

Министерство высшего и среднего специального образования СССР

Московское
ордена Ленина и ордена Трудового Красного Знамени
высшее техническое училище имени Н. Э. Баумана

В. Г. АЛЕКСЕЕВ, А. Н. МАЛОВ

ПРИМЕНЕНИЕ ЦИФРОВЫХ ВЫЧИСЛИТЕЛЬНЫХ
МАШИН ДЛЯ ПРОЕКТИРОВАНИЯ
ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ

Методические указания

Министерство высшего и среднего специального образования СССР

**Московское ордена Ленина и ордена Трудового Красного Знамени
высшее техническое училище им. Н.Э.Баумана**

В.Г.АЛЕКСЕЕВ, А.Н.МАЛОВ

**ПРИМЕНЕНИЕ ЦИФРОВЫХ ВЫЧИСЛИТЕЛЬНЫХ МАШИН
ДЛЯ ПРОЕКТИРОВАНИЯ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ**

Методические указания

Под редакцией А.Н.МАЛОВА

Методические указания по курсу "Вычислительная техника и ее применение при конструировании и производстве ЭВА" издаются в соответствии с учебным планом специальности 0648.

Рассмотрены и одобрены кафедрой П-8 3/IX-1973 г., Методической комиссией факультета П и Учебно-методическим управлением.

Редактор Н.Г.Ковалевокая

Корректор В.Т.Карасева

Заказ 1955

Д-137030 от 31/Х-73г.

Объем 3 3/4 п.л.

Бесплатно

Тираж 250 экз.

Печ. 1973 г.

Ротапринт МВТУ. 107005. Москва, Б-5, 2-я Бауманская, 5.

Введение

Применение электронной вычислительной техники и математических методов в различных областях инженерной деятельности и управления производством становится одним из основных факторов технического прогресса.

В настоящее время в связи с непрерывным ростом качества новых изделий и сокращением сроков их освоения технологическая подготовка производства является одной из первоочередных задач автоматизации управления предприятием.

Технологическая подготовка производства занимает большой удельный вес в общем объеме инженерных работ, особенно при индивидуальном и мелкосерийном производстве. Кроме того, она является информационной базой для планирования и учета производства и основой организации всего производственного процесса.

Ядром технологической подготовки производства являются разработка и нормирование технологических процессов обработки деталей.

Задача автоматизации проектирования технологических процессов изготовления деталей в приборостроении сводится к моделированию на электронно-вычислительных машинах деятельности инженера-технолога по разработке технологической документации. Задачей же моделирования является построение машинного алгоритма (программы) для ЭВМ, при котором машина вела бы себя подобно человеку, занимающемуся моделируемым видом умственной деятельности. При этом не обязательно стремиться к точному копированию работы специалиста, важно получить необходимые результаты, которые заносят в технологическую документацию.

Ряд авторов в основу разработки алгоритмов положили методы "фотографирования" действий технологов и частные методики типизации технологических процессов. Эти разработки ориентированы на существующие методы проектирования технологии и поэтому не содержат расчетов, отражающих современный научный уровень

технологии, а лишь обобщают опыт ее проектирования на различных предприятиях.

Построение машинного алгоритма требует формализации всех этапов проектирования технологии. Возможность такой формализации не вызывает сомнений, однако эта работа весьма трудоемка и степень ее трудоемкости во многом определяется современным уровнем математизации методов проектирования технологии.

Основные сведения о процессах обработки и методах проектирования технологии можно найти в книгах А.П.Соколовского, В.М.Кована, Б.С.Балакшина, Д.В.Чарико, А.Н.Малова, С.П.Митрофанова и многих других советских ученых и инженеров, в частности, некоторые сведения по проектированию технологии приведены в [I]. В указанной литературе обобщен многолетний производственный опыт по механической обработке, что делает эти книги практическими руководствами при проектировании. Следует, однако констатировать, что в указанной литературе не все вопросы проектирования четко разработаны. По ряду вопросов приводимые рекомендации предполагают дополнительные решения технолога, а по ряду вопросов расчетные методы нуждаются в корректировке.

Теория автоматизированного проектирования технологии в приборостроении только создается. Существующая и описываемая в литературе методика проектирования технологических процессов носит ярко выраженный отпечаток приспособленности для "ручного" проектирования и совершенно не учитывает возможностей и особенностей применения ЭВМ.

При автоматизации проектирования технологических процессов с помощью вычислительной машины необходимо решить, каким образом осуществлять:

1) кодирование исходной информации, необходимой для технологического проектирования (чертежи обрабатываемой детали, оборудование, зажимные приспособления, режущий инструмент, вспомогательный и мерительный инструмент, сортамент основных материалов, название технологических операций, переходов и т.д.);

2) формирование операций и определение их оптимальной последовательности;

3) проектирование различных технологических операций;

4) определение оптимальных размеров и способа получения заготовок;

5) печать операционных или расширенных маршрутных технологических карт с операционными эскизами и чертежами заготовок.

В пособии рассматриваются лишь элементы теории автоматизированного проектирования технологии в приборостроении с помощью современной вычислительной техники.

I. Задачи при проектировании технологических процессов и математические методы их решения

Основная задача технологического проектирования заключается в том, чтобы при заданных технических ограничениях спроектировать технологический процесс, обеспечивающий получение с наименьшей себестоимостью требуемых чертежом точности размеров, формы и взаимного расположения поверхностей, чистоты обработки и физико-механических свойств детали.

Исходными данными для этой задачи служат информация об обрабатываемой детали, заданная в виде таблицы или любым другим способом, и программа выпуска деталей. Техническими ограничениями, определяющими возможные варианты структуры технологического процесса и параметры отдельных операций, служат:

состав и техническая характеристика оборудования;

набор универсальной, типовой, групповой и специализированной оснастки, применяемой на предприятии;

применяемость основных материалов, из которых могут изготавливать детали.

В соответствии с основной задачей цель применения математических методов в автоматизации проектирования технологических процессов заключается в получении количественных данных, необходимых для принятия наилучших решений по различным вопросам проектирования технологических процессов.

Среди математических методов, применяемых для этой цели, наибольшее распространение получили: теория множеств, функциональный анализ, теория графов, комбинаторный анализ, теория вероятности и математическая статистика; логическое, нелинейное, динамическое программирование.

Все многообразие задач, которые приходится решать при машинном методе проектирования технологических процессов (ТП), с математической точки зрения можно условно разделить на три вида:

арифметические, логические, геометрические.

Первые два вида задач являются основными при машинном проектировании ТП, а все три вида используются при конструировании оснастки. Арифметические задачи в основном имеют место при анализе и расчете технологических режимов изготовления деталей. Методы решения арифметических задач в зависимости от их сложности могут быть различными. Анализ показывает, что при разработке ТП с помощью ЭВМ наиболее часто используется метод линейного программирования.

Целью логической задачи является выбор наилучшего варианта из множества различных решений, а также проверка истинности (или ложности) различных простых и сложных высказываний, определяющих характеристики параметров конструкции или процесса.

Основной задачей данного вида является выбор оптимального ТП, из множества известных вариантов на основе анализа множества условий. Решением логической задачи должен быть оптимальный метод осуществления ТП или его отдельных частей. При решении логических задач применяются методы теории множеств, теории графов, теории операций и др. Следует отметить, что выделить чисто арифметические или чисто логические задачи практически невозможно, за исключением лишь небольшого числа задач, представленных формулами или логическими уравнениями (правилами).

2. Основные этапы автоматизированного проектирования технологических процессов

Анализ работ в области автоматизированного проектирования технологии (АПТ) изготовления деталей показывает, что процесс автоматизации технологического проектирования сводится к выполнению следующих основных этапов: выбор объекта автоматизации; разработка методики технологического проектирования; разработка системы кодирования исходной информации; алгоритмизация методики технологического проектирования; составление и отладка программы; реализующей технологический алгоритм; внедрение результатов АПТ в конкретных производственных условиях.

3. Выбор объекта автоматизации

Объектом автоматизации, как правило, является отдельная

технологическая задача. Независимо от ее содержания процесс автоматизированного решения задачи начинается с разработки методики проектирования технологии. Такая методика должна учитывать весь опыт, имеющийся в данной области. Качество проектируемой с помощью ЭВМ технологии изготовления деталей определяется эффективностью правил, заложенных в методику решения задачи.

4. Разработка методики технологического проектирования

При разработке АПТ технологические задачи имеют следующие методы решения: 1) индивидуальный метод; 2) метод групповой обработки; 3) метод типизации технологических процессов; 4) метод направленного отбора; 5) метод многоступенчатого синтеза и др.

Первый метод основан на моделировании действия технолога при проектировании индивидуальной технологии.

Разработанные С.П.Митрофановым основы групповой технологии позволяют повысить коэффициент загрузки станков, увеличить производительность, сократить сроки технологической подготовки производства [3]. С использованием ЭВМ резко снижается трудоемкость проектирования групповых технологических процессов.

Для реализации идей типизации проектирования технологических процессов в конкретных условиях того или иного предприятия используют ЭВМ с цифровым методом переработки информации. При составлении алгоритма и программы для ЭВМ в качестве основного параметра выбирают порядок следования операций. С помощью ЭВМ устанавливается принадлежность детали к определенному типу по порядку следования операций. Для каждого типового порядка следования операций есть своя ветвь программы. В ходе реализации команд, входящих в эти ветви, ЭВМ автоматически (по алгоритму) выполняет необходимые действия и выдает результаты в виде набора в определенной последовательности технологических переходов. По каждому переходу указываются обрабатываемые поверхности, их размеры, число проходов, режимы резания, нормы времени, режущий и измерительный инструмент, приспособления. Алгоритм проектирования типовых технологических процессов не является универсальным, увеличение номенклатуры обрабатываемых деталей приводит к его корректировке и переделке.

Блок-схемы основных программ, реализующие данный метод проектирования технологических процессов, приведены в приложении на рис. I-6.

При проектировании методом направленного отбора используется ряд алгоритмов, которые по назначению делятся на две группы: 1) алгоритмы для решения общих вопросов, относящихся к заготовке (выбор типа и метода получения заготовки; назначение маршрута обработки заготовки; выбор станка и приспособления); 2) алгоритмы для решения вопросов, относящихся к отдельным операциям. Алгоритмы обладают некоторой универсальностью, однако исходная информация содержит много упрощений, что не позволяет полностью использовать все преимущества ЭВМ.

Метод многошагового синтеза предусматривает разделение технологического процесса на определенное количество шагов, при котором структура процесса выражается в явном виде [5]. Процесс проектирования происходит последовательно от этапа к этапу, начиная от отделочных операций и кончая выбором формы и размера заготовки.

При многошаговом методе проектирования технологического процесса его структура в целом характеризуется совокупностью этапов (см. табл. I).

Таблица I

Наименование и характеристика этапов технологического процесса

Номер этапа	Наименование этапа	Назначение и характеристика этапов
Э1	Заготовительный	Получение заготовки и ее термообработка
Э2	Черновой	Съем лишних напусков и припусков.
Э3	Термический I	Термообработка - "улучшение", старение
Э4	Получистойой I	Точность обработки - 4-5-й класс, чистота поверхности ≤ 6
Э5	Термический II	Цементация
Э6	Получистойой II	Съем цементационного слоя на поверхностях, предохраняемых от цементации
Э7	Термический III	Закалка, "улучшение"
Э8	Чистойой I	Точность обработки 2-3-й класс, чистота поверхности ≤ 8

Номер этапа	Наименование этапа	Назначение и характеристика этапов
Э9	Термический IV	Азотирование, старение
Э10	Чистовой II	Шлифование поверхностей, предохраняемых от азотирования
Э11	Чистовой III	Точность обработки - I-2-й класс, чистота поверхности ≤ 9
Э12	Гальванический	Хромирование, никелирование и др.
Э13	Доводочный	Получение высокой чистоты поверхностей до II-12-го классов

В каждом из этапов сосредоточены операции обработки различных поверхностей, характеризующихся примерно одинаковой точностью и величиной снимаемого припуска. При формировании операций обработки используются варианты планов обработки элементарных поверхностей. Многошаговому процессу формирования операций на каждом этапе соответствует алгоритм, построенный на основе многократного применения стандартной программы сортировки и упорядочения операций по заданным признакам. После формирования операций технологического процесса каждая из них строится по универсальной блок-схеме. Целевой функцией оптимального многошагового проектирования является минимальная технологическая себестоимость.

Примеры алгоритмов, характерных для данного метода проектирования, приведены в приложении на рис. 7-16.

На рис. I представлена укрупненная схема работы системы проектирования ТП. Доминирующее положение в системе, на наш взгляд, должно занимать проектирование специальных (индивидуальных) технологических процессов.

Такой подход к проектированию позволяет наиболее рационально учитывать как сложившиеся на практике методы проектирования, так и вновь разрабатываемые.

С помощью системы проектирования специальных ТП использование принципов типизации и групповой технологии позволит автоматически пополнять информационно-поисковую систему типовыми и групповыми ТП.

Разработка же самих типовых и групповых ТП должна проводиться на базе программ проектирования индивидуальных ТП, что

позволяет автоматически получить типовые и общие решения для однотипных деталей.

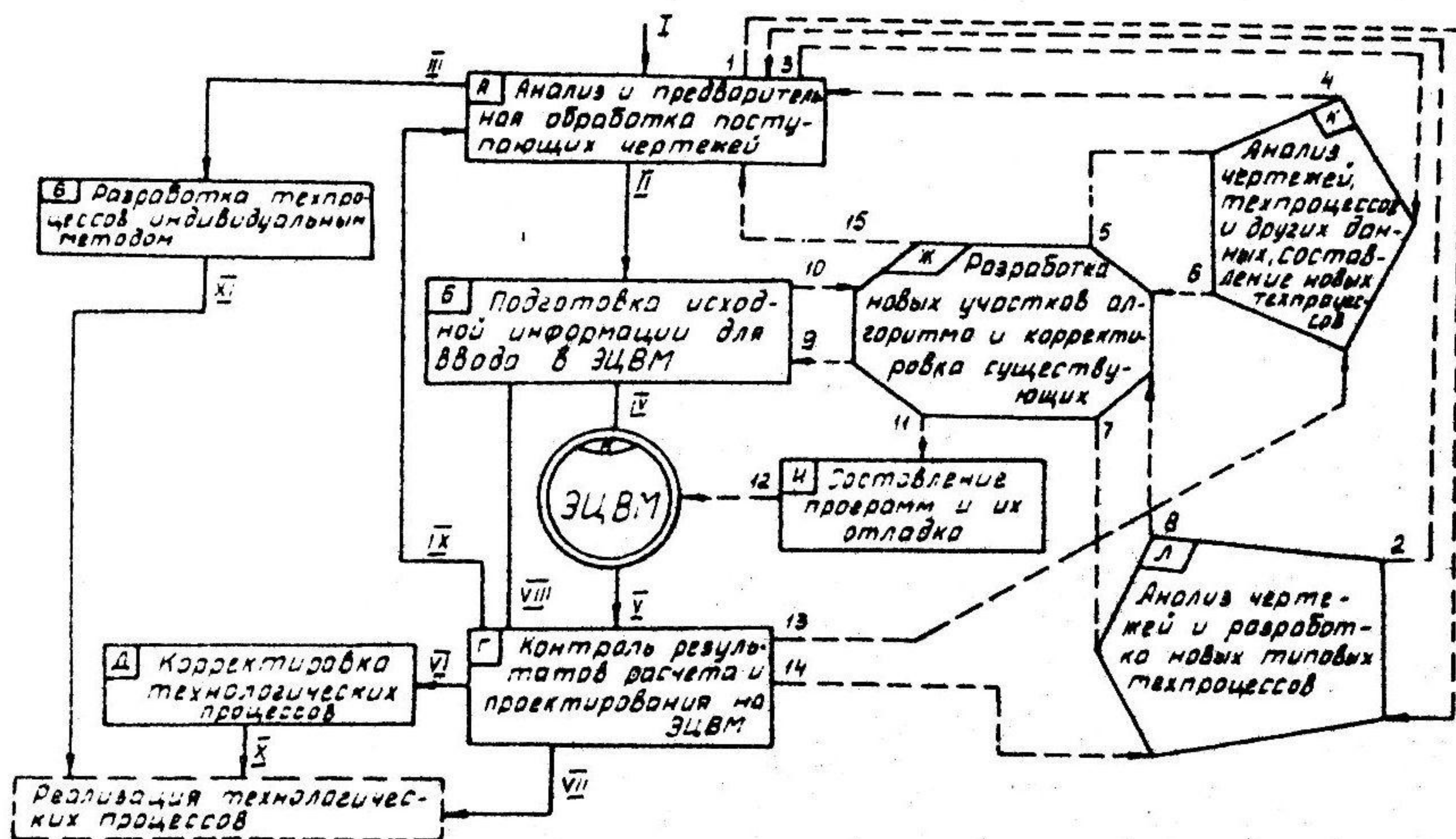


Рис. I. Схема идеализированной кибернетической модели системы подготовки и разработки техпроцессов с использованием ЭЦВМ

5. Разработка системы кодирования исходной информации

При АПТ одним из важных этапов является разработка системы кодирования исходной информации, т.е. представление всей качественной и количественной информации, необходимой для технологического проектирования, в виде, удобном для ввода в запоминающее устройство ЭВМ.

Информация, используемая при АПТ, делится на постоянную и переменную. К постоянной относятся сведения об имеющемся на предприятии оборудовании, оснастке, нормалах на режущий и мерительный инструмент и т.д. Эти сведения многократно используются в алгоритме решения задачи и хранятся вместе с программой в запоминающем устройстве ЭВМ.

К переменной информации относятся сведения, получаемые из чертежа детали. Данная информация вводится в оперативную память ЭВМ в каждом отдельном случае для проектирования технологичес-

кого процесса изготовления конкретной детали. Кодирование и ввод кодов в запоминающее устройство пока еще проводится "вручную". Наиболее целесообразным решением этих вопросов было бы создание автомата для чтения чертежей. Работы в этом направлении ведутся.

Учитывая, что АПТ относится к классу информационно-логических задач, характеризующихся большим объемом исходных данных и результатов решения, можно сформулировать следующие основные требования, которым должна отвечать система кодирования информации:

простота, точность и рациональность ручного описания информации на языке вводных устройств ЭВМ, обеспечивающие минимальное время кодирования и наименьшую вероятность ошибок;

минимальный объем памяти, занимаемой исходной информацией;

способность обеспечить кодирование всех данных, необходимых для разработки алгоритмов и программ автоматического решения конкретной задачи.

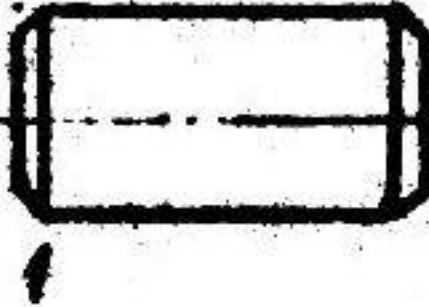

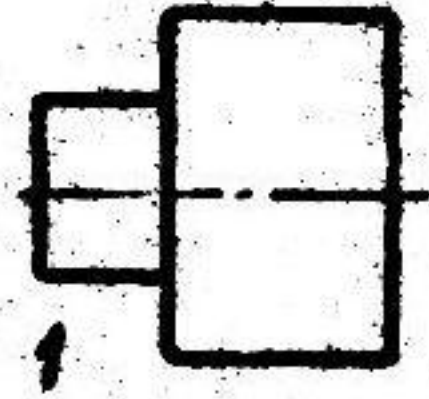
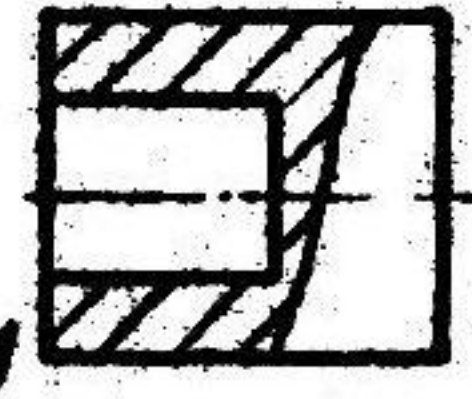
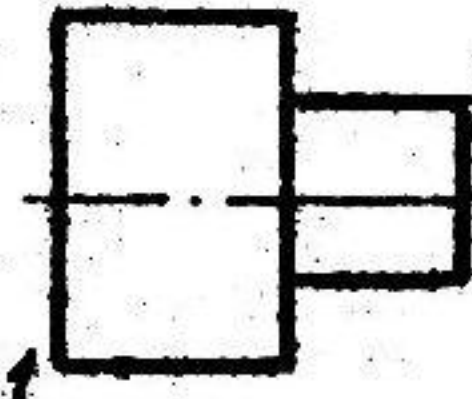
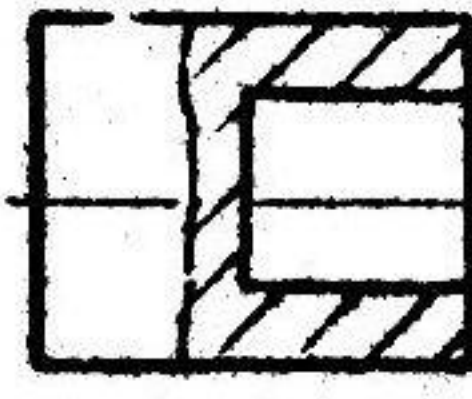
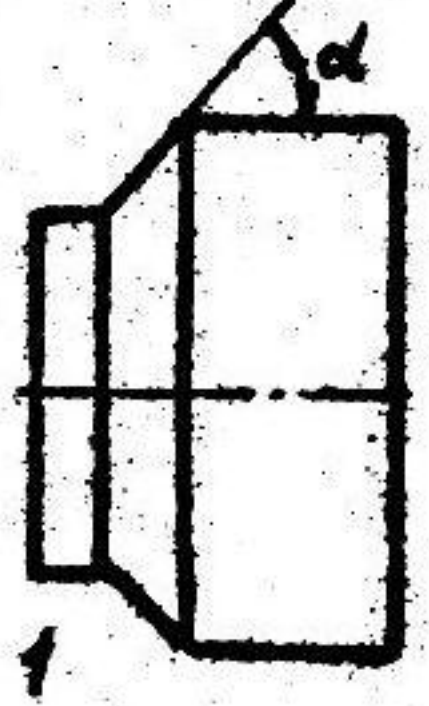
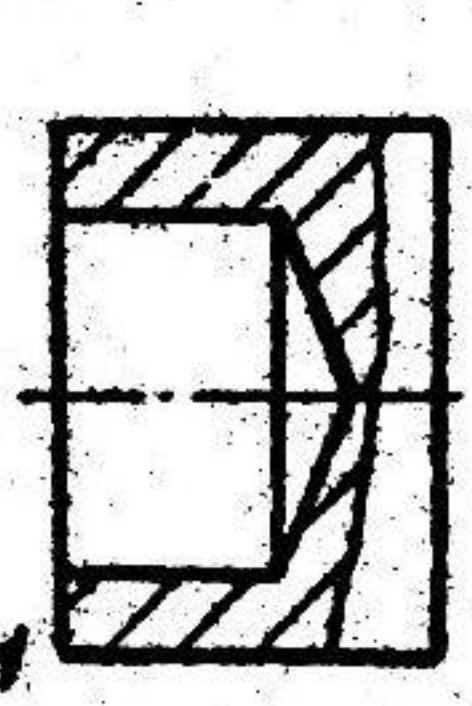
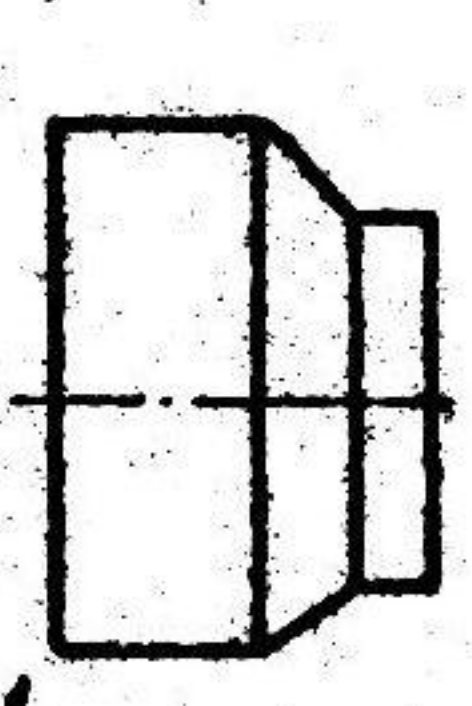
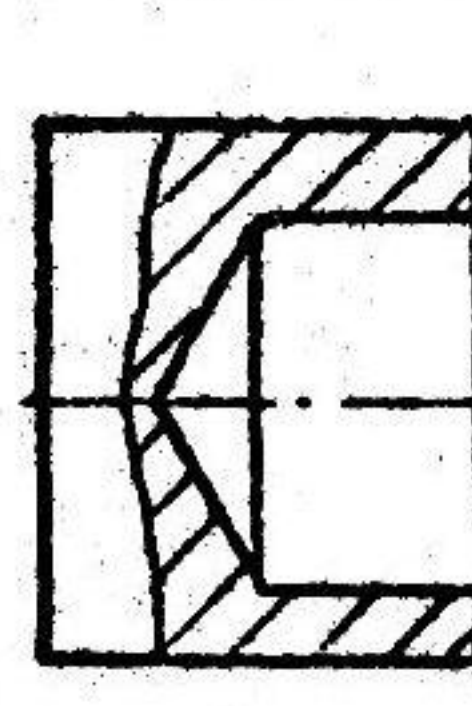
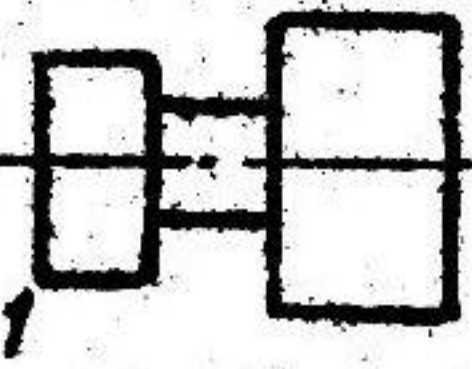
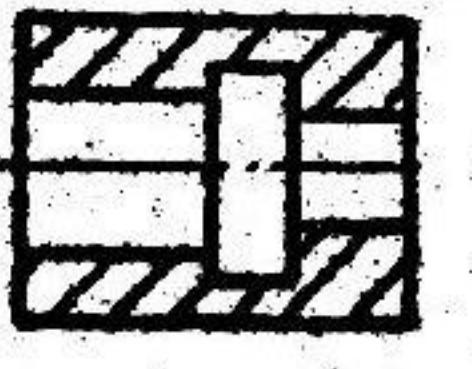
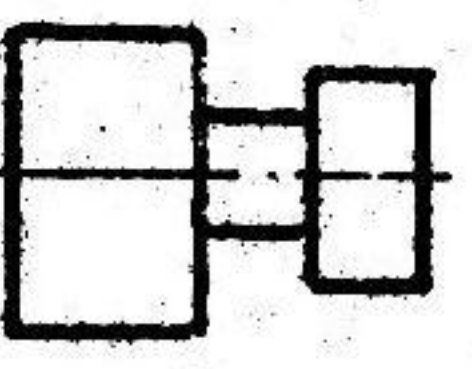
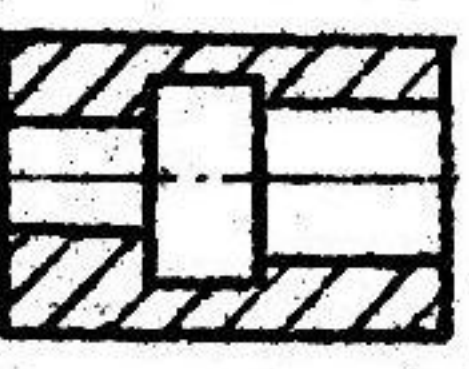
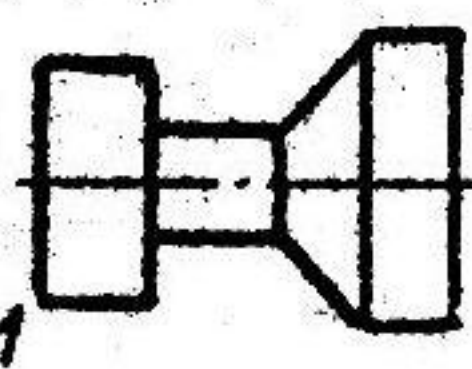
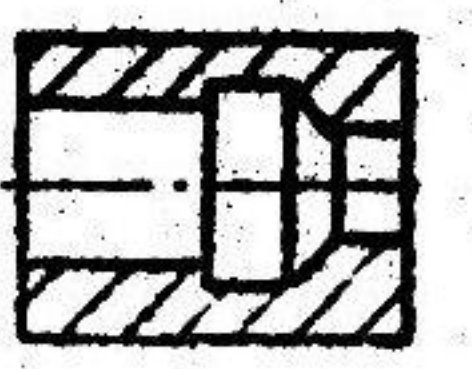
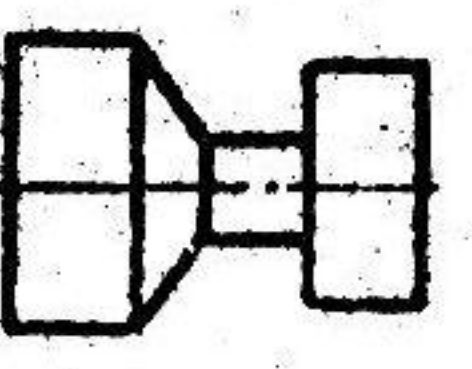
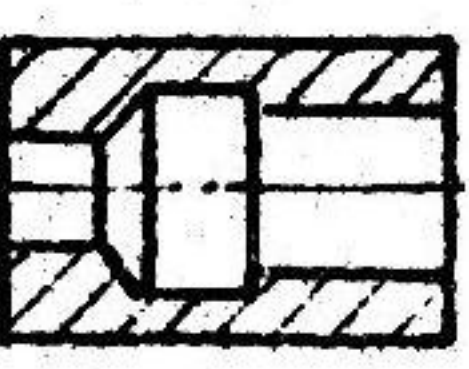
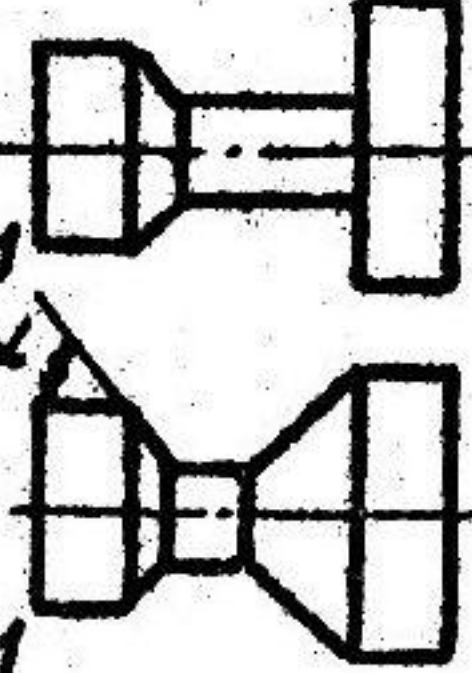
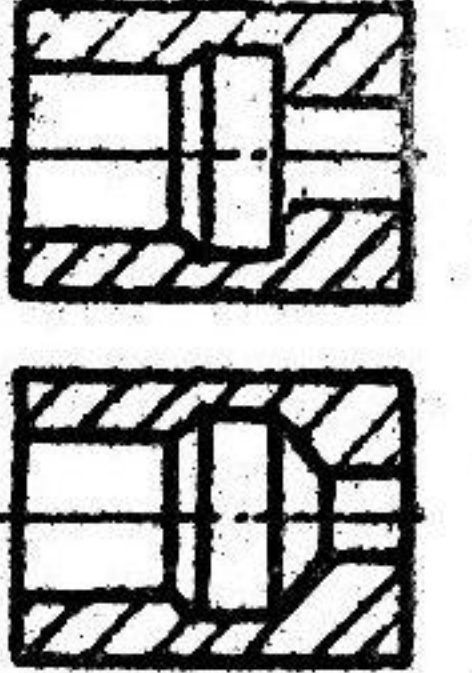
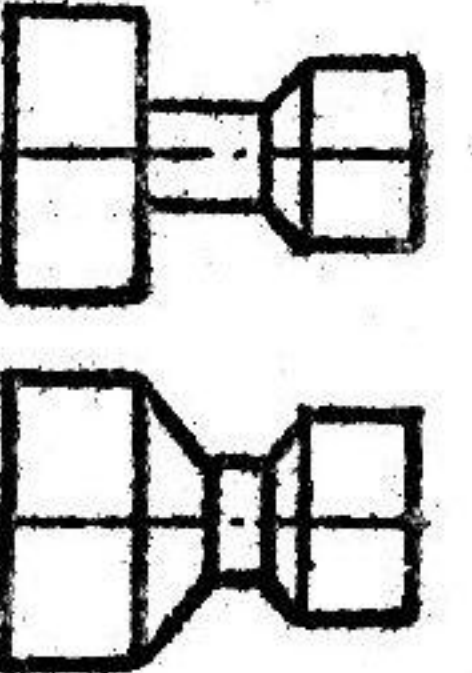
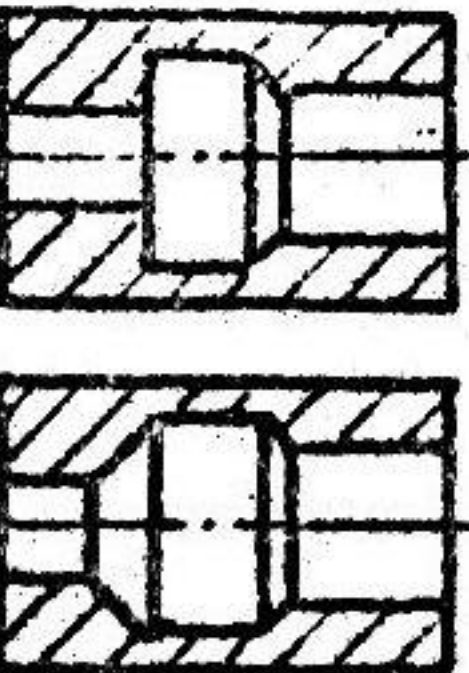
Системы кодирования в зависимости от характера формы закодированной исходной информации разделяются на системы с использованием кодировочной таблицы и с использованием специального информационного языка.

В систему кодирования, использующую кодировочную таблицу, входят: перечень исходных данных и ограничений по ним, кодировочные таблицы с перечнем кодовых обозначений, инструкция по кодированию и заполнению таблицы кодированных сведений (ТКС). ТКС заполняется на каждую деталь и позволяет представлять исходные данные в упорядоченном виде. Упорядоченное представление информации обеспечивается в строго определенной последовательности.

В качестве примера решения задачи кодирования исходной информации в условиях АПТ может служить система кодирования сведений с деталях типа тел вращения (рис.2). По этой системе информация о чертеже конкретной детали, заносимая в ТКС (табл. 2), подразделяется на четыре группы. В первую группу включены сведения, являющиеся общими для всей детали. Вторую группу составляют сведения о заготовке, из которой изготавливается деталь. К третьей группе относятся сведения об отдельных цилиндрических и торцевых поверхностях детали (наружных и внутренних). Здесь содержится информация о диаметральных и линейных

Термические операции		Покрyтия		Виды отверстий		Кл. числ		Кл. точн		Посл. дна	
Табл. 3		Табл. 5		Табл. 6		Табл. 10		Табл. 11		Табл. 12	
Код	Код	Код	Код	Код	Код	Код	Код	Код	Код	Код	Код
Отжиг	400	Оксидирование	430	Нет отверстий	-	∞	-	1	1	A	10
Нормализация	401	Фосфотирование	431	Отверстие одностор.	1	1	1	2	2	B	11
Закалка	402	Воронение	432	Открытое слева	2	2	2a	3		Гр	12
Закалка ТВЧ	403	Пассивирование	433	Открытое сквозное	3	3	3	4		Пр3	13
Отпуск	404	Лужение	434	Отверстие одностор.	4	4	3a	5		Пр2	14
Азотирование	405	Хромирование	435	Открытое слева	5	5	4	6		Пр	16
Цианирование	406	" блестящее	436	Отверстия глухие	6	6	5	7		Пл	17
Цементация	407	" размерное	437	с двух сторон	7	7	7	8		Г	18
Старение	408	Никелирование	438	Заготовки	8	8				Т	19
Улучшение	409	" блестящее	439	Прокат	9	9				Н	20
Терморихтовка	410	Цинкование	440	Круг	10	10				П	21
Шкалы твердости	Табл. 4	Кадмирование	441	Квадрат	11	11				С	22
по Бринелю HB	420	Травление	442	Шестигр.	12	12				Д	23
по Роквеллу HRA	421	Окраска	443	" "	13	13				Х	24
HRB	422		444	" "	14	14				Л	25
HRC	423		445	Труба	15	15				Ш	26
				стенки	16	16				Т	27
					17	17				Пш	28

Группы материалов				Табл. 8		Группы материалов			Табл. 9		Применение
Черные металлы		Цветные металлы		Код		Металлы и сплавы пред. стабилител		Бв кг/мм ²		HB	
Представи-тели	Бв кг/мм ²	HB	Предста-вители	Бв кг/мм ²	HB	Код	Металлы и сплавы пред. стабилител	Бв кг/мм ²	HB	Код	При кодирова-нии группы об-работывае-мости код группы пи-сать:
Сталь А12 А20	40-60	160-168	Бронза бр.КМЦ-3-1 РМЦ15-2	40-55		1	Сталь 15, 20, А-12, ст. 3	30-50		1	из табл. 8 в графу 34
Ст. 10-35, ст. 2, ст. 3, ст. 4, ст. 5	40-75	130-210	Бронза бр. АЖ9-4 АЖН10-44	55-65	130-180	2	Сталь 45	50-60		2	из табл. 9 в графу
Ст. А-30 А-40 40-50, 30Х, 40Х	65-100	130-210				3	Ст. 40А, 40А, 40Х		187-197	3	
Ст. 47 + 410		130-210				4	Ст. Х18Н9Т, Х18Н10Т, 0Х22Н5Т	50-65		4	
Ст. 1Х13, 2Х13, 3Х13, 4Х13	60-70	130-220				5	Ст. 2Х13, 4Х13	80-100		5	
Ст. Х18Н9Т, Х17Н9	50-60	130-190				6	Бронза 0Ф-8, 8-0,15; Л-85	35-55		6	
Ст. ШХ15, ШХ5, 4Х9С2	100-125	180-250				7	Алюм. сплавы Д1, АМЦ	30-47		7	
			Медь М1-М3 Латунь А359-1 Л-62	25-35		8	Винипласт, эбонит	4-8		8	
			Ал. сплавы Д1, Д16, АЛ-2	18-25		9				9	
Оргстекло эбонит	4-8					10				10	

Кодирование видов поверхностей					Табл. 13	
α°	Открытые слева		Код	Открытые справа		Код
	наружные	внутренние		наружные	внутренние	
0			10			10
90			31			11
45			32			12
60			33			13
75			34			14
30			35			15
15			36			16
*	прочие		37	прочие		17
α°	Закрытые слева		Код	Закрытые справа		Код
	наружные	внутренние		наружные	внутренние	
90 x 90			41			21
45 x 90			47			27
90 x 45			42			22
90 x 60			43			23
90 x 75			44			24
90 x 90			45			25
90 x 15			46			26
*	прочие		48	прочие		28
*	Резцы будут перетачиваться					

размерах, о точности и чистоте обработки поверхностей. В четвертую группу включены сведения об особенностях (фасках, канавках, резьбах и т.д.) отдельных поверхностей детали.

Информация, получаемая из чертежа детали (см.рис.2), заносится в ТКС (см.табл.2) либо непосредственно с чертежа (например, диаметральные и линейные размеры отдельных поверхностей, параметры особенностей и т.п.), либо в соответствии со специально составленными таблицами для кодирования нечисловых данных (термообработка, покрытие и т.д.), причем кодирование в этом случае осуществляется по правилам, исключающим различное толкование таблиц (табл.3-13). Наиболее простой способ кодирования нечисловой информации состоит в присвоении порядковых номеров предусматриваемым наименованиям или понятиям. Например, 1 - резец проходной; 2 - резец расточной; 3 - центровочное сверло и т.д.

Однако во многих случаях более целесообразно применять для кодирования многозначные числа, придавая каждому разряду особое смысловое значение. В табл.14 показан один из возможных вариантов кодирования инструментов.

Таблица 14

Таблица для формирования многозначных кодов наименования инструментов

Наименование инструмента	Шифр	Вид инструмента	Шифр	Материал инструмента	Шифр
Резец	1	Проходной	1	Быстрорежущая сталь	1
Сверло	2	Подрезной	2	Углеродистая сталь	2
Фреза и т.д.	3	Спиральное и т.д.	3	Твердый сплав и т.д.	3

Используя приведенные шифры, можно определить, например, что код 1 2 3 означает "резец подрезной из твердого сплава".

Принципы кодирования информации о цилиндрических и торцевых поверхностях детали аналогичны принципам создания комплексной детали в условиях группового производства. Деталь условно разбивается на отдельные участки секущими плоскостями, проходящими по торцевым поверхностям перпендикулярно продольной

оси детали. Следы этих плоскостей называются линейными границами.

Участок цилиндрической поверхности, заключенный между линейными границами как по наружным, так и по внутренним поверхностям, должен иметь только один диаметр без ступеней, одну и ту же точность и чистоту обработки. Кодированию подлежит информация, получаемая с каждого отдельно взятого участка детали.

После того как в кодировочную таблицу внесена вся необходимая информация о детали и заготовке, процесс кодирования считается законченным. Закодированные данные переносятся с кодировочной таблицы на перфоноситель (перфокарты или перфолен-ты) и при его помощи вводятся в ЗУ ЭВМ. К этому времени программа, реализующая алгоритм технологического проектирования, уже должна быть занесена в ЭВМ и храниться на магнитной ленте.

В основу второго вида системы кодирования положено использование специально разработанных для этой цели языков. Исходные данные при АПГ, записанные в терминах языка, составляют языковую программу. Отличие второго вида системы кодирования заключается в том, что вместо кодировочных таблиц определяется перечень терминов и правила синтаксиса и семантика языка (синтаксис – система правил образования конструкций языка из терминов; семантика – система правил использования конструкций языка).

Для эффективного решения задач кодирования информационный язык должен быть: 1) простым, чтобы кодирование могли выполнять операторы, не имеющие высокой квалификации, но обеспечивающие минимально возможное время кодирования и наименьшую вероятность ошибок; 2) точным, полным, однозначно отображающим структуру исходной информации, и кратким, чтобы объем памяти ЭВМ, занимаемый исходной информацией, был минимальным; 3) универсальным, приемлемым для кодирования сведений о деталях различной формы; 4) гибким, допускающим внесение отдельных изменений и дополнений отображения специфических особенностей отдельных объектов; 5) целенаправленным, т.е. содержать только ту информацию, которая необходима для совершенно конкретных целей проектирования.

Существует несколько систем кодирования данного типа.

Первая система – система, в которой кодируются непосредственно пространственные образы – поверхности и их взаимное положение в трехмерном пространстве.

Вторая система кодирования базируется на применении стандартных и нормализованных элементов; кодирование информации о геометрии детали основано на кодировании формы элементов и взаимного расположения их в пространстве. Положение элементов в пространстве определяется линейными и угловыми размерами, связывающими их с базами; с каждым элементом конструкции связывается прямоугольная система координат. Точка начала координат (привязочная точка) для каждого нормализованного элемента принимает положение, соответствующее назначению элемента, принятой технологии его обработки и т.д.

Например, для цилиндрических, конических и резьбовых поверхностей точка начала координат лежит на оси поверхности на том торце, откуда начинается обработка.

Целесообразно направление осей координат выбирать таким, чтобы координатные плоскости совпадали с конструкторскими, технологическими, сборочными и другими базами. Это не всегда удастся сделать, поэтому при кодировании указываются номер базы, от которой задается тот или иной размер и способ задания размера. Применение этой системы кодирования геометрической информации дает возможность разрабатывать простые алгоритмы с использованием большого числа стандартных подпрограмм и удобные для автоматического программирования. Кодирование точности взаимного положения элементов конструкции в пределах, заданных чертежом или техническими условиями, на основе анализа размерных цепей детали осуществляется с помощью таблиц.

Третья система кодирования информации о деталях основана на кодировании пространственных объектов, конструкция которых представляется стандартными элементами, – элементарными обрабатываемыми поверхностями (ЭОП). Все ЭОП делятся на два ряда: охватывающие (шейки валов) и охватываемые (отверстия, пазы). Охватывающим ЭОП присваивается код "+", а охватываемым код "-". Каждый род поверхности подразделяется на семь классов, классы – на типы, типы – на виды. Размеры поверхностей и их взаимное расположение кодируются в основной прямоугольной системе координат аналогично предыдущей системе кодирования (но вместо баз ука-

зываются ЭОП). Все поверхности делятся на основные, поверхности первого ранга, второго и т.д. Основные поверхности образуют основной контур детали. Поверхности первого ранга располагаются на основных поверхностях, поверхности второго ранга на первичных и т.д.

Выбор того или иного вида системы кодирования определяется содержанием исходной информации. Если наибольший удельный вес составляют данные геометрического характера, определяющие форму и размеры детали, то целесообразней применить систему кодирования, использующую кодировочную таблицу. В случае, когда преобладает информация конструктивного и экономического характера, передаваемая словесным текстом, используется система кодирования второго вида.

Следует заметить, что принятая система кодирования непосредственно влияет на сложность алгоритма. Например, если в системе кодирования предусмотреть указания технолога, которые заранее определяют некоторые технологические решения, то, очевидно, алгоритм упростится. Эта возможность часто используется для упрощения алгоритмов, когда машинное решение какой-либо задачи оказывается слишком сложным, а технолог может решить ее просто и быстро.

6. Алгоритмизация методики технологического проектирования

Алгоритмизация методики технологического проектирования состоит в разработке и представлении в том или ином виде системы формальных правил, которые однозначно определяют процесс решения всех технологических задач. На данном этапе может быть качественно улучшена методика технологического проектирования путем использования математических методов, позволяющих выбирать оптимальный вариант решения всей задачи.

Разрабатываемые алгоритмы должны быть достаточно универсальными как в отношении проектирования технологии обработки деталей различных классов, так и в отношении их применения на различных предприятиях с мелкосерийным характером производства.

Алгоритм, который будет реализован на ЭЦВМ, должен включать такие действия, которые способна выполнять машина. Любое сложное умозаключение должно быть выражено цепочкой простых дейст-

вий. Совершенно очевидно, что алгоритм не может строиться только на основе интуиции, склонностей или опыта отдельного технолога. Каждое действие, предусмотренное в алгоритме, должно быть научно обоснованным итогом работы коллектива. Накопленный опыт показал, что составление технологических алгоритмов является сложной задачей. Прежде всего требуется обобщение технологического опыта для выработки на каждом этапе проектирования оптимальных решений. Эта сама по себе очень важная научная проблема должна получить наиболее четкое решение при составлении алгоритмов.

Алгоритмизация технологического проектирования является в настоящее время одним из важнейших направлений развития технологии как научной дисциплины. Действительно, при составлении алгоритмов в наибольшей степени учитываются последние достижения науки и техники, происходит обобщение этих достижений, на каждом этапе проектирования максимально используются математические методы, в крупном масштабе решаются вопросы оптимизации решений.

Построение технологических алгоритмов базируется на функциональных и логических зависимостях, определяющих структуру и параметры технологического процесса с учетом конструктивных данных обрабатываемой детали, программы выпуска, реальных условий производства и выбранного критерия оптимальности (например, минимальная себестоимость изготовления детали).

Процесс алгоритмизации начинается с составления общей блок-схемы решения задачи технологического проектирования, состоящей из отдельных логически законченных частных алгоритмов, расположенных в последовательности, необходимой при проектировании технологии изготовления детали (рис. 3, 4). В свою очередь, каждый частный алгоритм развертывается в виде блок-схемы. Характер связей в блок-схеме определяет правила и приемы решения частной задачи.

Рассмотрим для примера простой частный алгоритм, определяющий необходимость в развертывании отверстия круглой детали. При этом следует учесть, что многие вопросы проектирования решены на предыдущих стадиях при помощи других частных алгоритмов. Алгоритм представлен блок-схемой (рис. 5), где каждый прямоугольник определяет элементарное действие, которое необходимо совершить, чтобы осуществить выбор инструмента. Прямоугольники связаны стрелками, которые указывают последовательность

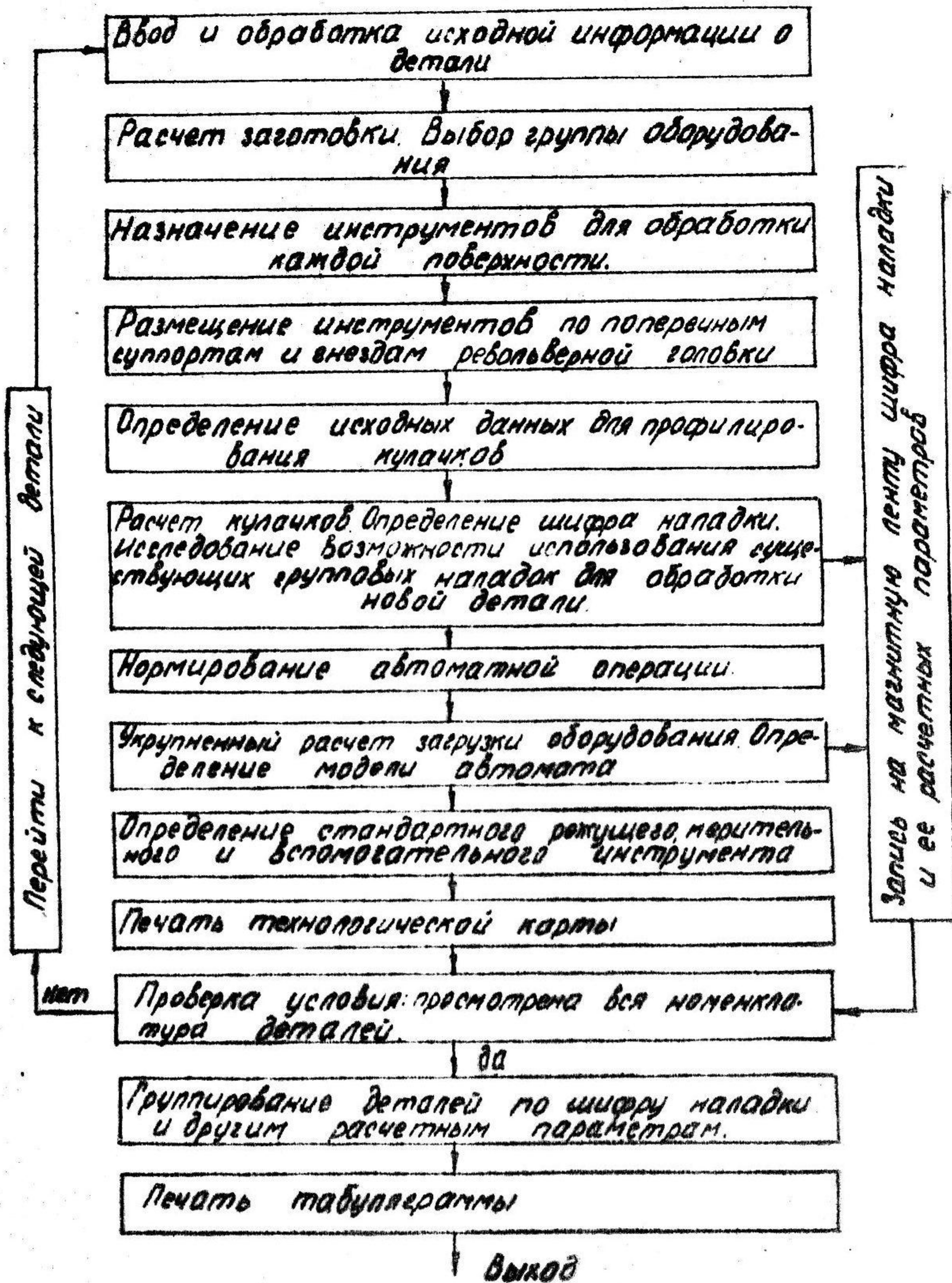


Рис.3. Укрупненная схема технологического алгоритма

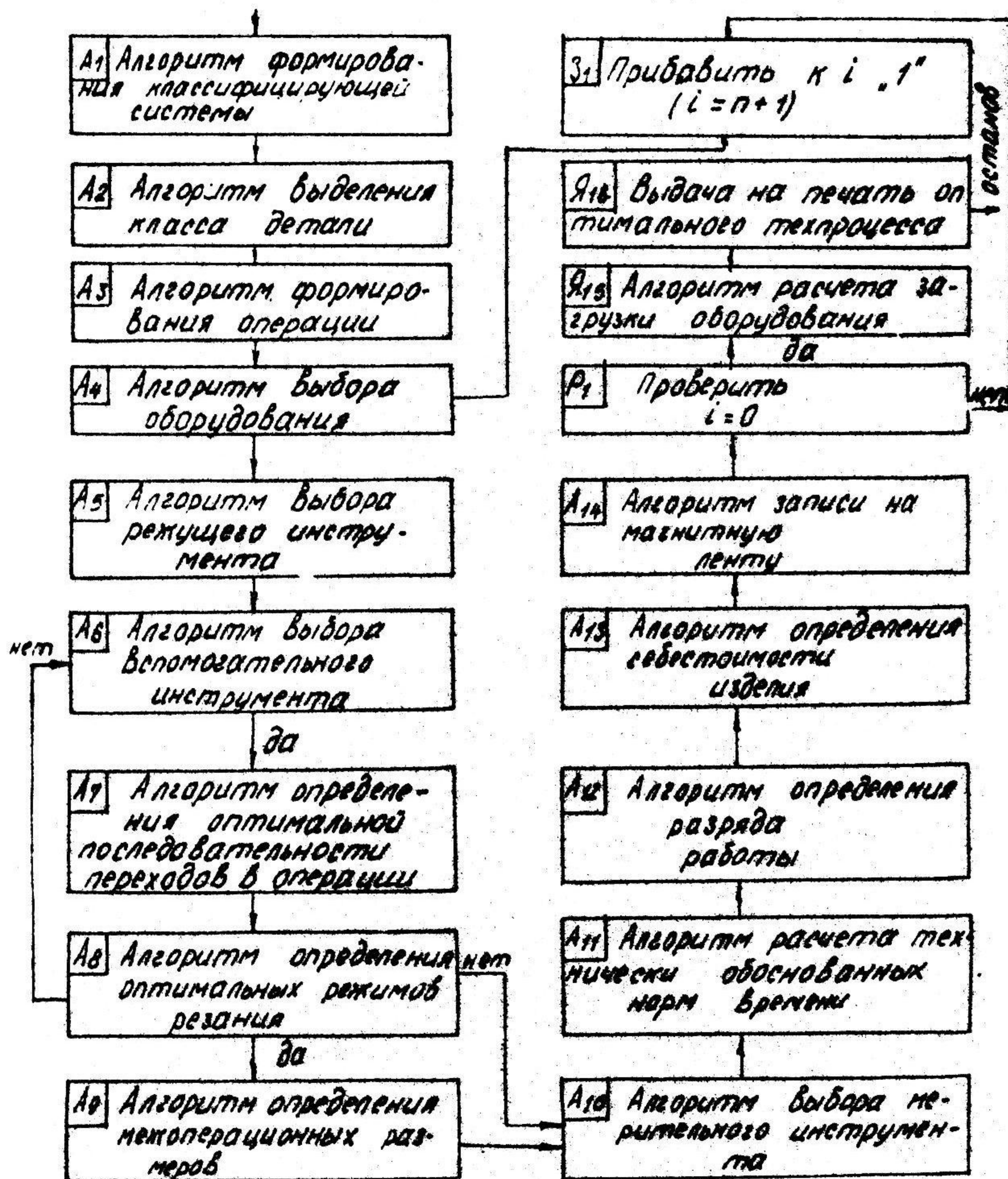


Рис. 4. Блок-схема алгоритмов проектирования операционной технологии

действия. Действия по алгоритму начинаются со стрелки "Вход", указывающей начало работы. Заканчивается алгоритм стрелкой "Выход", которая представляет результаты работы по данному алгоритму. Из каждого прямоугольника, предполагающего выполнение логической операции, выходят две стрелки. Одна обозначается словом "нет". Эта стрелка используется, когда логическое условие не выполняется. Стрелка "да" используется при выполнении условия. Такая операция называется условным переходом, так как дальнейшая последовательность действий определяется результатом проверки логического условия. Из некоторых прямоугольников выходит одна стрелка. Значит, выполнение операции не предусматривает логического заключения. Здесь осуществляется безусловный, т.е. раз и навсегда установленный, переход от одного действия к другому.

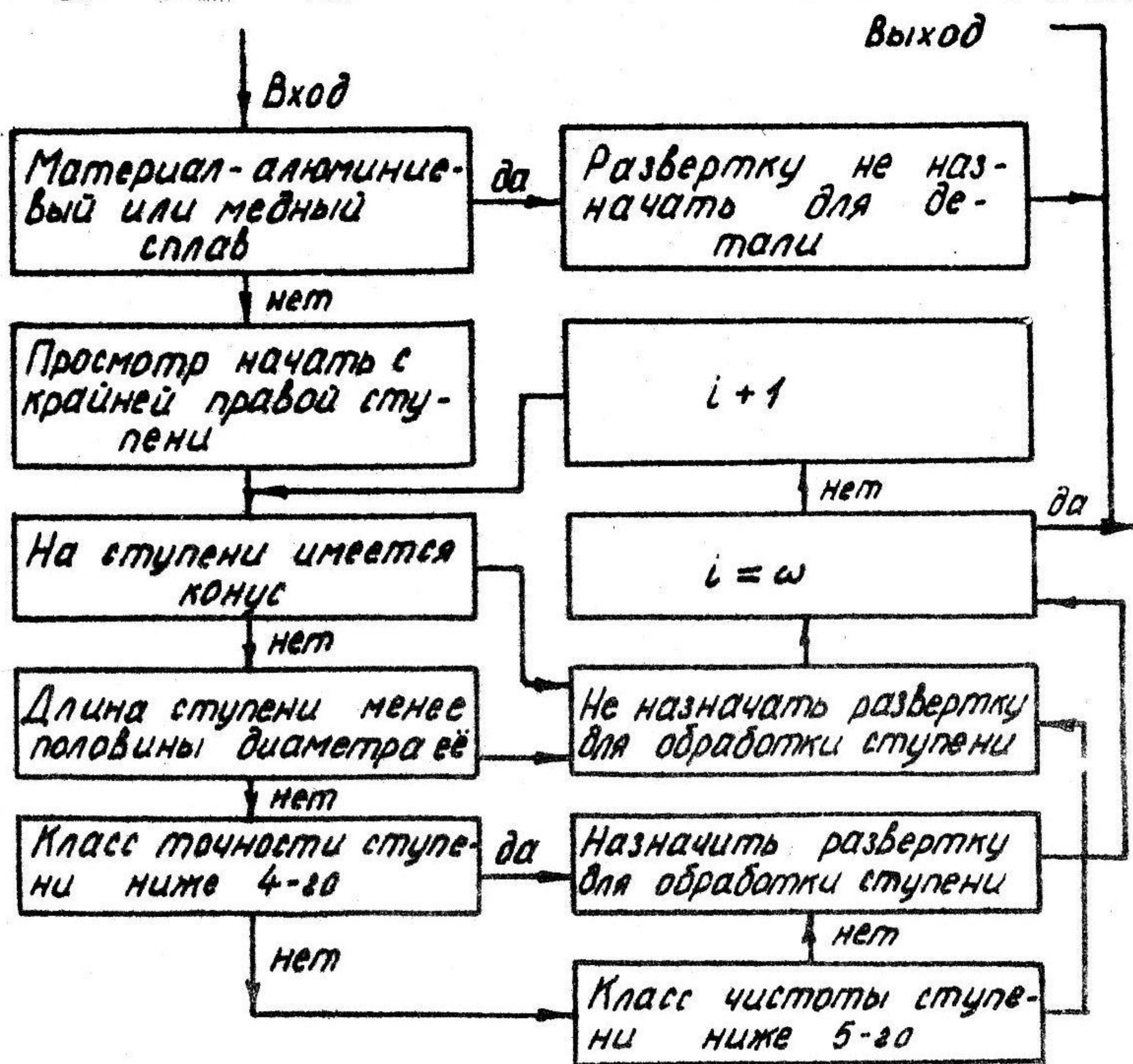


Рис.5. Блок-схема частного алгоритма назначения развертки

В схеме имеются следующие условные обозначения: i — номер ступени отверстия, к которой относится размер; ω — наибольший номер ступени, доступной для обработки на токарно-револьверном автомате, устанавливается на предварительных стадиях проектирования.

Стрелка "Вход" приводит к первому блоку, в котором формируется ответ на вопрос: является ли материал детали алюминиевым или медным сплавом? Если ответ утвердительный, то развертывание не назначается. В случае отрицательного ответа по стрелке "нет" приходим к первому блоку, который определяет общую последовательность просмотра ступеней отверстия. Первой рассматривается крайняя правая ступень, а затем последовательно все остальные. Для первой ступени последовательно формируются ответы на вопросы: имеется ли конус, меньше ли длина ступени половины диаметра и т.д. В зависимости от характера ответа формируется решение о необходимости развертывания. В любом случае стрелки сходятся в прямоугольник проверки логического условия $i = \omega$. Это означает, что номер рассматриваемой ступени сравнивается с номером конечной ступени, доступной для обработки. Если имеется всего одна ступень, то условие $i = 1$ будет сразу выполнено, и по стрелке "да" попадаем на "выход". Этим заканчивается последовательность действий, определяемых алгоритмом для рассмотренного простейшего сочетания параметров детали. В том случае, если имеется не одна, а несколько поверхностей, необходимо повторить указанные действия для каждой из них. Чтобы повторить эти действия для каждой поверхности, используется операция увеличения номера рассматриваемой поверхности на единицу $i + 1$. После этой операции стрелка ведет к блоку проверки наличия конуса, и таким образом осуществляется повторение действий для каждой поверхности, пока не будет выполнено условие $i = \omega$ и все поверхности не будут просмотрены.

Общий алгоритм проектирования состоит из большого числа частных алгоритмов. Характер связи частных алгоритмов между собой определяет структуру алгоритма. Структурные варианты весьма многочисленны, в связи с чем возникает задача выбора оптимальной структуры. Прежде всего блок-схемы частных алгоритмов организуются между собой по иерархическому принципу. Блок-схемы первого, самого высокого, уровня дают принципиальные решения и расчеты в первом приближении, например выбор загото-

вок, назначение последовательности обработки и т.п. В отличие от приведенной выше блок-схемы частного алгоритма (см.рис.5) в отдельных блоках принципиальных схем допускается записывать словами названия выполняемых действий с указанием номеров элементарных блок-схем, в которых подробно описываются эти действия. Возможные варианты такого описания показаны ранее на рис. 3,4.

Дальнейшее уточнение и детализация решений осуществляются последовательно блок-схемами низших уровней, например блок-схема расчета размеров заготовки детали уточняется блок-схемой назначения припусков. Такое построение алгоритма открывает большие возможности для корректировки блок-схем в зависимости от результатов статистического анализа, который устанавливает связь между частотой использования определенного участка алгоритма и его сложностью, а также позволяет корректировать отдельные блок-схемы, не нарушая общей структуры алгоритма, при учете особенностей конкретного производства. Благодаря структурным усовершенствованиям можно упростить алгоритм за счет рациональной связи между исходными, промежуточными и окончательными данными расчетов. Задача состоит в определении минимального количества промежуточных результатов, использование которых позволяет выполнить формальные преобразования информации, помещаемой в технологической операционной карте.

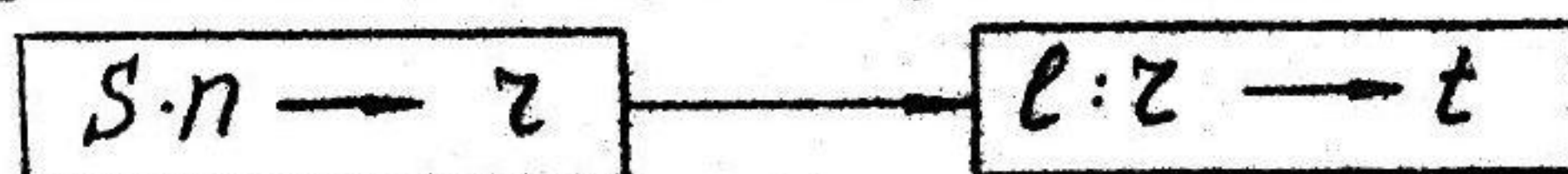
Основные требования к оформлению технологических алгоритмов сводятся к следующему: форма алгоритмов должна быть удобна для выражения технологических правил и расчетов и для программирования.

Во избежание неправильного понимания и истолкования программистом сути алгоритмов при составлении последних необходимо по возможности использовать те же приемы и правила, которые применяются при составлении программы, т.е. смысловая сущность "языка" для описания алгоритма должна быть сходна с сущностью "языка", используемого при составлении программ. Чем глубже технолог будет знать особенности программирования и конкретной модели ЭВМ, тем правильнее и экономичнее могут быть разработаны и описаны алгоритмы.

В блок-схемах частных алгоритмов для обозначения выполняемых действий используется общепринятая математическая символика (+, -, :, x, = и др.). Кроме того, часто применяется знак пересылки, например ($a \rightarrow b$), т.е. число, находящееся в ячейке "a",

пересылается в ячейку "b", и знак добавления ($a \rightarrow b$), т.е. число, находящееся в ячейке "a", добавляется к числу, находящемуся в ячейке "b". При пересылке число, ранее находившееся во второй ячейке "b", стирается и вместо него записывается новое число, которое после выполнения указанного действия сохраняется в двух ячейках ("a" и "b"). При добавлении ($a \rightarrow b$) число, находящееся в ячейке "a", также сохраняется. При необходимости очищения какой-либо из ячеек, например ячейки "a", в схеме пишется обозначение $0 \rightarrow a$.

В качестве примера использования указанной символики можно привести блок-схему определения времени резания при токарной обработке детали. Если в ячейки оперативного ЗУ, условно обозначенные ℓ , S и n , будут соответственно засланы конкретные значения длины обработки в мм, подачи инструмента в мм/об и числа оборотов шпинделя в минуту, то решение поставленной задачи можно представить в виде двух блоков:



Здесь z – вспомогательная ячейка оперативного ЗУ, t – ячейка, отведенная для хранения полученного результата.

Обычно в практике разработки алгоритмов блок-схемы составляются из значительно большего количества блоков. Их количество особенно увеличивается при использовании действий сравнения, которые применяются при операциях с числовыми и абстрактными таблицами.

В процессе расчетов, связанных с проектированием технологических процессов, большое место занимают действия с числовыми таблицами. К заранее составленным и вводимым в память ЭЦВМ относятся таблицы с паспортными данными станков, с характеристикой применяемых материалов, с размерами инструментов и приспособлений, с данными о режимах обработки, нормах времени и др. Обычно применяются одномерные или двумерные таблицы; в одномерных таблицах искомая величина зависит от одного параметра (аргумента), в двумерных таблицах – от двух. Например, в одномерной табл.15 выбираемый поправочный коэффициент зависит только от стойкости инструмента, а в двумерной табл.16 скорость резания выбирается в зависимости от диаметра сверла и от подачи.

При разработке технологических процессов используются и многомерные таблицы, в которых искомая величина зависит от трех и

более параметров. Но такие таблицы легко преобразуются в несколько двумерных таблиц.

Таблица 15

Поправочные коэффициенты скорости резания для проходных резцов

Скорость резания T , мин	30	60	100	120	180	240	300
Поправочный коэффициент K	1,27	1,09	1	0,96	0,89	0,84	0,8

Таблица 16

Скорости резания для сверления отверстий

Диаметр сверла, мм	Подача, мм/об										
	0,1	0,12	0,14	0,16	0,18	0,2	0,22	0,25	0,28	0,3	0,35
Скорость резания, м/мин											
10	30	27	24	22	20	19	18	17	15	14	12
12	33	29	26	24	22	20	19	18	16	15	13
14	35	31	28	25	23	22	21	19	18	17	16
16	37	32	29	27	24	23	22	21	20	19	17
18	38	33	30	28	25	24	23	22	21	20	18
20	40	35	31	29	26	25	24	23	22	21	19
25	44	39	34	32	29	27	26	25	24	23	21
30	46	42	36	34	31	30	28	26	25	24	22
35	48	43	38	35	33	31	29	28	26	25	23

При разработке технологических процессов используются и многомерные таблицы, в которых искомая величина зависит от трех и более параметров. Но такие таблицы легко преобразуются в несколько двумерных таблиц.

Для описания действий, связанных с использованием числовых таблиц, при составлении блок-схем должны указываться номер таблицы; номер вертикальной строки (графы) и номер горизонтальной строки. Эту запись принято оформлять следующим образом:

Номер таблицы

< Номер графы >

Номер строки

Например, выражение $\langle \text{стр } 5 \rangle_3^{16} \rightarrow a$ означает, что число, находящееся в графе 5 третьей строки табл. 16, должно быть заслано в ячейку "a".

При разработке алгоритмов, описывающих действия с числовыми таблицами, могут возникнуть следующие задачи: по заданному значению стойкости инструмента в минутах T определить и заслать в ячейку K величину поправочного коэффициента, или по заданному значению подачи S и по диаметру сверла (зенковки) d определить и заслать в ячейку V величину скорости резания. Алгоритмы решения этих задач, представленные в виде блок-схем, показаны на рис. 6. Блок-схемы задач № 1 и 2 имеют соответственно пять и девять блоков.

Для удобства приводимого ниже пояснения каждый из блоков в этих схемах пронумерован, что не обязательно делать в практике решения аналогичных задач. В блок-схеме задачи № 1 в блоках, отмеченных соответствующими номерами, выполняются следующие действия:

1. Засылается 0 (нуль) во вспомогательную ячейку R .
2. К числу, находящемуся в ячейке R , добавляется единица ($R \leftarrow R + 1$).

3. Проводится сравнение заданной величины T с числом, находящимся в табл. 15 в первой строке и графе R . Число R в процессе расчета увеличивается от 1 до последнего номера графы в таблице.

Таким образом осуществляется просмотр всей таблицы. Рассматриваемый блок является логическим и имеет выход по двум стрелкам "да" и "нет". Если значение T меньше или равно найденному в таблице, то дальнейшие расчеты выполняются по блоку 5. Если T больше табличной величины, то выполняется действие, указанное в блоке 4.

4. Величина R сравнивается с наибольшим количеством граф табл. 15, т.е. с числом 7. Если $R \neq 7$, то выполняется действие, указанное в блоке 2, в котором к значению добавляется единица. Если $R = 7$, то выполняется действие, указанное в блоке 5.

5. Число, расположенное в графе с найденным значением R , из второй строки табл. 15 пересылается в ячейку K , где хранится найденное значение поправочного коэффициента скорости резания, для последующего использования.

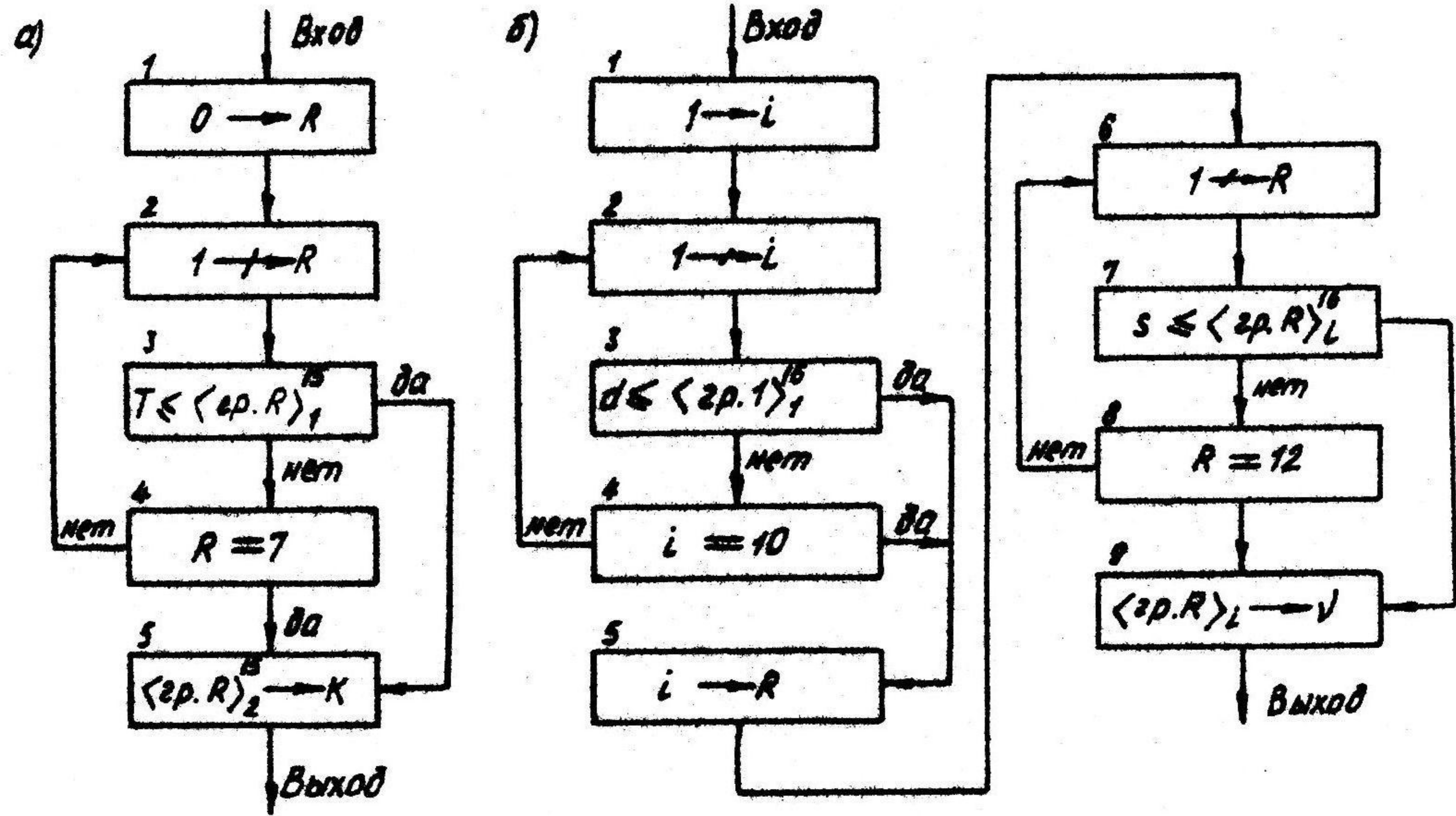


Рис.6. Блок-схема алгоритма операций с числовыми таблицами: а - блок-схема задачи № 1 для определения поправочного коэффициента скорости резания для проходных резцов; б - блок-схема задачи № 2 для определения скорости резания при сверлении.

Особенностью структуры блок-схемы задачи № 1 является организация в ней одного цикла расчетов. Под циклом в расчетах понимается многократное повторение действий, указанных в одном или нескольких блоках, с изменяющимися исходными данными. Внешне наличие цикла в блок-схеме определяется по стрелкам, соединяющим замкнутым кольцом группу блоков. Как видно из рис. 6 б, в блок-схеме задачи № 2 организованы два последовательных цикла расчетов. В первом из этих циклов определяется номер строки i табл. 1б, во втором — номер графы R той же таблицы. Поскольку структура блок-схемы задачи № 2 аналогична структуре, описанной выше блок-схемы задачи № 1, можно не останавливаться на подробных пояснениях ее. Из приведенных примеров составления блок-схем очевидно большое значение организации циклов. Как правило, с применением циклов блок-схемы можно составить более компактными и менее трудоемкими для программирования. Переадресация в цикле может осуществляться не обязательно на единицу. В зависимости от формы, содержания и структуры используемой числовой таблицы можно предусмотреть и соответствующим образом оформить добавление к рассматриваемой переменной величине любого целого положительного числа. В некоторых случаях необходимо осуществлять поиск чисел в таблицах в обратном порядке, тогда переадресация осуществляется с добавлением к ранее зачисляемому наибольшему значению переменной величины отрицательного числа, например $(-1 \rightarrow R)$.

Несмотря на простоту составления и программирования блок-схем, связанных с использованием числовых таблиц, считается более целесообразным аппроксимировать такие таблицы, т.е. найти математическую формулу, по которой с достаточной степенью точности можно вычислять значение искомой величины.

Для упрощения решения вопросов, связанных с рациональным использованием ЭУ ЭВМ, а также для большего удобства составления и описания элементарных блок-схем целесообразно создавать абстрактные таблицы, аналогичные ячейкам оперативного ЭУ, которые используются для внесения в них переменной информации. В отличие от описанных ранее числовых таблиц абстрактные таблицы не заполняются числами. При составлении блок-схем предполагается, что в этих таблицах могут находиться любые числа исходной, промежуточной и искомой информации.

Примеры оформления абстрактных таблиц для разработки алгорит-

мов показаны на рис.7. Каждая из трех приведенных таблиц имеет по три графы и по пять строк. При разработке алгоритмов количество таблиц, а также число граф и строк в них может быть велико. Количество их устанавливается **технологом** – разработчиком алгоритма. Пользуясь абстрактными таблицами, можно составить блок-схемы для решения самых разнообразных математических и логических задач, встречающихся при технологическом проектировании. Основой методики решения большинства элементарных технологических задач служит организация в блок-схемах циклов для полного перебора и сравнения информации, которая может находиться в используемых абстрактных таблицах. В качестве примера использования абстрактных таблиц при составлении блок-схем рассмотрим следующую задачу: диаметры ступеней детали, внесенные во вторую графу таблицы исходных данных (ИД) на рис. 7, расположить в первой графе таблицы промежуточных результатов (ПР) в порядке их убывания.

<i>Исходные данные (ИД)</i>			<i>Промежуточные результаты (ПР)</i>			<i>Конечные результаты</i>		
<i>№ поверхности</i>	<i>Диаметр поверхности</i>	<i>Класс точности размеров</i>	<i>Последовательность обработки поверхности</i>	<i>Число проходов</i>	<i>Скорость резания</i>	<i>Шифр наименования перепада</i>	<i>Шифр инструмента</i>	<i>Время резания</i>
1	2	3	1	2	3	1	2	3

Рис.7. Формы абстрактных таблиц

Для решения этой задачи может служить алгоритм, приведенный в виде блок-схемы на рис.8. В схеме использована следующая сим-

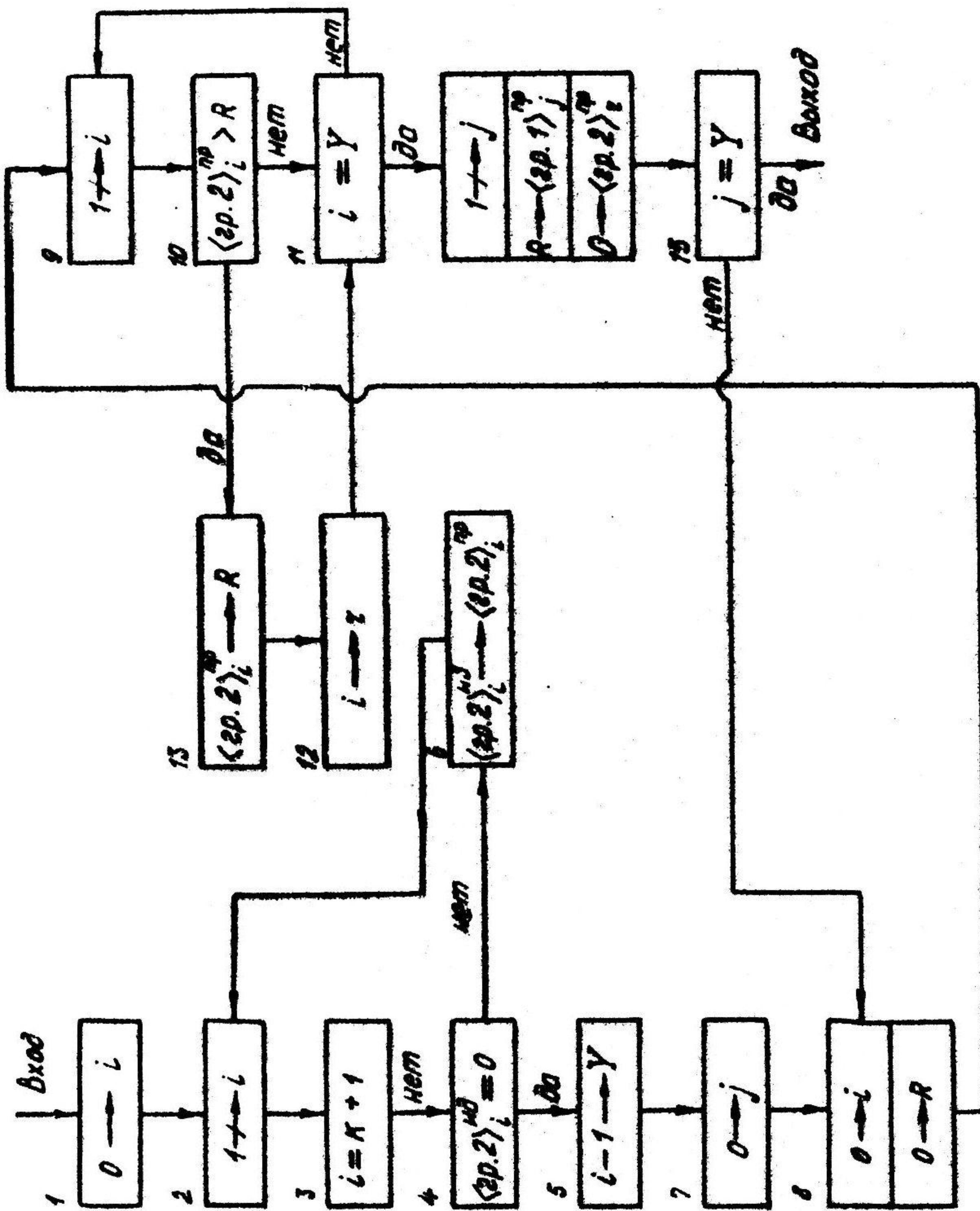


Рис. 8. Блок-схема алгоритма для размещения чисел в порядке их убывания

волика: i - номер строки таблицы; K - общее число строк таблицы ИД (в данном случае $K = 16$); U - количество диаметров детали, внесенных в таблицу ИД (получается в результате расчета); j - номер строки таблицы ПР, используемой для внесения конечных результатов; R - наибольший из рассматриваемых диаметров детали.

Ход расчетов по блок-схеме состоит в следующем: значения диаметров (произвольно расставленных чисел) пересылаются из второй графы таблицы ИД в те же строки второй графы таблицы ПР (см. рис. 7 и 8), одновременно определяется количество внесенных чисел U (блоки 1-6). Из перенесенных в таблицу ПР чисел выбирается наибольшее. Значение этого числа и номер строки, в которой оно находится, засылаются соответственно в ячейки R и U (блоки 9-12). Найденное число R записывается в первую строку первой графы таблицы ПР, а вторая графа той же таблицы очищается - зануляется (блок 14). Во втором цикле расчетов из оставшихся во второй графе таблицы ПР чисел отыскивается следующее наибольшее число, которое засылается во вторую (j) строку второй графы той же таблицы. Общее количество таких циклов будет равным U .

Основные особенности расчетов по рассмотренной блок-схеме состоят в следующем: 1) используется вспомогательная таблица (вторая графа таблицы ПР), в которую сначала засылаются числа из таблицы ИД, а затем они в процессе расчета зануляются; 2) организуются циклы, один из которых находится внутри другого.

7. Составление программы

Составление программы, реализующей технологический алгоритм, является ответственным этапом АПТ. Программирование алгоритмов представляет собой сложный этап работы. Сложность его определяется, с одной стороны, сложностью алгоритма, а с другой - значительной трудоемкостью самого процесса перевода действий, предусмотренных алгоритмом, в команды машины.

Существуют различные способы создания рабочей программы. В основу ее построения могут быть положены: единая жесткая программа, метод подпрограмм, табличный метод и др. Окончательный выбор метода программирования проводится на основании анализа всех условий выполнения работы по АПТ.

При использовании иерархического принципа построения технологического алгоритма, когда весь алгоритм разбивается на отдельные элементарные алгоритмы, описывающие правила решения частных технологических вопросов, в памяти ЭЦВМ необходимо содержать определенный набор программ различного функционального назначения. Такие программы принято называть подпрограммами. Управляющая программа обеспечивает по определенным признакам, которые задаются исходными данными о детали или вырабатываются в процессе работы программы, последовательность вызова подпрограмм и объединение результатов их работы в технологическом процессе обработки конкретной детали (приложение, рис. 2).

При программировании задач АПТ приходится решать очень много вопросов, которые повторяются в частных алгоритмах решения разнообразных технологических задач, например вопросы геометрического анализа чертежа детали, обработка табличной информации, вывод результатов АПТ на печать и т.д. Для решения таких задач составляются стандартные программы, применение которых дает очевидный эффект при программировании различных частных технологических алгоритмов.

Программирование технологической задачи может осуществляться методами ручного и автоматического программирования. Ручное программирование или программирование в кодах ЭЦВМ является ответственной и очень трудоемкой работой. Например, программирование 2000 команд занимает 20-30 дней.

Одним из первых этапов программирования является распределение "памяти" ЭЦВМ, т.е. определение ячеек оперативного ЗУ для записи исходных данных, постоянно используемой числовой информации, промежуточных и конечных результатов расчета, размещение программы. Сущность этой работы можно понять из табл. I7, где показан один из возможных вариантов распределения "памяти" для программирования блок-схемы задачи № I, приведенной на рис. 6.

Из табл. I7 видно, что после распределения "памяти" необходимая для расчетов информация будет храниться в определенных ячейках оперативного ЗУ (номер ячейки получается комбинацией цифр строки и графы):

1) используемые в блок-схеме числа 0, 1, 2, и 7 - соответственно в ячейках 0000, 0001, 0104 и 0103;

2) заданная для выполнения расчетов исходная величина **T** (стойкость инструмента) - в ячейке 0101;

3) промежуточная переменная величина R (номер графы табл.15) - в ячейке 0100;

4) конечный результат расчета K (поправочный коэффициент) - в ячейке 0102;

5) числа, находящиеся в табл.15, - в ячейках 0200-0700 и 0201-0701.

Таблица 17

Распределение памяти ЭЦВМ для расчетов по блок-схеме задачи № I (рис.6 а)

№ ячеек 03У	00	01	02	03	04	05	06	07	10	11	12
00	0	R	T	a	δ	л	u	и	σ	N	15
01	I	T									
02		K									
03		7									
04		2									
05											
06											
07											
10											
11											
...											
27											
...											

Место в "памяти" ЭЦВМ и количество ячеек для команд программы определяются ориентировочно, в данном случае для этого отведены ячейки 0002-0027. Не останавливаясь на специфике решения вопросов, связанных с рациональным распределением "памяти" ЭЦВМ, отметим, что при программировании сложных технологических алгоритмов правильное решение этой задачи имеет очень важное значение.

Резкое сокращение затрат ручного труда и времени программирования дает автоматическое программирование. В этом случае алгоритм записывается на каком-либо символическом языке, близком к обычному. Перевод слов, символов и фраз символического языка в программу осуществляется автоматически при помощи осо-

ых программ, называемых трансляторами.

В настоящее время имеется большое число символических языков, специализированных для решения задач в различных областях науки и техники. Для описания технологических алгоритмов **могут** быть применены автокод "Инженер", язык "Технол" и др.

В качестве примера рассмотрим, как осуществляются составление алгоритмов и автоматическое программирование с помощью автокода "Инженер" (АКИ).

Алгоритм, выраженный в виде блок-схемы или словесно, записывается программистом на входном языке АКИ. Таким образом образуется автокодовая программа (АП). Эта программа заносится далее на перфоленту и вводится в ЭЦВМ. С помощью так называемой программирующей программы, или транслятора, автоматически составляется рабочая программа в машинных командах, по которой в дальнейшем и производят необходимые вычисления. При записи на входном языке АКИ используются цифры и другие символы: русские заглавные буквы - для наименования операторов, латинские - для наименований переменных и функций, а также некоторые специальные обозначения [4]. Запись АП осуществляется с помощью операторов, которые позволяют составить программу любой сложности.

Рассмотрим, как можно записать АП решения задачи № I (рис. 6 а). В том участке алгоритма, где помещена эта задача, технолог представляет только табл. I5 и пишет необходимые положения: по заданному значению стойкости инструмента (в мин.) определить из таблицы и заслать в ячейку Р величину поправочного коэффициента. Вычерчивание блок-схемы решения данной задачи от технолога не требуется. Программист представляет табл. I5 в виде двумерного массива (назовем его **MAS**) со столбцами J (семь столбцов) и строками I (две строки) и далее записывает автокодовую программу:

Метка 2. ЕСЛИ $T > MAS [1, J]$ ТО 3 X

 ВЫЧИСЛИТЬ $P = MAS [2, J]$ X

 ПЕРЕЙТИ 4 X

Метка 3. ПОВТОРИТЬ 2 $J = 1$ (1) 7 X

Метка 4. ...

Эта запись означает, что если переменная Т больше величины элемента **MAS [1, J]**, находящегося в I-й строке и J-м столбце массива **MAS**, то необходимо перей-

ти к участку АП с меткой 3. Эта же метка ставится перед оператором ПОВТОРИТЬ, которому передается управление.

При написании операторов используют символ \sqcup (пробел), каждый оператор заканчивается символом \times .

С помощью оператора ВЫЧИСЛИТЬ осуществляется расчет по формуле.

Важным является оператор ПОВТОРИТЬ, предназначенный для организации циклов в автокодовой программе. Оператор ставится в конце цикла. Метка, стоящая за названием оператора (2), указывает, с какого места автокодовой программы будет начинаться цикл. В операторе указываются параметры (J), подлежащие изменению при каждом прохождении цикла, и правила их изменения (повторить J раз, всякий раз увеличивая индекс J на I, начиная с J = I).

Задача № 2 (рис. 6 б) может быть решена таким же способом: сначала находится нужная строка таблицы, затем необходимый столбец таблицы. Запись решения на входном языке АКИ займет примерно 10 строк. Можно записать решение более компактно, если воспользоваться вычислениями по методу "цикл в цикле". Для этого табл. 16 представляется в виде массива V с 12 столбцами и 10 строками. В первой строке зафиксированы подачи в мм/об, а в первом столбце - диаметры сверла, остальные члены массива - скорости резания.

Пусть D и S - исходные параметры. Задача заключается в том, чтобы найти такой элемент $V[I, J]$ массива, для индексов I и J которого выполнялись бы следующие условия:

$$\begin{aligned} V[I-1, 1] < D \leq V[I, 1] \\ V[I, J-1] < S \leq V[I, J], \end{aligned} \quad (I)$$

и заслать его величину в ячейку V.

Программа поиска нужного элемента выглядит следующим образом:

Номер метки	Автокодовая запись	Номер строки
2	ЕСЛИ $\sqcup D \sqcup V[I, 1] \sqcup TO \sqcup 5 \times$	1
3	ЕСЛИ $\sqcup S \sqcup V[I, J] \sqcup TO \sqcup 4 \times$	2
	ВЫЧИСЛИТЬ $\sqcup V = V[I, J] \times$	3
	ПЕРЕЙТИ $\sqcup 6 \times$	4

Номер метки	Автокододовая запись	Номер строки
4	ПОВТОРИТЬ $\cup 3 \cup \mathcal{J} = 2 \cup (1). 1 2 \mathcal{X}$	5
5	ПОВТОРИТЬ $\cup 2 \cup 1 = 2 \cup (1). 1 0 \mathcal{X}$	6
6	...	

Расшифровка этой записи такова. Первая и шестая строки программы организуют внешний цикл поиска нужной строки таблицы. При исходном диаметре сверла, меньшем, чем очередное значение элемента $V|I,1|$, строка найдена, условие (1) выполнено и происходит переход к оператору с меткой 3. Начинается внутренний цикл поиска нужного столбца найденной строки таблицы. Поиск организован с помощью второй и пятой строк программы. Как только столбец найден, результирующей переменной присваивается значение найденного элемента и происходит обращение к участку программы с меткой 6; вычисления закончены.

Рассмотрим составление программы для решения задачи, представленной на рис.8. Технолог в соответствующем участке алгоритма дает следующие пояснения: записать во вторую графу таблицы ИД (табл.ИД на рис.7), различные диаметры поверхностей детали, расположить в первой графе таблицы ПР в порядке убывания их размеров. Программист выбирает метод решения; если нужно, вычерчивает блок-схему. Для решения задачи используется следующая символика: $I\mathcal{D}|I|$ - элемент таблицы ИД, оформленный в виде одномерного массива; I - номер элемента таблицы ИД; $PR|J|$ - элемент таблицы ПР; J - номер элемента таблицы ПР; R - счетчик строк таблицы ИД; ST - номер строки наибольшего по величине элемента $I\mathcal{D}$.

Программа решения данной задачи выглядит следующим образом:

Номер метки	Автокододовая запись	Номер строки
1	ВЫЧИСЛИТЬ $\cup PR J = 0 \cup R = 0 \mathcal{X}$	1
2	ВЫЧИСЛИТЬ $\cup R = R + 1 \mathcal{X}$	2
	ЕСЛИ $\cup PR J \cup I\mathcal{D} I \cup T0 \cup 3 \mathcal{X}$	3
	ВЫЧИСЛИТЬ $\cup PR J = I\mathcal{D} I \cup ST = R \mathcal{X}$	4
3	ПОВТОРИТЬ $\cup 2 \cup I = 1 \cup (1). 16 \mathcal{X}$	5
4	ВЫЧИСЛИТЬ $\cup I\mathcal{D} I = 0 \mathcal{X}$	6
	ПОВТОРИТЬ $\cup 4 \cup I = ST \cup (1). 1 \mathcal{X}$	7
	ПОВТОРИТЬ $\cup 1 \cup \mathcal{Y} = 1 \cup (1). 16 \mathcal{X}$	

С помощью первой и восьмой строк готовится очередной элемент таблицы ПР. Строки вторая – пятая содержат действия по нахождению наибольшего элемента в таблице ПР и определению номера строки такого элемента. Строки шестая и седьмая необходимы для обнуления члена массива ИД, который переписан в массив ПР. Таким образом, можно компактно записать решение задачи.

Трудоемкость разработки алгоритмов технологического проектирования заставляет искать новые формы представления их, которые позволили бы унифицировать, т.е. привести к единообразию, отдельные участки алгоритма, где решаются формально аналогичные задачи. Такие единообразные участки алгоритма можно одинаково представлять на блок-схемах и обрабатывать по единой программе.

Одним из методов, позволяющих существенно сократить время программирования и отладки логических участков сложных алгоритмов, является метод алгоритмических таблиц (АТ). Эти таблицы используются, когда возникает необходимость реализовать алгоритмы выработки однотипных решений по заданной совокупности исходных параметров. Подобные алгоритмы содержат большое число проверок выполнения различных логических условий.

Алгоритмическая таблица (см. табл. I8) представляет собой матрицу значений параметров $\{a_{ij}\}$ и столбец решений $\{y_i\}$, выбираемых в зависимости от исходных параметров $\{b_j\}$. Элемент матрицы a_{ij} содержит: числовое значение c_{ij} ; знак отношения; адрес перехода.

Таблица I8

Матрица логических решений

b_1	b_2	...	b_j	...	b_n
a_{11}	a_{12}		a_{1j}		a_{1n}
..
a_{i1}	a_{i2}		a_{ij}		a_{in}
..
a_{m1}	a_{m2}		a_{mj}		a_{mn}

y_1
..
y_i
..
y_m

В качестве знака отношений применяют символы: (- "меньше";) - "больше"; (= - "меньше или равно";)= - "больше или равно"; = - "равно"; =/ - "не равно".

Выборка решения из АТ происходит следующим образом. Если V_j удовлетворяет отношению $V_j * C_{ij}$; где * - какой-либо знак отношения, то происходит переход к элементу данной строки АТ, находящемуся в следующем столбце. Далее берется следующее значение из строки текущих параметров V_{j+1} и проверяется отношение $V_{j+1} * C_{i,j+1}$. Если заданное отношение не выполняется, то происходит переход по адресу.

Адрес перехода может быть трех типов:

1) "стандартный адрес", определяющий переход к первому элементу следующей по порядку строки АТ данного столбца; этот адрес специально не указывается;

2) метка автокодированной программы, к которой осуществляется переход из данной АТ;

3) номер строки из столбца данной АТ.

Процесс просмотра элементов O_{ij} заканчивается, если следующим элементом строки оказывается решение U_i , т.е. некоторому параметру присваивается числовое значение, или если происходит выход из АТ по метке к какому-либо участку АП.

Язык записи АТ подразделяется на два: внешний язык АТ, на котором производится запись алгоритмов, и внутренний язык АТ, на котором АТ обрабатываются в процессе счета. Во внешнем языке используются символы АКИ, при этом русский алфавит применяется для названия АТ и пояснения к ней. Когда элемент $O_{i+1,j}$ совпадает с элементом O_{ij} ($i \neq 1$), то вместо $O_{i+1,j}$ может быть проставлен в соответствующей клетке АТ знак ".". Если знаки отношения в элементах $O_{i+1,j}$ и O_{ij} совпадают, то в элементе $O_{i+1,j}$ знак отношения может опускаться. Знак " " (пробел) обозначает безусловный переход к следующему столбцу данной строки.

После того как алгоритмическая таблица записана на внешнем языке, осуществляется перфорация ее данных на перфоленте, ввод перфоленты в ЭЦВМ, перевод АТ на внутренний язык и запись на магнитную ленту. Программа обработки алгоритмических таблиц оформляется в виде библиотечной программы АКИ, которой присваивается определенный номер, например, 1973. В том месте АП, где должен находиться алгоритм, записанный с помощью АТ, происходит

обращение к библиотечной программе 1973:

БИБЛ ПРОГР 1973 (АМЗ, В|1|У) 8 , (2)

где АМЗ - наименование переменной, которой до обращения к библиотечной программе должно быть присвоено значение адреса на магнитной ленте первого элемента АТ;

В|1| - начальный адрес первого элемента массива В (массив исходных параметров);

У - наименование переменной, которой присваивается числовое значение (результат).

Таким образом, алгоритмы, записанные с помощью АТ, не фигурируют в явном виде в автокодированной программе. Все алгоритмические таблицы рассматриваются как исходные данные, которые в любой момент достаточно просто могут быть изменены.

Рассмотрим для примера решение задачи выбора типа токарно-револьверного автомата по конструктивным характеристикам детали. Предельные параметры обрабатываемых деталей для каждой модели автомата зафиксированы в табл.19. При использовании алгоритмических таблиц искомый алгоритм может быть представлен в виде табл.20. Информация, обведенная жирной рамкой, вводится в машину. Рассмотрим, как происходит выбор модели автомата с помощью АТ.

Таблица 19

Таблица условий выбора модели автомата

Модели автоматов, используемых при создании алгоритмов	1112	1118	1124	1136	Индекс 52
Шифр модели автомата (Q)	2	3	4	5	6
Диаметр заготовки максимальный (AZ), если заготовка круглая, т.е. при Z < 4	12	18	24	36	52
Размер заготовки под ключ (AZ), если заготовка шестигранная, т.е. при Z = 6	9	14	20	30	44
Размер заготовки под ключ (AZ), если заготовка четырехгранная, т.е. при Z = 4	7	10	17	29	36
Максимальный диаметр наружной резьбы (AR) для стали, т.е. при m = 1	8	10	18	22	18

Максимальный диаметр наружной резьбы (DR) для остальных материалов, т.е. при $M > I$	10	12	22	27	22
Длина детали (LZ)	60	60	90	90	150

Таблица 20

Алгоритмическая таблица выбора автомата

ω	Z	DZ	M	DR	LZ	Q
	1	2	3	4	5	6
ТАБЛ. 15, 6. ВЫБОР МОДЕЛИ АВТОМАТА						
1	(4/6, 1/	(=12.	=1/6, 4/.	(=8.	(=60/3, 6/.	2 X
2	U	18.	1/7, 4/.	10.	.	3 X
3	U	24.	1/8, 4/.	18.	90/5, 6/.	4 X
4	U	36.	1/9, 4/.	22/10/.	.	5 X
5	U	52/10/.	1/10, 4/.	18/10/.	150/1, 0/.	6 X
6	=6/11, 1/.	9.	1/1, 4/.	10.	60/8, 6/.	2 X
7	U	14.	1/2, 4/.	12.	.	3 X
8	U	20.	1/3, 4/.	22.	90/10, 6/.	4 X
9	U	30.	1/4, 4/.	27/10/.	.	5 X
10	U	44/10/.	1/5, 4/.	22/10/.	150/10/.	6 X
11	=4.	7.	=1/6, 4/.	8.	60/13, 6/.	2 X
12	U	10.	1/7, 4/.	10.	.	3 X
13	U	17.	1/8, 4/.	18.	90/15, 6/.	4 X
14	U	29.	1/9, 4/.	22/10/.	.	5 X
15	U	36/10/.	1/10, 4/.	18/10/.	150/10/.	6 X

Пусть взята деталь со следующими значениями параметров: $Z = 6$; $DZ = 14$; $M = 1$; $DR = 10$; $LZ = 40$. Таким образом, массив исходных данных (обозначим его ω) имеет вид: $\omega = (6, 14, 1, 10, 40)$.

В результате вычислений по табл. 20 переменной Q будет присвоено значение 3: этот шифр обозначает, что деталь можно обработать на станке модели IIIB. Правильность такого решения легко проверить по табл. 19. Поиск решений шел по следующим клет-

кам табл.20 : A_{11} , A_{61} , A_{62} , A_{72} , A_{73} , A_{24} , A_{25} , A_{26} .

При этом стандартный адрес использовался при переходе от элемента A_{11} к A_{61} , от A_{62} к A_{72} ; адрес с использованием номера строки и столбца применялся при переходе к A_{24} .

8. Реализация задачи АПТ

Основные операции по проектированию с помощью ЭВМ технологического процесса изготовления деталей приведены на рис.9.

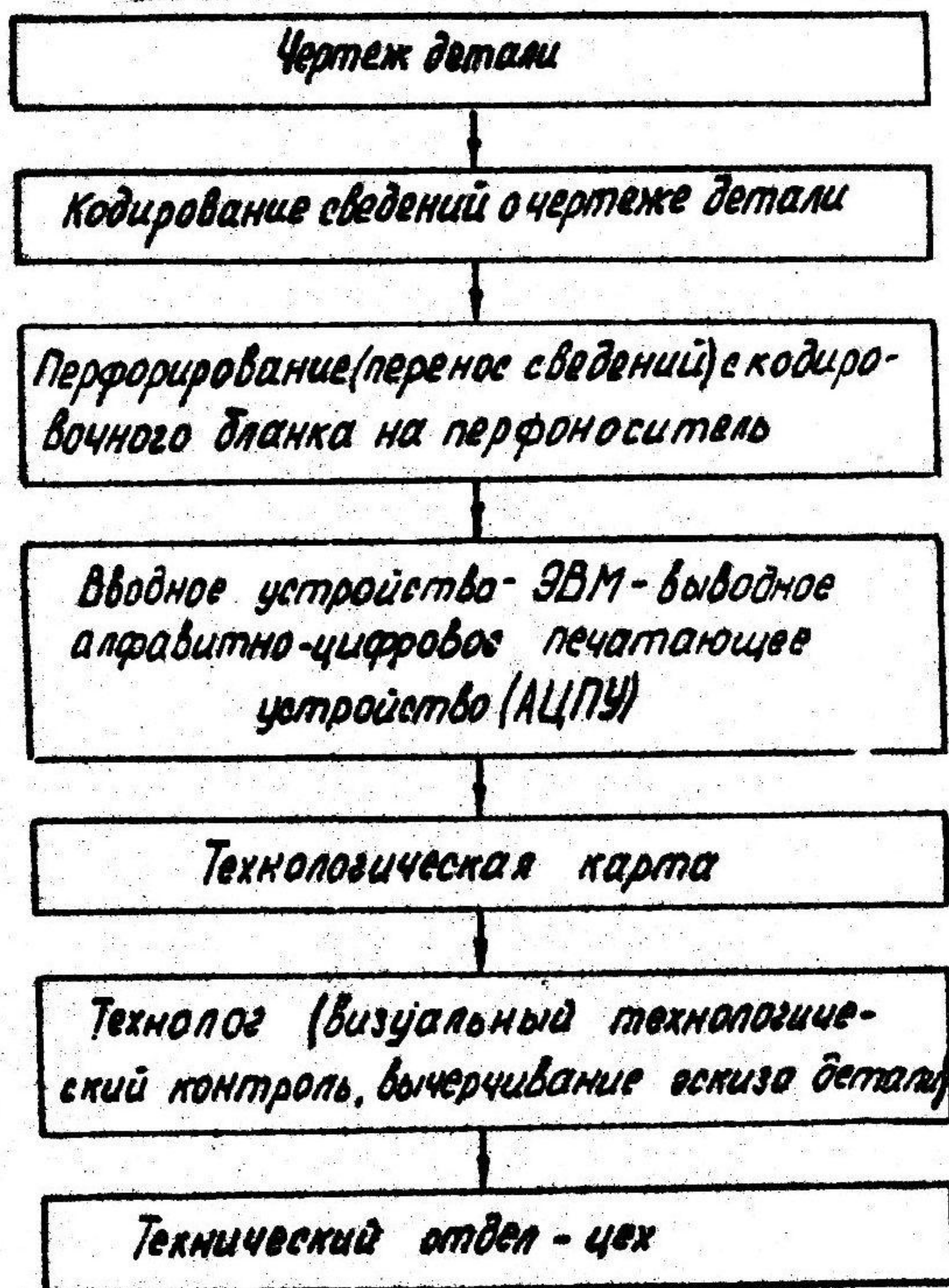


Рис.9. Схема автоматизированного проектирования технологических процессов с помощью ЭВМ

Для кодирования необходимо лишь уметь читать чертеж, но не проектировать технологический процесс обработки деталей. Поэтому кодирование может проводить техник средней квалификации.

Перфорация данных с кодировочной таблицы осуществляется по единым макетам для всех классов деталей: один макет - для заглавной строки кодировочной таблицы - содержит алфавитно-циф-

ровую информацию о детали (номер детали, наименование детали, марку материала и пр.), другой макет — для остальных строк — содержит только цифровую информацию, отображающую геометрию детали и технические требования, указанные на чертеже.

Каждая строка кодировочной таблицы перфорируется на отдельной перфокарте.

Машина формирует и нормирует технологический маршрут обработки детали, после чего компоует и печатает его в форме технологических карт (рис.10). Следует заметить, что методы разработки алгоритмов и программ определения оптимальных режимов резания и технически обоснованных расчетных норм времени приведены в [2]. Перед выдачей в цех технологические карты, составленные на ЭВМ, просматриваются и подписываются работниками отдела главного технолога.

Однотипность и четкость формулировки операций и переходов облегчают рабочему ознакомление с технологическим процессом.

Автоматизация технологического проектирования сокращает сроки выполнения и стоимость технологической подготовки производства.

Получение с ЭВМ технически и экономически обоснованных технологических решений, материальных и трудовых нормативов является основой оптимального планирования и будет способствовать повышению экономических показателей эффективности производства.

Заключение

В заключение можно отметить следующее. Предлагаемые методические положения по разработке алгоритмов проверены при решении нескольких задач по проектированию экономически целесообразных технологических процессов изготовления различных классов деталей. Несмотря на некоторую неполноту, они дают возможность сформулировать определенные понятия и систематизировать опыт в новом перспективном направлении применения творческого труда инженеров-технологов.

Литература

1. Алексеев В.Г., Малоз А.Н. Методическое пособие к курсовому проектированию по технологии приборостроения, вып.2, М., изд. МВТУ, 1973.
2. Горанский Г.К. и др. Автоматизация технического нормирования работ на металлорежущих станках с помощью ЭВМ, М., "Машиностроение", 1970.
3. Митрофанов С.П. Научная организация серийного производства, Л., "Машиностроение", 1970.
4. Неменман М.Е. Программирование на АКИ, Минск, "Вышэйшая школа", 1972.
5. Цветков В.Д. Система автоматизации проектирования технологических процессов, М., "Машиностроение", 1972.

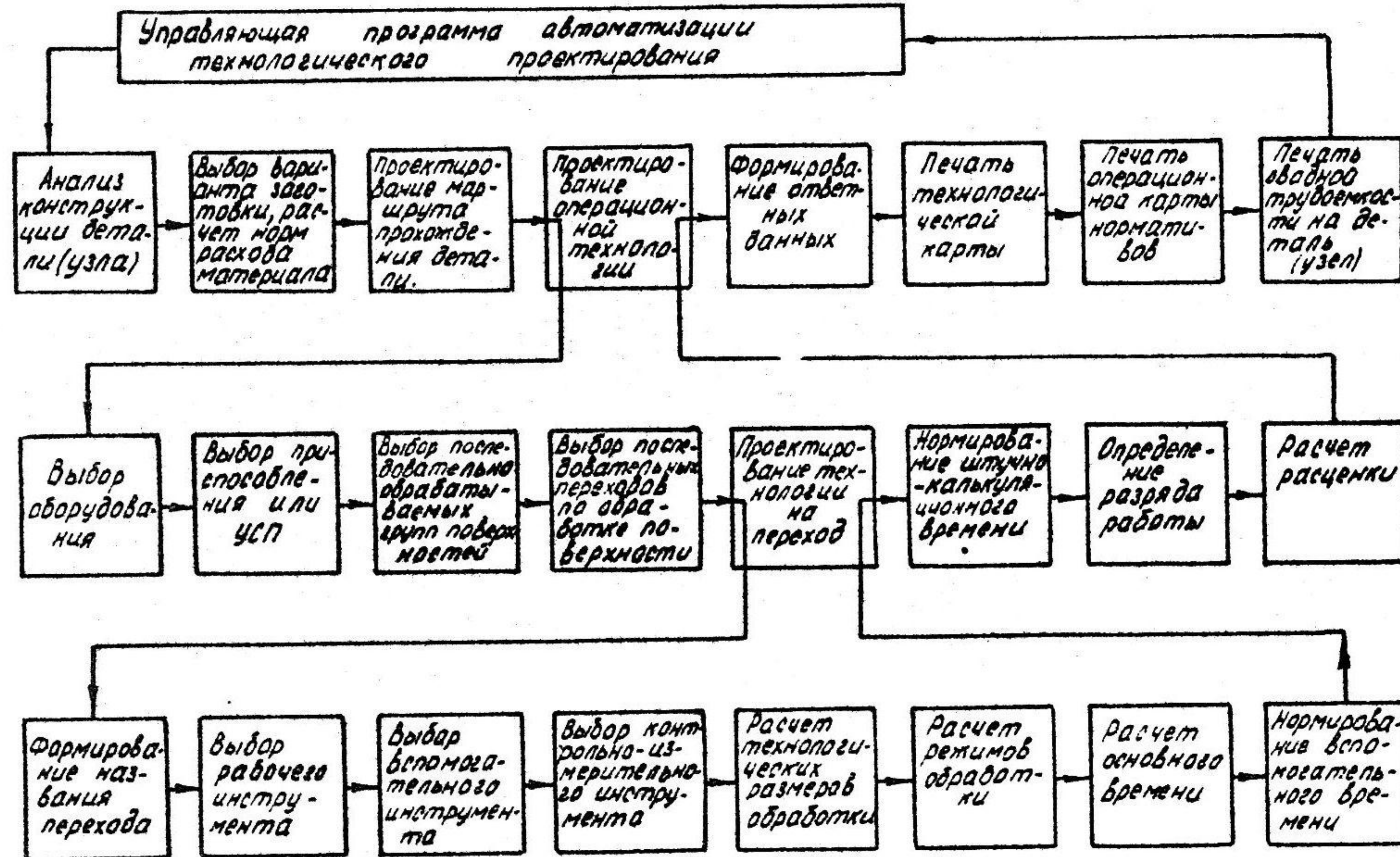


Рис.1. Структурная схема программы автоматизации технологического проектирования

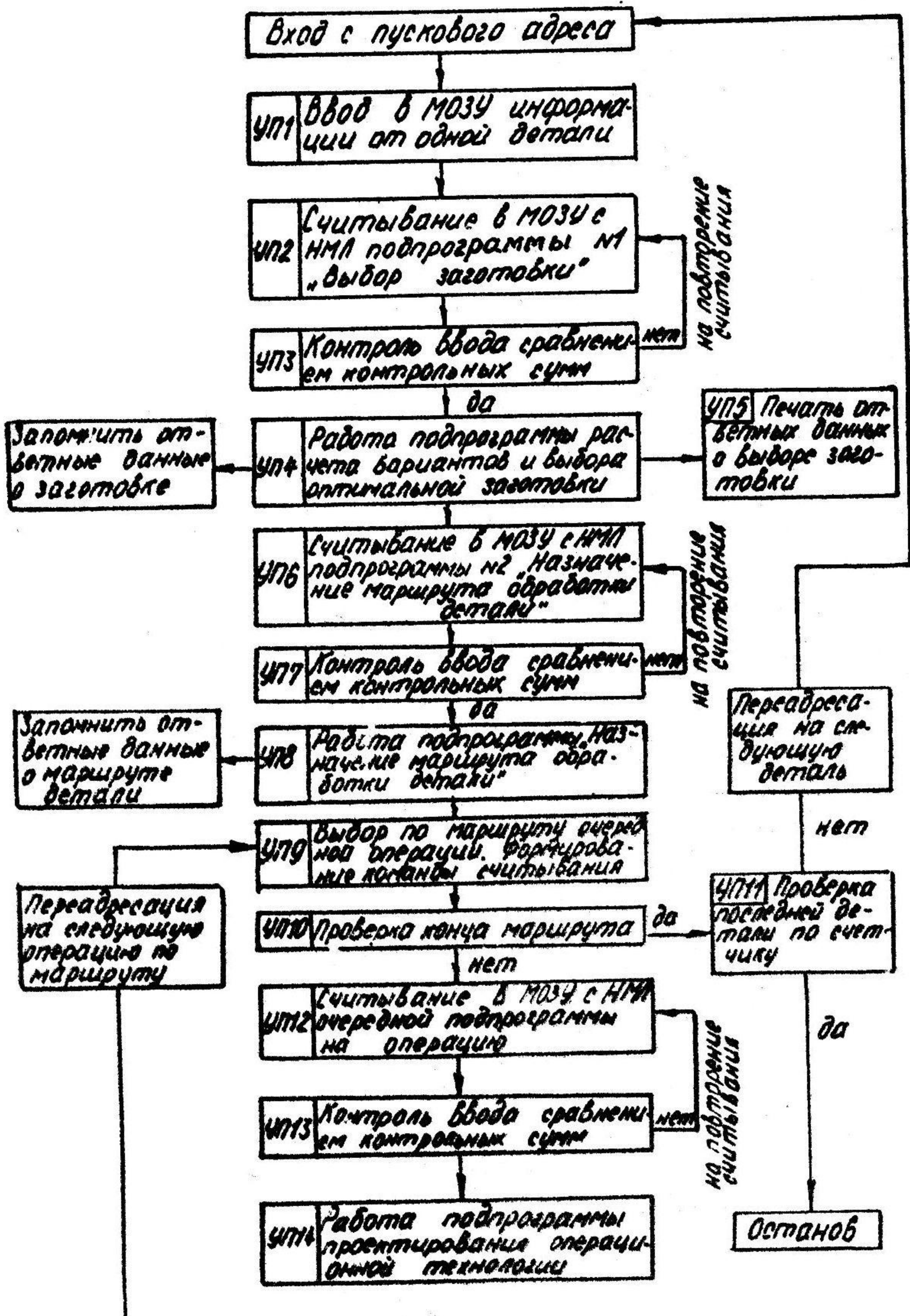


Рис.2. Блок-схема управляющей программы

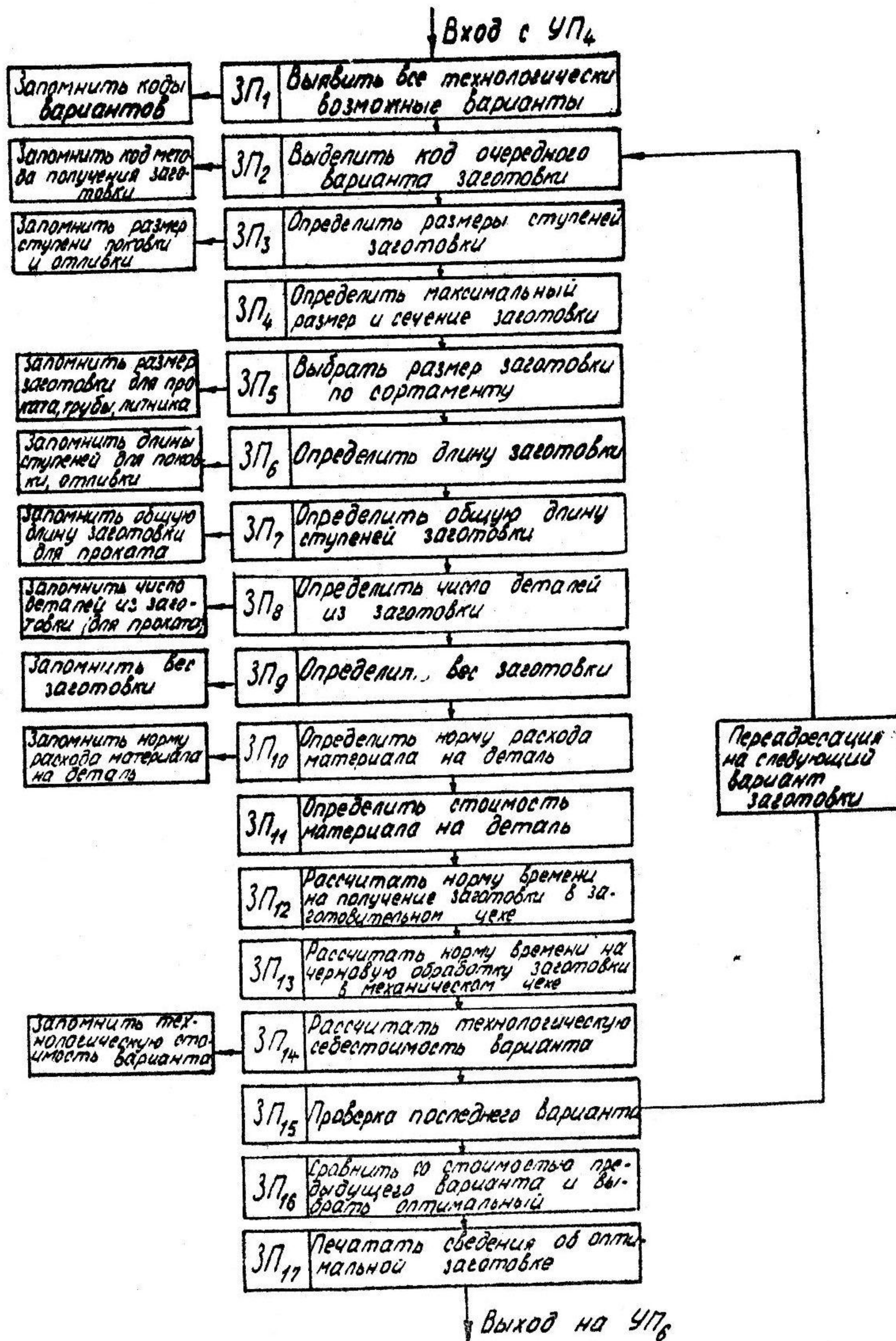


Рис. 3. Блок-схема подпрограммы расчета вариантов и выбора оптимальной заготовки

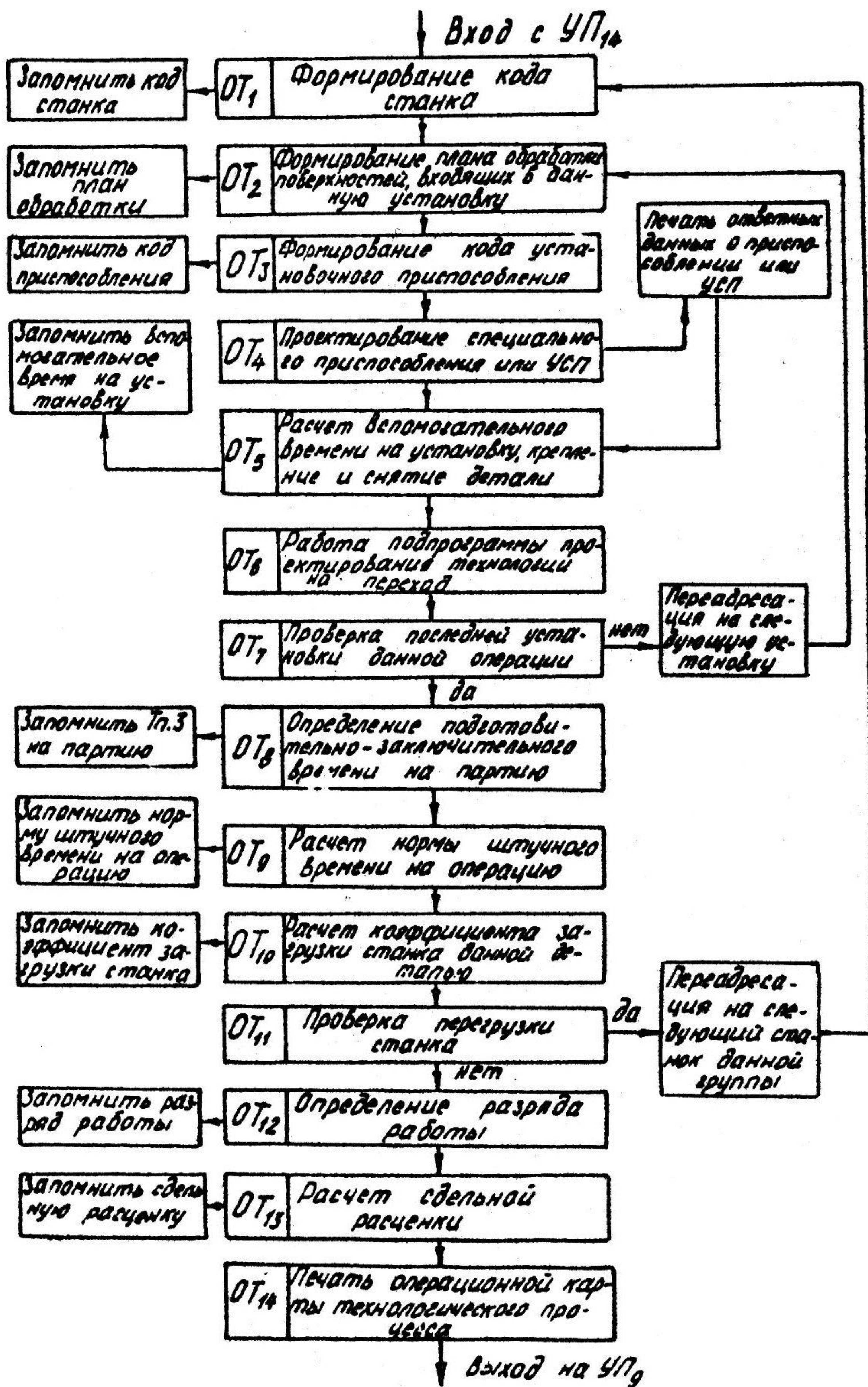


Рис. 4. Блок-схема подпрограммы проектирования операционной технологии

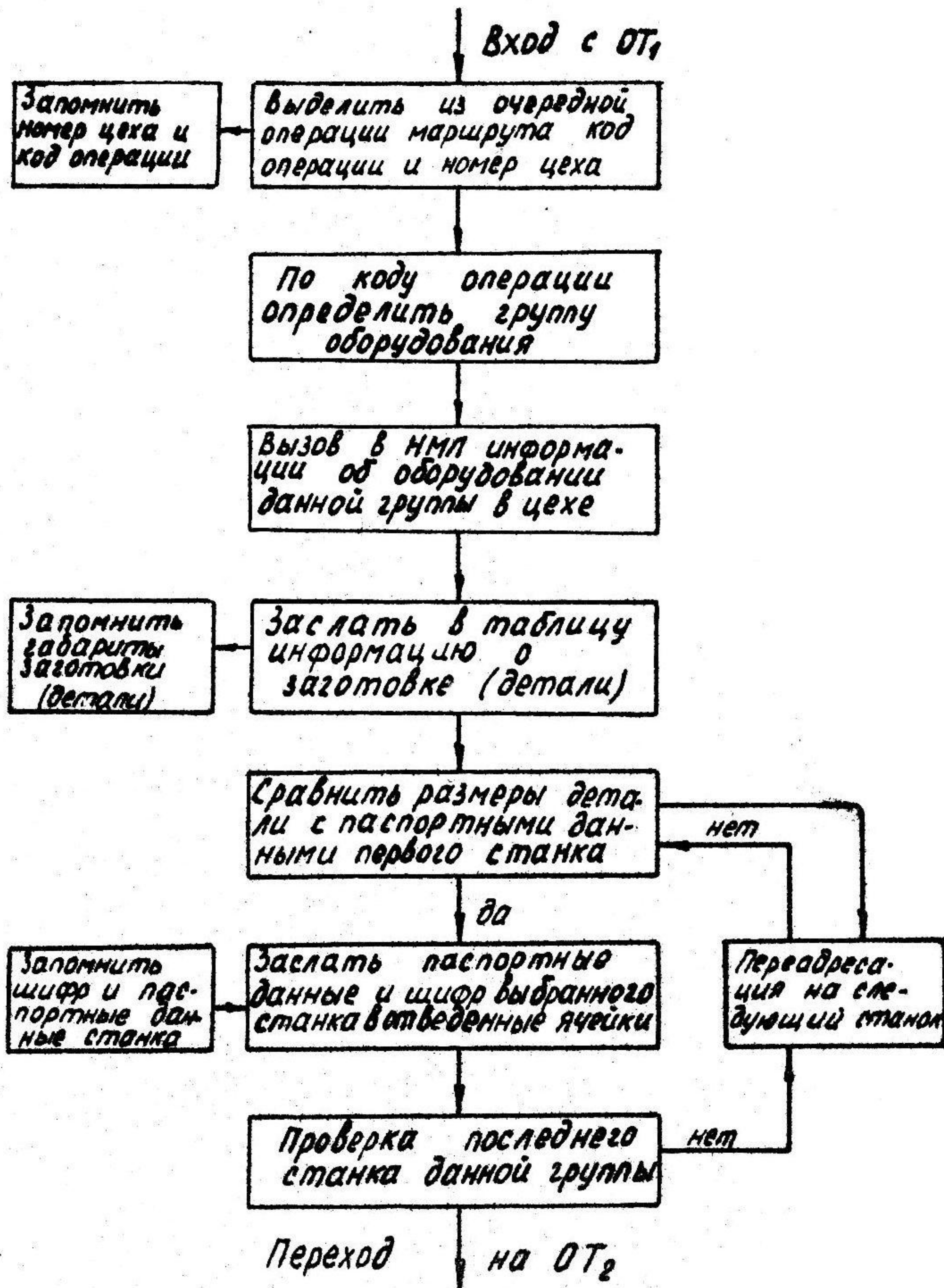


Рис.5. Блок-схема алгоритма выбора оборудования

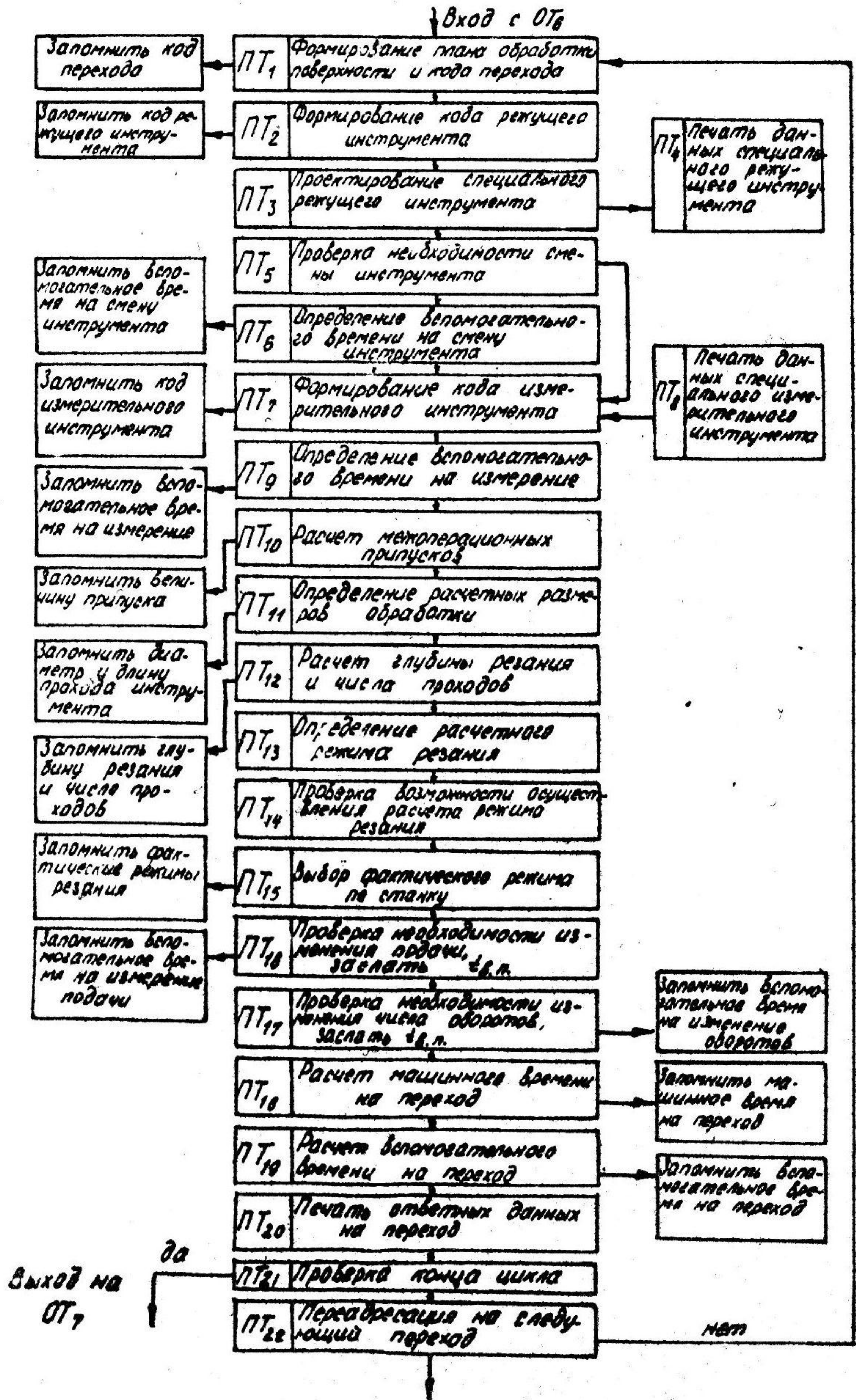
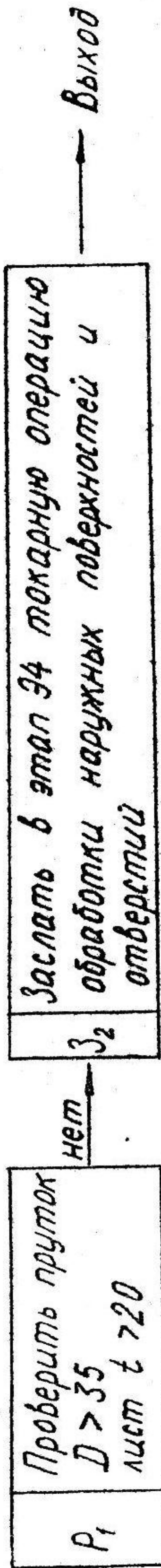


Рис. 6. Блок-схема подпрограммы проектирования технологии на переход



Виды токарных операций в этапе Э2							
Условия, определяющие выбор черновых операций	Точение шеек под люнет. Расточка центровых отверстий. Точение наружных поверхностей. Расточка отверстий	Точение наружных поверхностей. Расточка отверстий	Точение шеек под люнет. Расточка отверстий	Точение наружных поверхностей	Расточка отверстий	Точение наружных поверхностей и отверстий	
Вид заготовки	поковка, пруток					пруток	—
Класс детали	вал					втулки, диски	—
Необходимость черновой обработки наружных поверхностей	да	да	нет	да	нет	да	
Необходимость черновой обработки отверстий	да	да	да	да	да	да	
Длина детали ≥ 600	да	нет	нет	нет	—	—	
Диаметр центрального сквозного отверстия ≥ 20	—	—	—	—	—	—	

Рис.7. Алгоритм формирования черновых токарных операций

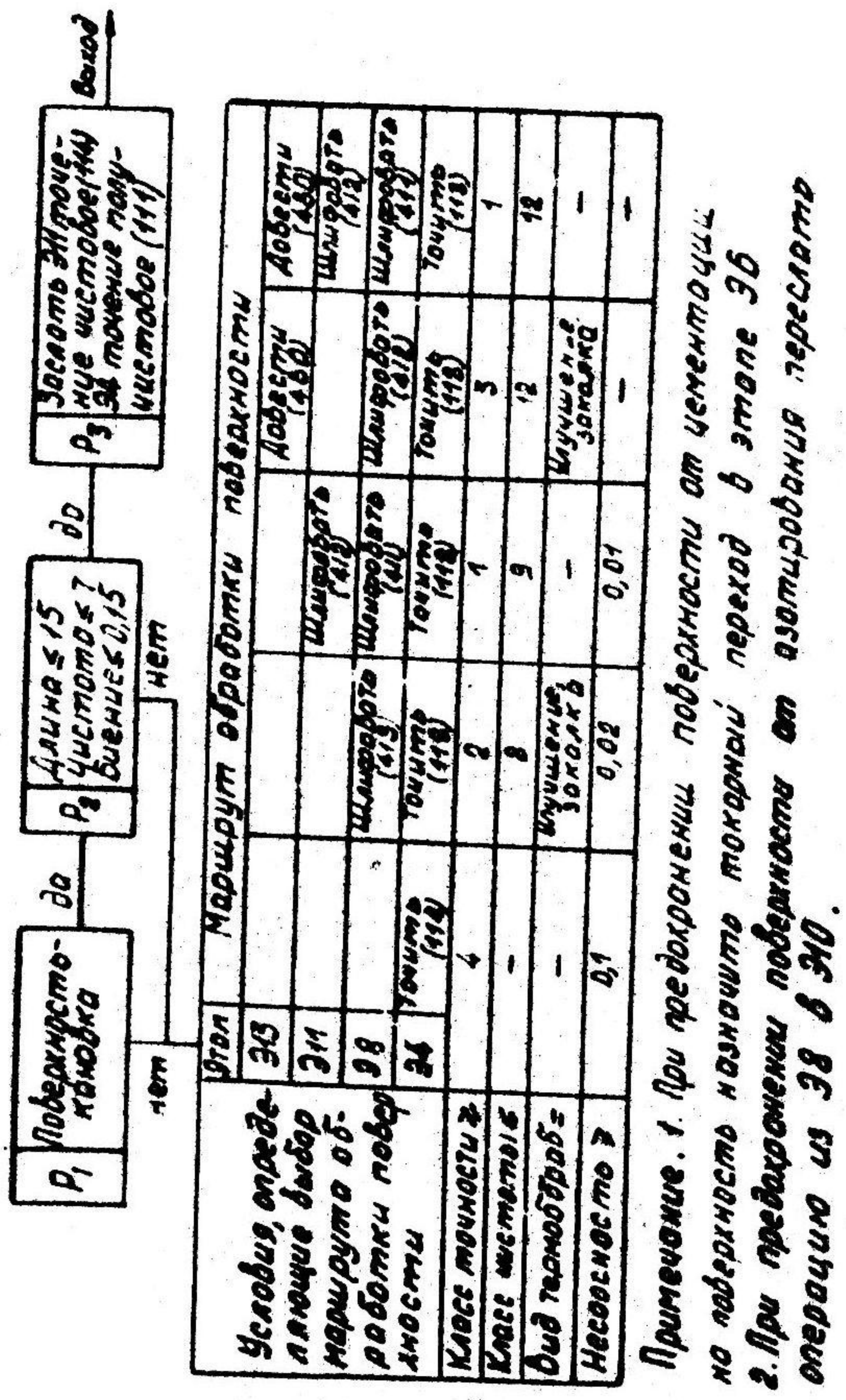


Рис.8. Алгоритм выбора маршрута обработки поверхностей вращения

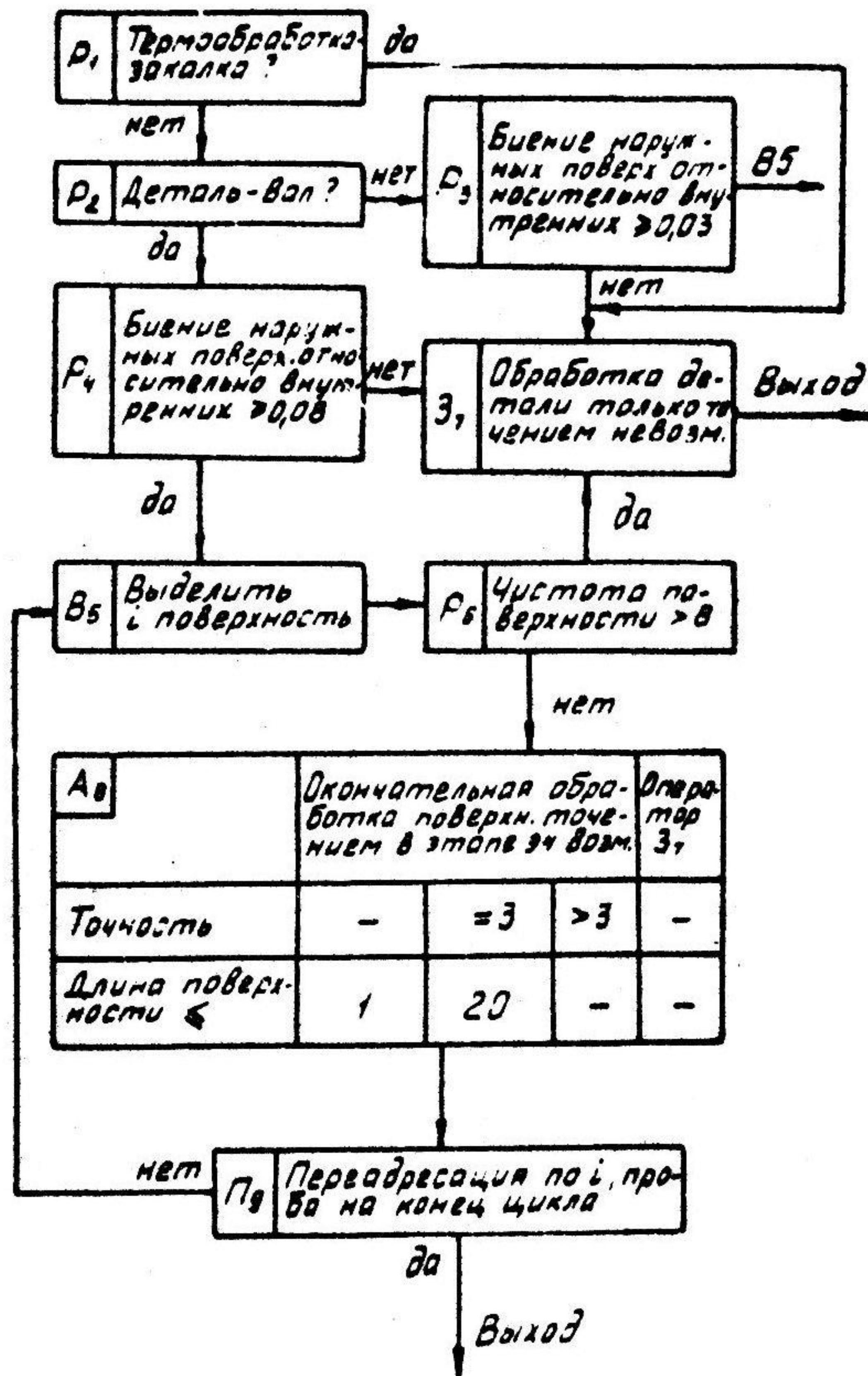


Рис.9. Алгоритм назначения операции точения в этапе Э4

1. Алгоритм формирования операции.
2. Выбор схемы установки и базирования.
3. Выбор i -станка и зажимного приспособления
4. Определение оптимальной последовательности установок и переходов в операции.
5. Выделить j -переход:
 - 5а. Определение оптимальных припусков и межоперационных размеров перехода.
 - 5б. Выбор режущего и вспомогательного инструмента.
 - 5в. Выбор мерительного инструмента.
 - 5г. Выбор метода обработки элементарных обрабатываемых поверхностей.
 - 5д. Определение оптимальных режимов резания и машинного времени.
 - 5е. Расчет вспомогательного времени на переход.

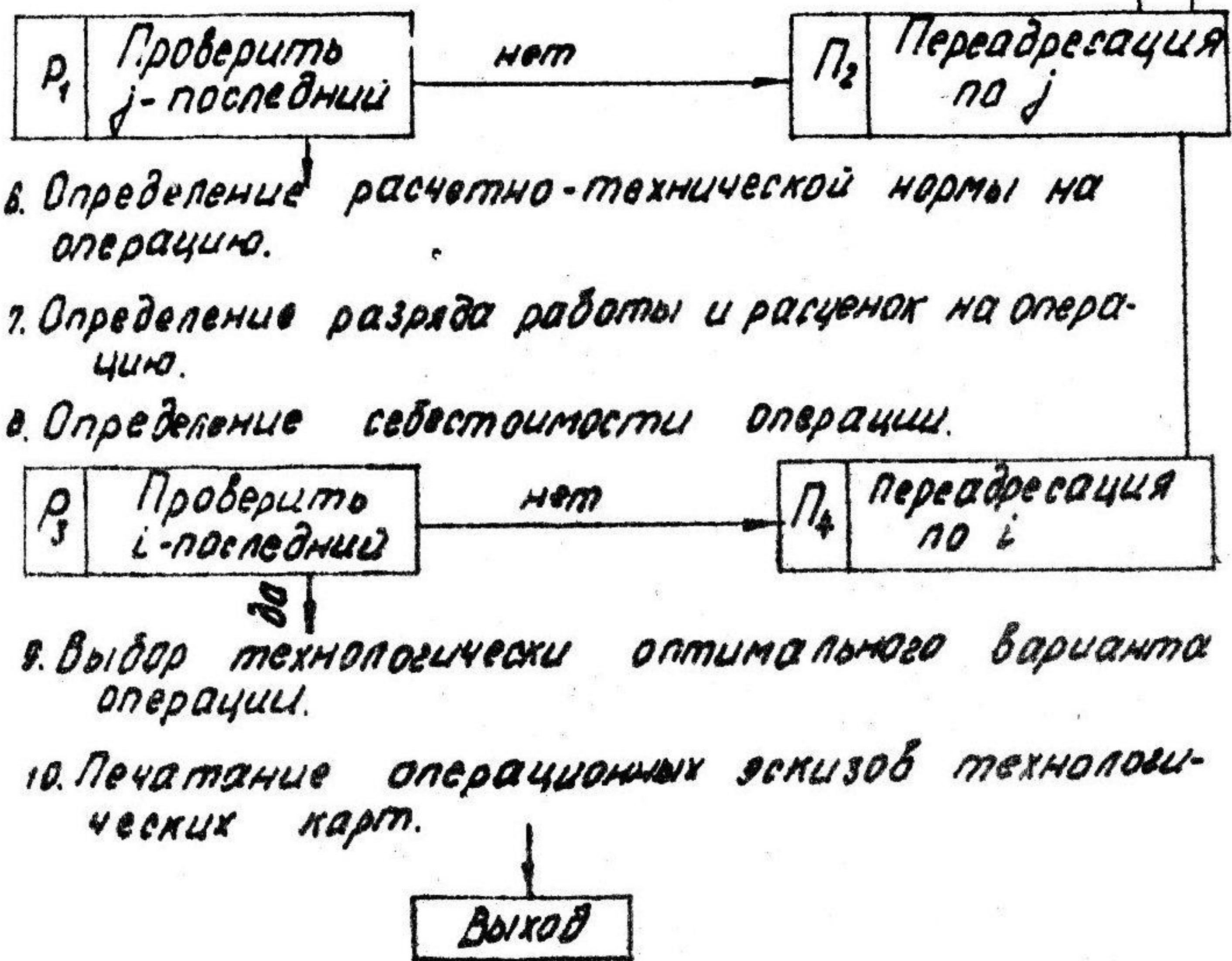


Рис. 10. Универсальная блок-схема алгоритма проектирования оптимальных технологических процессов обработки деталей

Условия, определяющие выбор типа токарной операции				Тип токарной операции			
				автомат- ная	револьвер- ная	универ- сальная	
Максимальный диа- метр заготовки \leq	Вид заготовки =	индивиду- альная	класс детали	Втулки	300	250	300
				диски	30	500	550
	группо- вая			Втулки	36	80	180
				диски	36	80	250
Минимальный диаметр заготовки \geq				4	5	8	
Длина детали \leq				140	250	2300	
Класс точности \geq				3а	3	2	
Класс чистоты \leq				6	6	7	
Несоосность \geq				0,10	0,05	0,03	
Размер партии \geq				150	50	-	
Число обрабатываемых по- верхностей в операции \geq				4	3	-	

Рис. II. Алгоритм определения типа токарной операции

	1. Черновая обработка наружных поверхностей 2. Черновая обработка отверстий	1. Чистовая обработка отверстий 2. Чистовая обработка наружных поверхностей	1. Чистовая обработка наружных поверхностей 2. Чистовая обработка отверстий	1. Черновая обработка наружных поверхностей 2. Черновая обработка отверстий 3. Чистовая обработка отверстий 4. Чистовая обработка наружных поверхностей	3. Чистовая обработка наружных поверхностей 4. Чистовая обработка отверстий
Вид операции	черновая	чистовая		совмещенная	
Вид обрабатываемых поверхностей	наружные и внутренние				
Необходимость обработки точных отверстий в операции	—	да	нет	да	нет

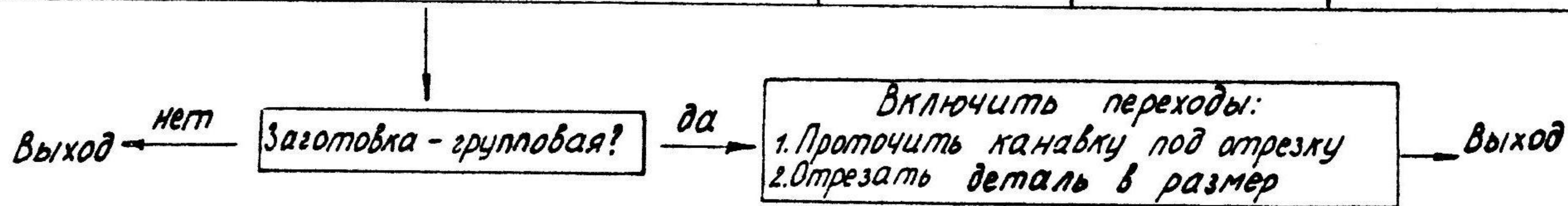


Рис.12. Алгоритм определения последовательности обработки групп поверхностей в токарно-револьверной операции

Условия, определяющие выбор модели револьверного станка	Модель станка															
	с горизонтальной осью								с вертикальной осью							
	1322		1Г325		1336Р		1А341		1Н318		1338		1М38		1П365	
Вид заготовки =	А	Б	А	Б	А	Б	А	Б	А	Б	А	Б	А	Б	А	Б
Наибольший диаметр заготовки ≤	22	80	25	80	38	100	40	140	18	130	38	220	60	350	80	450
Наибольшая длина обрабатываемой детали ≤	80	—	100	—	130	—	140	—	90	—	150	—	740	—	550	—
Количество обрабатываемых поверхностей	4	3	4	3	4	3	4	3	—	—	—	—	—	—	—	—
Наличие конических поверхностей	нет		—		нет		—		нет		нет		нет		нет	

Примечание. А - заготовка из прутка; Б - заготовка штучная.

Рис. 13. Алгоритм выбора типоразмера револьверного станка.

ОГЛАВЛЕНИЕ

Введение	3
I. Задачи при проектировании технологических процессов и математические методы их решения	5
2. Основные этапы автоматизированного проектирования технологических процессов.	6
3. Выбор объекта автоматизации.	6
4. Разработка методики технологического проектирования.	7
5. Разработка системы кодирования исходной информации	10
6. Алгоритмизация методики технологического проектирова- ния.	19
7. Составление программы.	33
8. Реализация задачи АПТ.	43
Заключение	45
Литература	46
Приложение	47