
[]	КІНЕЦЬ КАДРУ.

ДОДАТОК Е

Приклад файлу алгоритмів

1; (Код команди CLDATA-програма)

```
Q=0;
G17=0;
G1=2;
G2=3;
X=4;
Y=5;
Z=6;
I=7;
J=8;
K=9;
F=10;
M=11;
L=12;
P5=0;
END;
```

3; (Верстат)

```
G->17;
КАД;
END;
```

4; (Кінець КП)

```
F->624;
КАД;
F->624;
M->2;
АОКНО 13;
КАД;
END;
```

22; (Стоп програми)

```
КАД;
M->0;
КАД;
END;
```

23; (Включити робочу подачу)


```

IF S>=1000 F->4700+S:100;
ELSE IF S>=500 F->4600+S:10;
ELSE IF S>=100 F->600+S:10;
ELSE IF S>=10 F->500+S;
ELSE IF S>=1 F->400+S*10;
ELSE IF S>=0.1 F->300+S*100;
ELSE F->310;
P1=0;
P5=0;
END;

```

24; (Включити шпиндель)

```

КАД;
M->3;
КАД;
END;

```

25; (Включити прискорену подачу)

```

P1=1;
END;

```

27; (Включити вистій)

```

G->4;
X->ВЫСТВР;
F->560;
END;

```

33; (Умовний останов програми)

```

M->1;
END;

```

35; (Завантаження інструменту)

```

IF P1=1 DO;
F->624;
КАД;
F->624;
M->0;
КАД;
ENDDO;
P1=1;

```

END;

181; (Лінійна інтерполяція)

```

IF КСЛКОМ=23 P5=0 // СЛЕД. КОМАНДА - ВКЛ. ПОДАЧИ;
ELSE P5=4000;
IF P1=1 DO // УСКОРЕННОЕ ПЕРЕМЕЩЕНИЕ;
IF ДЛПЕ >400 F->P5+730;
ELSE IF ДЛПЕ >200 F->P5+724;
ELSE IF ДЛПЕ >10 F->P5+712;
ELSE F->P5+680;
ENDDO;
IF Q=201 DO // ПОДХОД НА ВКЛ. РАДИУСНОЙ КОРРЕКЦИИ;
L->800+G17;
Q=200;
ENDDO;
IF DX!=0 X->DX;
IF DY!=0 Y->DY;
IF DZ!=0 Z->DZ;
IF Q=400 DO // ВКЛЮЧЕНИЕ РАДИУСНОЙ КОРРЕКЦИИ;
G1->50;
L->G17;
КАД;
G1->4;
F->624;
L->0;
Q=0;
ENDDO;
ELSE IF Q=300 DO // ВЫКЛЮЧЕНИЕ КОРРЕКТОРА ПО Z;
G->40;
L->400+G20;
Q=0;
ENDDO;
ELSE G->1;
КАД;
END;

```

183; (Кругова інтерполяція)

```

X->DX;
Y->DY;
IF ABS(ХС-ХЦОКР)>=0.01 I->ABS(ХС-ХЦОКР);
IF ABS(УС-УЦОКР)>=0.01 J->ABS(УС-УЦОКР);
IF НАПРОКР=ЧС G->2;
ELSE G->3;
IF Q=200 DO // ВКЛЮЧЕНА РАДИУСНАЯ КОРРЕКЦИЯ;
IF ВОКР=ВОГ L->800+G17;
ELSE L->G17;
ENDDO;
КАД;
END;

```

701; (Виключити шпиндель)

```
КАД;  
M->5;  
END;
```

705; (Включити коректор по осі X)

```
G20=ВКЛКОРZ;  
L->ВКЛКОРZ+400;  
END;
```

706; (Включити радіусний коректор)

```
G17=ВКЛКОРR;  
Q=201;  
END;
```

710; (Виключити радіусний коректор)

```
Q=400;  
END;  
END;
```

(Кінець алгоритму)