

Министерство высшего и среднего специального образования СССР

Московское ордена Ленина, ордена Октябрьской Революции
и ордена Трудового Красного Знамени
высшее техническое училище им. Н. Э. Баумана

В. Г. АЛЕКСЕЕВ, К. Б. ЛУКИН, Э. С. НАПАЛКОВ

**АЛГОРИТМИЗАЦИЯ ПРОЕКТИРОВАНИЯ
ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ ПРОИЗВОДСТВА
ЭВА И РЭА**

Методические указания
для курсового и дипломного проектирования
по курсу
«Автоматизация проектирования конструкций и
технологических процессов ЭВА и РЭА»

Данные методические указания издаются в соответствии с учебным планом. Рассмотрены и одобрены кафедрой П-8 03.12.84 г., методической комиссией факультета П 19.12.84 г. и учебно-методическим управлением 13.02.85 г.

Рецензент к.т.н. доц. А.В.Фролов

© Московское высшее техническое училище им. Н.Э.Баумана

Оглавление

Введение	3
Глава 1. Алгоритмизация составления тестов для автоматизированного контроля микросхем и ТЭЗов с полупроводниковыми элементами памяти	3
§ 1.1. Составление контрольных таблиц соответствия для триггеров	3
§ 1.2. Правила составления контрольных тестов для цифровых микросхем и типовых элементов замены с триггерами	9
§ 1.3. Пример составления контрольного теста микросхемы с RS-асинхронными триггерами	12
Литература	16
Глава 2. Технологические основы автоматизации проектирования технологических процессов	16
§ 2.1. Методические основы разработки правил технологического проектирования	16
§ 2.2. Классификация элементарных планов и условия их применимости	20
§ 2.3. Разработка граф-схем алгоритмов выбора элементарных планов обработки поверхностей	24
Глава 3. Оптимизация структуры технологического процесса	27
§ 3.1. Определение величин уступок	27
§ 3.2. Выбор метода решения задачи по критерию минимума числа операций	29
§ 3.3. Последовательный алгоритм определения оптимальных покрытий	30

Редактор Г.Ф.Хлебинская

Корректор Л.И. Малютина

Заказ 854. Объем 2,25 п.л. (2,25 уч.-изд.л.) Тираж 500 экз.
Бесплатно. Подписано к печати 11.05.85 г. План 1985 г., № 40.

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

ВВЕДЕНИЕ

Выполнение задачи, поставленной XXVI съездом КПСС по внедрению в промышленность "миниатюрных электронных управляющих машин", требует дальнейшего повышения качества и степени интеграции элементной базы этих машин — микросхем. При сборке электронной аппаратуры наибольшая доля ее брака определяется использованием негодных микросхем и дискретных радиоэлементов. Поэтому разработка автоматизированных средств контроля статических и динамических параметров микросхем является актуальной задачей.

Изложение алгоритма разработки тестов для статического контроля микросхем и ТЭЗов с комбинационными элементами приведено в [1]. В первой главе настоящей работы описан алгоритм разработки контрольных тестов для микросхем и ТЭЗов с полупроводниковыми элементами памяти — триггерами. Разработаны контрольные таблицы соответствия для наиболее распространенных в электронной аппаратуре триггеров, приведен пример контрольного теста для типовой микросхемы.

Во второй главе рассмотрены технологические основы автоматизации проектирования технологических процессов. Разработана методика технологического проектирования (правила экстраполяции) для любого технологического процесса (на примере механической обработки детали). По этим правилам находят наилучший путь для достижения конечной цели на определенном этапе процесса проектирования. Рассмотрен вариант правил экстраполяции в виде обратных полиномов.

В третьей главе рассмотрены критерии оптимизации технологического проектирования и методика решения задач оптимизации технологического проектирования.

Глава I. АЛГОРИТМИЗАЦИЯ СОСТАВЛЕНИЯ ТЕСТОВ ДЛЯ АВТОМАТИЗИРОВАННОГО КОНТРОЛЯ МИКРОСХЕМ И ТЭЗОВ С ПОЛУПРОВОДНИКОВЫМИ ЭЛЕМЕНТАМИ ПАМЯТИ

§ 1.1. Составление контрольных таблиц соответствия для триггеров

Для каждого триггера схемы составляют контрольную таблицу соответствия. Последняя позволяет определить состояние триггера как функцию от входного набора и предыдущего состояния, а также проверяемые неисправности входов триггера для каждого входного набора. Контрольную таблицу соответствия составляют следующим

образом: для каждого входного набора x , исключая запретные, и каждого предыдущего состояния Q^n триггера определяют новое состояние Q^{n+1} , в результате чего получают таблицу соответствия, которая реализует логическое уравнение

$$Q^{n+1} = \lambda(Q^n, x).$$

Схемы и логические уравнения наиболее распространенных триггеров приведены в табл. 1.1. Ниже логического уравнения в той же клетке указаны условия, при которых выполняется данная функция. Например, логическое уравнение для RST -триггера по асинхронному инверсному входу выполняется, если хотя бы на одном входе (\bar{R} или \bar{S}) существует логический сигнал "1". Одновременно с этим условием должен отсутствовать (т.е. равен 0) сигнал на тактовом входе C RST -триггера. В левой колонке указаны обозначения ИС (интегральная схема), в которых применяется данный тип триггера [2]. Контрольную таблицу соответствия получают из таблицы соответствия введением графы "неисправность входов". Эту графу заполняют для каждого входного набора следующим образом (табл. 1.2).

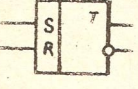
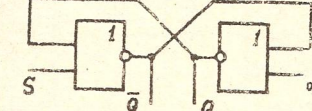
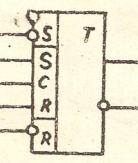
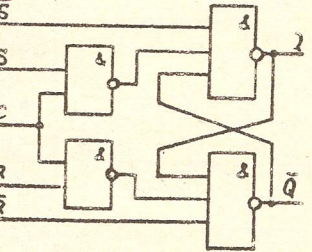
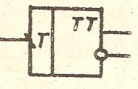
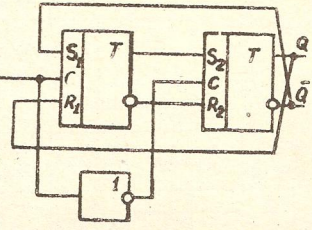
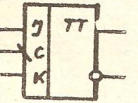
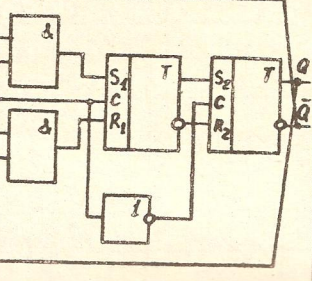
Предполагают неисправность какого-либо входа, вследствие чего на выходе Q^{n+1} получатся противоположные значения. Анализируют каждый вход триггера и устанавливают, на каком из входов изменение сигнала на противоположный приводит к переключению триггера. В графе "Неисправность входов" записывают обозначение этого входа и значение сигнала на нем, который приводит к переключению триггера.

Например, при подаче первого входного набора RS -асинхронный триггер (табл. 1.2) должен сохранять свое исходное состояние ($Q^n = Q^{n+1}$). Если при подаче первого набора триггер из состояния $Q^n = 0$ переключается в состояние $Q^{n+1} = 1$, то на входе S вследствие неисправности вместо "0" подается "1". Поэтому в графе "Неисправность входов" записывают " $S = 1$ ", что свидетельствует о наличии на входе S неисправности — постоянного сигнала "1".

Изменение значения Q^{n+1} на противоположное может быть при наличии независимых друг от друга двух или трех неисправностей входа. Например, на седьмом наборе RST -синхронного триггера любая из трех неисправностей входа ($S = 0$ или $C = 0$ или $R = 1$) приводит к отличию значения Q^{n+1} от истинного значения на данном наборе.

Классификация триггеров

Таблица I.I

Тип. Обозначение ИС	Условное обозначение	Логическое уравнение. Условия	Функциональная схема
1	2	3	4
RS К155ХМ1 К561ТР2 КН02АМ1		$Q^{n+1} = S + \bar{R}Q^n$ $RS = 0$	
RST К511ТР2 К571ИЕ1 К502ИМ1		синхронный вход $Q^{n+1} = C(S + \bar{R}Q^n)$ $RS = 0; C = 1$ асинхронный вход $Q^{n+1} = S + \bar{R}Q^n$ $\bar{R} \cdot \bar{S} = 1$ $C = 0$	
T К145ИК15 КИ5ИК14		$Q^{n+1} = TQ^n$ Во время тактового импульса $Q^{n+1} = \bar{T}Q^n$ После окончания тактового импульса	
JK К155ТВ1 К176ТВ1 К561ТВ1 К531ТВ1М К531ТВ1Ю1 К555ТВ6		$Q^{n+1} = J\bar{Q}^n + \bar{K}Q^n$ После окончания тактового импульса	

1	2	3	4
D КИ34ТМ2 КИ55ТМ5 КИ55ТМ2 КИ55ТМ7 КИ76ТМ1 КИ76ТМ2 КИ55НР15		$Q^{n+1} = DC + Q^n \bar{C}$	
DV К551ТМ3		$Q^{n+1} = Q^n$ при $V=0$ $Q^{n+1} = DC + Q^n \bar{C}$ при $V=1$	

Таблица 1.2

Контрольная таблица соответствия триггеров

Тип триггера по нумерации	Входы	Выходы	Неисправность входов	№ нумерации	Входы	Выходы	Неисправность входов
RS-асинхронный	- S R	Q^n Q^{n+1}			- S R	Q^n Q^{n+1}	
	1 - 0 0	0 0	$S=1$	4 - 0 1	1 0	$R=0$ или $S=1$	
	2 - 0 0	1 1	$R=1$	5 - 1 0	0 1	$S=0$ или $R=1$	
	3 - 0 1	0 0	$S=1$	6 - 1 0	1 1	$R=1$	
RS-асинхронный с инверсными входами	\bar{S} \bar{R}	Q^n Q^{n+1}	Неисправность входов	№	\bar{S} \bar{R}	Q^n Q^{n+1}	Неисправность входов
	1 - 1 1	1 1	$\bar{R}=0$	4 - 1 0	0 0	$\bar{S}=0$	
	2 - 1 1	0 0	$\bar{S}=0$	5 - 0 1	1 1	$\bar{R}=0$	
	3 - 1 0	1 0	$R=1$ или $\bar{S}=0$	6 - 0 1	0 1	$\bar{S}=1$ или $\bar{R}=0$	

1	2					3					4					5					6					7					8					9																																																																																																
	№	C	S	R	Q ⁿ	Q ⁿ⁺¹	неисправность входов	№	C	S	R	Q ⁿ	Q ⁿ⁺¹	неисправность входов	№	C	S	R	Q ⁿ	Q ⁿ⁺¹	неисправность входов	№	C	S	R	Q ⁿ	Q ⁿ⁺¹	неисправность входов	№	C	S	R	Q ⁿ	Q ⁿ⁺¹	неисправность входов																																																																																																	
RST - асинхронный	1	0	0	0	0	0	не проверяется	8	0	0	0	1	1	не проверяется	2	0	0	1	0	0	0	3	0	1	0	0	0	C=1	4	0	1	1	0	0	C=1	5	1	0	0	0	0	S=1	6	1	0	1	0	0	S=1	7	1	1	0	0	1	C=0 или S=0 или R=1																																																																												
	2	0	0	1	0	0	не проверяется	9	0	0	1	1	1	0=1	10	0	1	0	1	1	не проверяется	11	0	1	1	1	1	не проверяется	12	1	0	0	1	1	R=1	13	1	0	1	1	0	C=0 или S=1 или R=0	14	1	1	0	1	1	R=1																																																																																			
	3	0	1	0	0	0	C=1	10	0	1	0	1	1	не проверяется	11	0	1	0	1	1	не проверяется	12	1	0	0	1	1	R=1	13	1	0	1	1	0	C=0 или S=1 или R=0	14	1	1	0	1	1	R=1																																																																																										
	4	0	1	1	0	0	C=1	11	0	1	1	1	1	не проверяется	12	1	0	0	1	1	не проверяется	13	1	0	0	1	1	R=1	14	1	1	0	1	1	R=1																																																																																																	
	5	1	0	0	0	0	S=1	12	1	0	0	1	1	не проверяется	13	1	0	1	1	0	C=0 или S=1 или R=0	14	1	1	0	1	1	R=1																																																																																																								
	6	1	0	1	0	0	S=1	13	1	0	1	1	0	C=0 или S=1 или R=0	14	1	1	0	1	1	R=1																																																																																																															
	7	1	1	0	0	1	C=0 или S=0 или R=1	14	1	1	0	1	1	R=1																																																																																																																						
T	№	-	-	T	Q ⁿ	Q ⁿ⁺¹	неисправность входов	№	-	-	T	Q ⁿ	Q ⁿ⁺¹	неисправность входов	1	-	-	0	0	0	T=1	3	-	-	1	0	1	T=0	2	-	-	0	1	1	T=1	4	-	-	1	1	0	T=0																																																																																										
	1	-	-	0	0	0	T=1	3	-	-	1	0	1	T=0																																																																																																																						
	2	-	-	0	1	1	T=1	4	-	-	1	1	0	T=0																																																																																																																						
JK	№	C	J	K	Q ⁿ	Q ⁿ⁺¹	неисправность входов	№	C	J	K	Q ⁿ	Q ⁿ⁺¹	неисправность входов	1	0	0	0	0	не проверяется	9	0	0	0	1	1	не проверяется	2	0	0	1	0	0	не проверяется	10	0	0	1	1	1	C=1	3	0	1	0	0	0	C=1	4	0	1	1	0	0	C=1	5	1	0	0	0	0	J=1	6	1	0	1	0	0	J=1	7	1	1	0	0	1	C=0 или J=0	8	1	1	1	0	1	C=0 или J=0	9	0	0	0	1	1	не проверяется	10	0	0	1	1	1	не проверяется	11	0	1	0	1	1	C=1	12	0	1	1	1	1	C=1	13	1	0	0	1	1	K=1	14	1	0	1	1	0	C=0 или K=0	15	1	1	1	1	0	C=0 или K=0
	1	0	0	0	0	0	не проверяется	9	0	0	0	1	1	не проверяется																																																																																																																						
	2	0	0	1	0	0	не проверяется	10	0	0	1	1	1	C=1																																																																																																																						
	3	0	1	0	0	0	C=1	11	0	1	0	1	1	не проверяется																																																																																																																						
	4	0	1	1	0	0	C=1	12	0	1	1	1	1	C=1																																																																																																																						
	5	1	0	0	0	0	J=1	13	1	0	0	1	1	K=1																																																																																																																						
	6	1	0	1	0	0	J=1	14	1	0	1	1	0	C=0 или K=0																																																																																																																						
	7	1	1	0	0	1	C=0 или J=0	15	1	1	1	1	0	C=0 или K=0																																																																																																																						
8	1	1	1	0	1	C=0 или J=0																																																																																																																														
D - синхронный	№	-	C	D	Q ⁿ	Q ⁿ⁺¹	неисправность входов	№	-	C	D	Q ⁿ	Q ⁿ⁺¹	неисправность входов	1	-	0	0	0	0	не проверяется	5	-	1	0	0	0	D=1	2	-	0	0	1	1	C=1	6	-	1	1	0	1	C=0 или D=0	3	-	0	1	0	0	C=1	7	-	1	1	1	1	D=0	4	-	0	1	1	1	не проверяется																																																																					
	1	-	0	0	0	0	не проверяется	5	-	1	0	0	0	D=1																																																																																																																						
	2	-	0	0	1	1	C=1	6	-	1	1	0	1	C=0 или D=0																																																																																																																						
	3	-	0	1	0	0	C=1	7	-	1	1	1	1	D=0																																																																																																																						
4	-	0	1	1	1	не проверяется																																																																																																																														
DV - синхронный	№	V	C	D	Q ⁿ	Q ⁿ⁺¹	неисправность входов	№	V	C	D	Q ⁿ	Q ⁿ⁺¹	неисправность входов	1	0	0	0	0	не проверяется	4	0	1	1	0	0	V=1	2	0	0	1	0	0	не проверяется	5	0	0	0	1	1	не проверяется	3	0	1	0	0	0	не проверяется	6	0	0	1	1	1	не проверяется																																																																													
	1	0	0	0	0	0	не проверяется	4	0	1	1	0	0	V=1																																																																																																																						
	2	0	0	1	0	0	не проверяется	5	0	0	0	1	1	не проверяется																																																																																																																						
3	0	1	0	0	0	не проверяется	6	0	0	1	1	1	не проверяется																																																																																																																							

1	2	3	4	5	6	7	8	9						
	7	0	1	0	1	1	V=1	12	1	1	1	0	1	V=0
	8	0	1	1	1	1	не проверяется	13	1	0	0	1	0	V=0
	9	1	0	0	0	0	не проверяется	14	1	0	1	1	1	не проверяется
	10	1	0	1	0	0	C=1	15	1	1	0	1	0	V=0 или C=0 или D=1
	11	1	1	0	0	0	D=1	16	1	1	1	1	1	D=0

Поэтому, если значение $Q^{Q'}$ на седьмом наборе не "1", а "0", то на каком-либо из входов C, S, R существует неисправность. Какой из входов C, S или R на этом наборе неисправен, определить невозможно.

Существуют наборы, на которых неисправности входа не проверяются. Например, на пятом наборе $2V$ -синхронного триггера. Переключение на этом наборе триггера из состояния $Q^{Q'} = "1"$ в состояние $Q^{Q'} = "0"$ свидетельствует об одновременном существовании неисправностей двух входов $V = 1, C = 1$. Если существует одна из указанных неисправностей (либо $C = "1"$, либо $C = "1"$), то выход триггера имеет свое истинное значение. Вследствие наличия одной непроверяемой неисправности в графе "Неисправность входов" записывают: "Не проверяется".

Наборы, которые проверяют исправность двух или трех входов, являются предпочтительными, если требуется малым количеством наборов проверить исправность нескольких входов. Если требуется узнать, какой из входов неисправен, используют наборы, устанавливающие единичные неисправности. Однако не для всех видов неисправностей входов такие наборы существуют. Например, для RS -синхронного триггера невозможно установить, какая существует неисправность $C = "0"$ или $S = "0"$. Можно лишь установить, что эти неисправности отсутствуют, если седьмой набор RS -триггера приводит к истинному выходному сигналу.

В табл. I.2 прочерки в клетках свидетельствуют об отсутствии входов, сверх указанных в соответствующих клетках. Например, у T -триггера существует один вход - T , поэтому в двух оставшихся клетках поставлены прочерки.

Примеры контрольных таблиц соответствия для наиболее широко используемых типов триггеров даны в табл. I.2. Если в цифровых узлах используют другие типы триггеров, то для каждого из них

составляет аналогичные контрольные таблицы.

При наличии в одном из плеч триггера нескольких информационных входов схему этого триггера для удобства составления контрольных таблиц преобразуют таким образом, чтобы эти входы являлись входами комбинационной подсхемы, а выход последней был единственным в этом плече входом триггера. Пример подобного преобразования для JK-триггера показан на рис. 1.1.

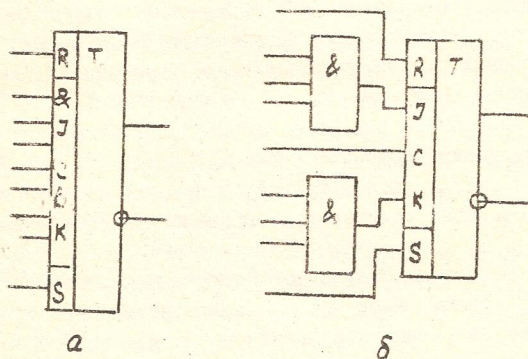


Рис. 1.1. Преобразование триггера: а - JK-триггер с логикой ЭИ на входе; б - преобразованный JK-триггер с комбинационными схемами ЭИ на входе

§ 1.2. Правила составления контрольных тестов для цифровых микросхем и типовых элементов замены с триггерами

Составление теста цифрового узла производят по эквивалентной нормальной форме (ЭНФ) каждой комбинационной подсхемы [1] и контрольным таблицам соответствия триггеров. Основой составления теста является принцип проверки букв в ЭНФ. Множество наборов, проверяющих каждую букву сокращенной ЭНФ (сокращенной обратной ЭНФ), каждой комбинационной подсхемы цифрового узла на фиксированный "0" или "1", является необходимым для получения теста цифрового узла.

Декомпозицию схемы на подсхемы [1] выполняют с выделением триггеров в отдельные подсхемы. Нумеруют все выходы комбинационных элементов и обозначают промежуточные выходы - точки разрывов связей с другими подсхемами, которые образуются при декомпозиции схемы. Обозначают входы и выходы триггеров, входы цифрового узла буквой X с соответствующим порядковым номером.

а выходы - У с порядковым номером.

Составляют контрольную таблицу совмещения, используя форму таблицы совмещения комбинационных схем (I). Для каждого триггера в таблице совмещения проставляют обозначения входов, предыдущих состояний, выходов.

Подсхемы в таблице совмещения следует располагать в соответствии с нумерацией, принятой при декомпозиции схемы.

Первый набор контрольного теста цифрового узла с триггерами - установочный, предназначен для снятия неопределенных состояний триггеров при включении. Триггеры цифровых узлов, не имеющих установочных входов, устанавливают в определенное состояние несколькими наборами.

Строки установочного (первого или нескольких первых) набора заполняют следующим образом:

а) в графах букв - входов триггеров заносит информацию, необходимую для установки триггеров в определенное состояние ("0" или "1");

б) в графы остальных букв информацию заносит произвольно, так как на установочном наборе не проверяют неисправности схемы. При этом избегают запретных комбинаций на входах триггеров.

Формирование набора начинают со стартовой подсхемы. Стартовую подсхему определяют для i -го набора по наименьшему номеру из числа тех подсхем, для которых на предыдущих $(i-1)$ -х наборах не был получен контрольный тест. Следовательно, на первом контрольном наборе стартовая подсхема - первая. Она будет стартовой до тех пор, пока не будет получен контрольный тест этой подсхемы. Далее стартовой подсхемой считают вторую, если на предыдущих наборах для нее не был получен тест и т.д.

Перед формированием каждого i -го контрольного набора следует:

а) перенести из $(i-1)$ -го набора в графы букв каждой подсхемы значения тех переменных, которые являются выходами триггеров;

б) определить номер стартовой подсхемы для i -го набора;

в) определить, является ли выход стартовой подсхемы входом триггера.

Для формирования i -го контрольного набора стартовой подсхемы, выход которой не является входом триггера, следует:

а) определить по ЭНФ (обратной ЭНФ) такой набор, который совместим с проставленными значениями букв из $(i-1)$ -го набора.

Если такой набор получить невозможно, следует установочным набором переключить триггеры в противоположное состояние, перенести значение выходов триггера в соответствующие графы букв последующего контрольного набора и составить по ЗНФ набор, проверяющий буквы термина на фиксированный "0" или "1";

б) определить по ЗНФ и функциям промежуточных выходов стартовой подсхемы значения на выходе и промежуточных выходах; проставить их в строку ζ -го набора;

в) проставить в таблицу для остальных подсхем значения букв, связанных с переменными стартовой подсхемы и определенных на ζ -м наборе;

г) определить наборы, совместимые с состоянием триггеров на ζ -м контрольном наборе, для каждой из подсхем с номером больше номера первой стартовой подсхемы. При этом следует проверить буквы ЗНФ этих подсхем, если проверяющие наборы совместимы с состоянием триггеров.

Если при данном состоянии триггеров и любом их переключении невозможно сформировать проверяющий набор для стартовой подсхемы, то переходят к проверке подсхемы, следующей после стартовой.

Для формирования ζ -го контрольного набора стартовой подсхемы, выход которой является входом одного или нескольких триггеров, используют контрольные таблицы соответствия этих триггеров (табл. I.2). Составляют такие наборы, которые исключают появление на входах триггеров запретных комбинаций сигналов и проверяют неисправности входов триггеров.

После окончания контроля входов всех подсхем с привлечением контрольных таблиц соответствия для триггеров анализируют, все ли входы проверены. При наличии непроверенных входов формируют контрольные наборы для определения их неисправностей.

Наборы для контроля входов подсхем и триггеров обозначают буквой "К", установочные наборы - "У", наборы для контроля только триггеров - "КТ".

Для составления контрольной программы (табл. I.3) используют данные таблицы совмещения и принципиальной схемы [I].

Форма контрольной программы

№ №- блора	Входные наборы						Выходные наборы					
	№ контакта		Разъема				№ контакта		Разъема			
	1	2	1	2	.	.	.	п
1												
2												
:												
п												

§ 1.3. Пример составления контрольного теста микросхемы с RS-асинхронными триггерами

Согласно методике § 1.2 проведено составление контрольного теста микросхемы (рис. 1.2). Декомпозиция схемы проведена так, чтобы у каждой подсхемы был один выход. Поэтому двухвыходной комбинационный элемент на выходе ИС К155АЛ1 [2] 2 - 2 - 2И - 2ИЛИ - И Б/ 2 - 2 - 2И - 2ИЛИ преобразован в две одновыходные подсхемы. Каждой подсхеме на рис. 1.2 присвоен двойной номер. Римская цифра означает номер группы подсхемы, а арабская - номер подсхемы в группе. По методике [1] составлены логические уравнения каждой комбинационной подсхемы в дизъюнктивной нормальной форме - ЗНФ (рис. 1.2).

В схеме есть RS-асинхронные триггеры с инверсным управлением и инверсными выходами. Поэтому из контрольной таблицы соответствия (табл. 1.2) необходимо брать для этого триггера инверсные значения выходных сигналов.

В схеме на рис. 1.2 есть обратные связи от выходов подсхем третьей группы к входам подсхем первой группы. Эти обратные связи являются одновременно промежуточными выходами, поскольку разрываются границами подсхем первой и третьей групп. Сигналы на промежуточных выходах - выходные сигналы микросхемы и входные для подсхем первой группы. Поэтому промежуточные входы введены в контрольную таблицу совмещения (табл. 1.4) как входные для подсхем первой группы.

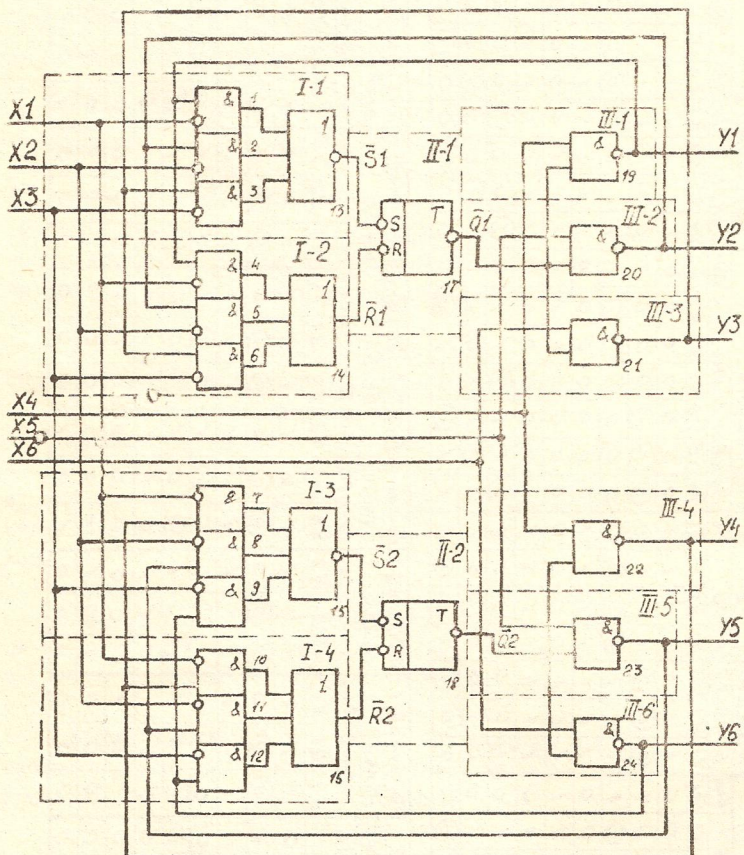


Рис. 1.2. Схема ИС К15БХ11 с разбивкой на подсхемы

(I, II, III - номер группы подсхемы; 1, 2, 3, 4, 5, 6 - ее номер в группе)

$$\text{ЭНФ I-1: } \bar{S}_1 = \bar{Y}_1 \bar{Y}_2 \bar{Y}_3 + X_1 \bar{Y}_2 \bar{Y}_3 + \bar{Y}_1 X_2 \bar{Y}_3 + X_1 X_2 \bar{Y}_3 + \bar{Y}_1 \bar{Y}_2 X_3 + \bar{Y}_1 X_2 X_3 + X_1 \bar{Y}_2 X_3 + X_1 X_2 X_3$$

$$\text{ЭНФ I-2: } R_1 = Y_1 \bar{X}_1 + Y_2 \bar{X}_2 + Y_3 \bar{X}_3$$

$$\text{ЭНФ I-3: } \bar{S}_2 = \bar{Y}_4 \bar{Y}_5 \bar{Y}_6 + X_1 \bar{Y}_5 \bar{Y}_6 + \bar{Y}_4 X_2 \bar{Y}_3 + X_1 X_2 \bar{Y}_6 + \bar{Y}_4 \bar{Y}_5 X_3 + X_1 Y_5 \bar{Y}_3 + \bar{Y}_4 X_2 X_3 + X_1 X_2 X_3$$

$$\text{ЭНФ I-4: } \bar{R}_2 = Y_4 \bar{X}_1 + Y_5 \bar{X}_2 + Y_6 \bar{X}_3$$

$$\text{ЭНФ III-1: } Y_1 = \bar{X}_4 + \bar{Q}_1; \text{ ЭНФ III-2: } Y_2 = \bar{X}_5 + \bar{Q}_1; \text{ ЭНФ III-3: } Y_3 = \bar{X}_6 + \bar{Q}_1$$

$$\text{ЭНФ III-4: } Y_4 = \bar{X}_4 + \bar{Q}_2; \text{ ЭНФ III-5: } Y_5 = \bar{X}_5 + \bar{Q}_2; \text{ ЭНФ III-6: } Y_6 = \bar{X}_6 + \bar{Q}_2$$

Первый установочный набор сформирован при условии переключения инверсных выходов триггеров в состояние "0", для чего на их входы \bar{S} поданы сигналы "0", а на \bar{R} - "1". Стартовыми подхемами для формирования установочного набора являются подсхемы третьей группы, так как от них согласно ЭНФ зависят выходные и одновременно входные сигналы микросхемы.

После установив инверсных выходов триггеров в состояние "0" формируют второй набор - первый контрольный набор. Для этого значения выходных сигналов триггеров на установочном наборе переписывают в графу исходного состояния триггеров на втором наборе и в графу входного набора подсхем третьей группы на том же втором наборе.

Так как входной набор подсхем третьей группы наполовину определен выходными сигналами триггеров, удобно их сделать стартовыми подхемами и для второго набора. Для проверки входов этой подсхем на неисправность "фиксированная 1" надо термам $\bar{X}_4, \bar{X}_5, X_6$ присвоить значение "0". Таким образом, все входы третьей подсхемы на втором наборе в табл. 1.4 проверяются на присутствие неисправности "фиксированная 1", поэтому их обводят кружком.

Значения выходного сигнала по обратным связям передаются на входы подсхем первой группы, поэтому у подсхемы I-1 проверяют входы $\bar{Y}_1, \bar{Y}_2, \bar{Y}_3$ на присутствие неисправности "фиксированная 0", у подсхемы I-2 - входы Y_1, Y_2, Y_3 на "фиксированную 1" и т.д.

Выходы подсхем первой группы переключают триггер на втором наборе в состояние "1", чем проверяется неисправность входов триггеров " $\bar{R} = 1$ " или " $\bar{S} = 0$ " (табл. 1.2).

Таким образом при подаче на входы микросхемы второго набора проверяют неисправности входов, обведенные кружком. Если при подаче второго набора на выходах $Y_1 - Y_6$ происходит переключение из "1" в "0", а затем опять в "0" (так как триггеры за время действия второго набора переключаются по цепи обратной связи в противоположное состояние), то неисправности входов, обведенные кружком, отсутствуют.

За девять наборов проверяют все неисправности входов ИС. Если сигналы на выходе ИС соответствуют указанным выходам подсхем третьей группы, то неисправности входов ИС отсутствуют. Отсутствуют в этом случае и неисправности входов триггеров,

так как правильность их работы проверена в четырех режимах. Комбинация входных сигналов " $\bar{S} = I$ ", " $\bar{Q} = I$ " в данной схеме никогда не создается, поэтому можно считать, что исправность триггеров проверена для данной схемы во всех режимах.

ЛИТЕРАТУРА

1. Лунин К.Б. Автоматизация контроля цифровых субблоков и ТЭВов (модулей I-го уровня) на комбинационных схемах: Методические указания для курсового и дипломного проектирования по курсу "Технология производства ЭВА и РЭА" - М.: МВТУ, 1984. - 24 с.
2. Тарабрина Б.В. и др. Интегральные микросхемы: Справочник /Под ред. Тарабрина Б.В. - М.: Радио и связь, 1983. - 528 с.

Глава 2. ТЕХНОЛОГИЧЕСКИЕ ОСНОВЫ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ

2.1. Методические основы разработки правил технологического проектирования

В основе построения имитационной модели M лежит анализ условий, влияющих на формирование проектных решений. Эти условия играют роль предикторов применимости правил экстраполяции (правил технологического проектирования), содержащих действия по изменению ситуаций в модели M .

Для изменения конструкторских ситуаций предназначены правила экстраполяции, с помощью которых происходит дополнение входного описания детали недостающими размерами (группа G_1^2) и выбор ЭПО* (группа G_2^2). Изменение технологических ситуаций вызывают правила, с помощью которых производится, например: расчет межоперационных размеров (группа G_1^T), выбор типоразмеров инструментов (группа G_2^T), вычисление режимов резания (группа G_3^T) и т.д.

Каждое правило экстраполяции характеризуется последовательностью шагов его выполнения, либо связано с привлечением программного модуля, если решается расчетная задача. Например, выбор ЭПО можно рассматривать как многошаговый процесс струк-

* ЭПО - элементарные планы обработки.

тального преобразования детали в соответствии с конечной целью проектирования. При решении других задач изменяются лишь значения отдельных параметров объектов в анализируемых ситуациях. Таким образом, благодаря экстраполяции шагами исходной ситуации представляется возможность осуществлять программный прогноз решений, т.е. находить наилучший путь достижения конечной цели в определенный момент последующего развития процесса проектирования.

Совокупность правил экстраполяции образует трансформационную грамматику G языка системы автоматизированного проектирования технологических процессов (САТП). В общем виде правила G записываются:

$$X_R = \{ [(x_1, \bar{z}_1, \bar{z}_2) \wedge (x_2, \bar{z}_2, x_3) \wedge \dots \wedge (x_n, \bar{z}_n, x_{n+1})], R \}, \quad (2.1)$$

где выражение в квадратных скобках определяет класс входной ситуации; X_R - решение, принимаемое по входной ситуации.

На формирование проектных решений в условиях той или иной ситуации оказывают влияние два ряда факторов: параметры объектов проектирования, параметры производственной среды (состав применяемого оборудования, традиционные приемы обработки, нормы предприятия и др.).

Анализ первых из них ведет к вариантности принимаемых решений. С учетом вторых всегда выбирается вполне определенное для данного производства решение. Исходя из этого, будем различать два возможных подхода к разработке правил G - по схемам "наложения" ситуаций и "привязки" решений.

Первая схема заключается в следующем: по результатам разузлования обучающей выборки необходимо выделить все входные ситуации, по которым принимаются одинаковые решения; через информацию об объектах и среде определить условия выбора данных решений. Такой подход целесообразно использовать, когда ставится задача разработки какого-либо программного модуля для установления однозначного соответствия между объектами входных ситуаций и решениями.

Если ставится задача построения более гибкого алгоритма принятия решений (как в случае выбора ЭПО), то наиболее приемлемым следует признать второй подход к разработке правил G , согласно которому анализ входных ситуаций должен производиться без учета факторов, характеризующих среду. Для этого нужно свя-

чала задать предикаты Pr_{κ} ($\kappa = \overline{1, K}$) применимости правил, а затем с их помощью выявить все альтернативные решения, принимаемые по входным ситуациям.

допустим, в обучающей выборке содержится некоторое множество $Md^d = \{d_i^d, i = \overline{1, n}\}$ описаний поверхностей, обрабатываемых по одному и тому же плану h_j . Тогда можно записать:

план h_j $\left\{ \begin{array}{l} \text{будет включен в структуру } \kappa\text{-го правила экстраполяции, если } \exists d_i^d \in Md^d, \text{ для которого справедливо} \\ Pr_{\kappa} d_i^d = 1; \\ \text{не будет включен в структуру } \kappa\text{-го правила, если} \\ \text{для } \forall d_i^d \in Md^d, Pr_{\kappa} d_i^d = 0. \end{array} \right.$

где

$$Pr_{\kappa} d_i^d = V_{\ell} (\Delta_{\ell} x_{\ell p}^{\kappa} z_{\ell} x_{\ell c}^d) - \quad (2.2)$$

оценка применимости κ -го правила в условиях конкретной ситуации;

$x_{\ell p}^{\kappa} - p - e$ - граничное значение ℓ -го параметра в составе Pr_{κ} ;

$x_{\ell c}^d$ - значение ℓ -го параметра i -й поверхности, обрабатываемой по плану h_j ; z_{ℓ}^{κ} - отношение порядка ($<, >, \leq, \geq, =, \neq$) между $x_{\ell p}^{\kappa}$ и $x_{\ell c}^d$.

Множество Mh^{κ} альтернативных ЭЮ, включаемых в структуру κ -го правила экстраполяции, получим после того, как в соответствии с (2.2) будет оценена применимость этого правила в условиях ситуаций, определяемых свойствами поверхностей из множеств $Md^1, Md^2, Md^3, Md^{\kappa}, \dots$, где Md^{κ} - общее число ЭЮ, используемых при обработке поверхностей κ -й группы ($\kappa = \overline{1, K}$). В результате можно записать:

$$\bigcup_{\kappa=1}^K Mh^{\kappa} = Mh, \quad \bigcap_{\kappa=1}^K Mh^{\kappa} \neq \emptyset, \quad (2.3)$$

где Mh - множество всех ЭЮ, используемых при обработке поверхностей.

Описанная схема построения правил выбора ЭЮ приведена на рис. 2.1. Процесс проектирования представляет иерархическую разветвленную структуру. Сходимость процесса обеспечивается правилами коррекции и различными критериями отбора частных и

общего решения. Правила коррекции задаются последовательностью целей вида:

$$(x, \bar{z}, \alpha_2) \wedge (\alpha_3, \bar{z}_3, \alpha_4) \wedge \dots \wedge (\alpha_n, \bar{z}_n, \alpha_{n+1}). \quad (2.4)$$

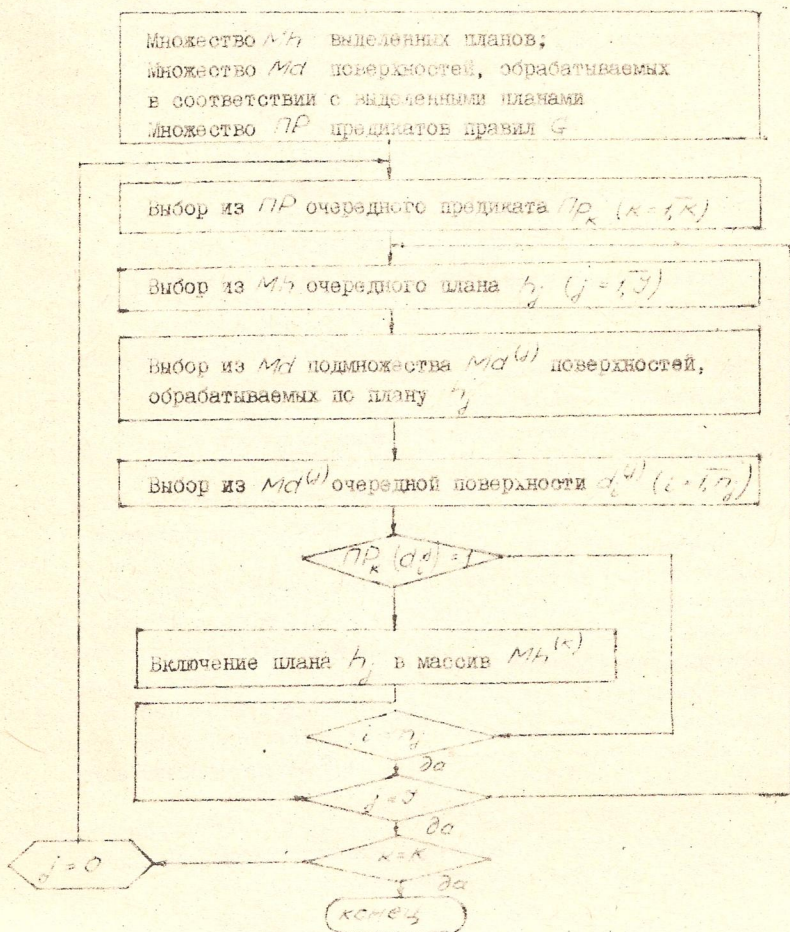


Рис. 2.1. Схема "приближки" элементарных планов обработки к правилам технологического проектирования

Ситуация (2.4) считается распознанной правилами, если $x_i \geq X_i$ и если для любой тройки $(x_i, \bar{x}_k, x_{i+1})$ отношение x_k и его знак ориентации совпадают с отношением и знаком тройки $(X_i, \bar{x}_k, X_{i+1})$. Иначе ситуация считается противоречивой.

При разработке правил выбора ЭПО для каждого из них должно выполняться условие

$$|Mh^k| \leq 4m_0, \quad (2.5)$$

где $|Mh^k|$ — мощность множества Mh^k ; m_0 — число опорных элементарных планов, применяемых в производстве деталей.

2.2. Классификация элементарных планов и условия их применяемости

Прежде чем приступить к построению модели M , определим состав информации, необходимый и достаточный для выбора ЭПО:

1) общие сведения о детали и заготовке (оказывают влияние на выбор этапов обработки детали);

2) вид поверхности, ее габаритные размеры и параметры качества (влияют на выбор схем достижения заданной точности поверхности);

3) параметры, описывающие относительное расположение поверхности (влияют на выбор схем базирования и последовательность операций в технологическом процессе).

Исследование первой закономерности позволяет для рассматриваемого класса деталей сформулировать множество MA' логических условий выбора опорных планов, а исследование остальных закономерностей — множество MA'' логических условий выбора ЭПО поверхностей. Последнее обусловлено тем, что ЭПО любой поверхности есть не что иное как сочетание некоторой схемы достижения заданной точности с некоторой последовательностью операций.

Обозначим через $MA = MA' \cup MA'' = \{A_q, q = \overline{1, Q}\}$ конечное множество сформулированных логических условий. Тогда выражение (2.2) можно представить как логическую функцию вида

$$P_{\kappa} = \left(\bigwedge_{q=1}^{\ell} A_q \right)_{\kappa}, \quad (2.6)$$

где ℓ — максимально допустимое количество условий, связанное с обязательным их выполнением при выбранной точности разработки

Чтобы перейти к формальной записи логических функций (2.6), построим сначала схему классификации применяемых ЭЮ, ограничиваясь при этом рассмотрением деталей класса "Тела вращения" (длиной $L \leq 2Z$), получаемых из круглого проката. В частности, для классификации опорных планов воспользуемся логической схемой, описываемой последовательностью следующих вопросов:

1. Деталь имеет сложную конфигурацию?
2. Термообработка детали (закалка и отпуск) предусматривается перед чистовым или окончательным этапом?
3. Перед термообработкой требуется производить цементацию?

В соответствии с данной схемой можно выделить такие классы опорных планов, как "Обработка сложных по конфигурации деталей, не подвергаемых закалке", "Обработка простых по конфигурации деталей, подвергаемых закалке с цементацией" и др.

Если перечисленные выше вопросы выразить через задаваемые сведения о детали и заготовке, то в итоге получим первое искомого множество $MA \subset MA$ необходимых логических условий. Эти условия приведены в табл. 2.1, данной ниже. Они вводятся для того, чтобы выполнялось требование однозначности выбора опорного плана. В качестве примера опишем правило экстраполяции:

$$(\forall \hat{D} \in Md) [(A_1 \wedge \bar{A}_2 \wedge A_3 \wedge \bar{A}_4 \wedge A_5) \supset (\hat{D} \in \hat{H}_3)]$$

что означает: если деталь \hat{D} , получаемая из прутка и закалываемая по твердости $HRC \leq 45$, отнесена к категории несложных и мелких (диаметром меньше 30 мм), то для ее обработки может быть выбран опорный план H_3 , согласно которому закалка детали должна выполняться сразу же после заготовительной операции, где \bar{A}_2 и A_4 - логические условия со знаком инверсии; H_3 является элементом множества применяемых классов опорных планов.

Таблица 2.1

Условия выбора опорных планов обработки деталей

Обозначение	Условие	Описание условия
A_1	$x_{21}^0 \neq 1 \vee 2$	Иметь уступы в наружном контуре
A_2	$x_{22}^0 \neq 1 \vee 2$	Иметь уступы во внутреннем контуре
A_3	$x_{34}^0 \leq 30$	Иметь максимальный диаметр $D \leq 30$
A_4	$x_{24}^0 = 0$	Не подлежать закалке
A_5	$x_{25}^0 \leq 45$	Иметь твердость $HRC \leq 45$
A_6	$x_{23}^0 = 1$	Использовать материал, не упроченный термообработкой

Для осуществления выбора элементарных планов обработки поверхностей за основу примем систему условий, описываемую следующим образом:

1. Требуется ли поверхность многократной обработки?
2. Связана ли обработка поверхности с введением черного этапа?
3. Должна ли быть поверхность защищена от цементации?
4. Производится ли завершающая обработка поверхности в начале или в конце чистового этапа?
5. Производится ли обработка поверхности от установочной базы повышенной точности?
6. Может ли выполняться обработка поверхности только с одной или с разных сторон детали?

Анализ аналитических возможностей изложенной схемы приводит к выявлению второго искомого множества $MA^c MA$ необходимых логических условий. Эти условия приведены в табл. 2.2. Они имеют объективный характер и не зависят от параметров производственной среды. Поясним их смысловое значение.

Условия $A_7 - A_9$ в совокупности с условиями $A_{15} - A_{17}$ служат для определения ориентировочного числа технологических переходов, требуемых для обработки поверхности.

Условие A_{10} вносит некоторое уточнение в состав предполагаемых технологических переходов. В частности, выполнение условия A_{10} исключает обработку отверстий методом зенкерования и растачивания.

Условия A_{13} и A_{14} ставят вопрос о необходимости использования планов, предусматривающих дополнительную обработку поверхности в этапах \mathcal{E}_2 или \mathcal{E}_0 (после меднения поверхности или оставления повышенной припуска при защите от цементации).

Условия $A_{18} - A_{22}$ касаются предварительного определения места обработки поверхности в общей структуре технологического процесса. Очевидно, чем сложнее размерная цепь, координирующая положение ЭОП по оси OY или относительно крайнего торца детали, тем больше появляется оснований для обработки таких ЭОП на заключительных операциях получистового и чистового этапов. И наоборот, в случае простой координации поверхностей более рациональной может оказаться их обработка в начале проектируемого процесса, например, с целью подготовки чистовых установочных баз.

Условия выбора элементарных планов обработки поверхностей

Обозначение	Условие	Описание условия
A_7	$x_{30}^0 = 1$	Иметь круглую форму
A_8	$x_{64}^0 \neq 0$	Иметь резьбу
A_9	$x_{65}^0 \neq 0$	Иметь накатку
A_{10}	$x_{34}^0 < 6$	Иметь диаметр $D < 6$
A_{11}	$x_{14}^0 > 0,5$	Иметь соотношение $\frac{\rho}{L_{get}} \geq 0,5$
A_{12}	$x_{15}^0 \geq 0,5$	Иметь соотношение $\frac{\rho^*}{L_{get}} \geq 0,5$
A_{13}	$x_{74}^0 = 0$	Не подвергать термообработке
A_{14}	$x_{74}^0 = 1$	Не подвергать цементации
A_{15}	$x_{39}^0 = 6,3$	Иметь шероховатость $Rz \leq 6,3$
A_{16}	$x_{39}^0 = 10$	Иметь шероховатость $Rz = 10$
A_{17}	$x_{39}^0 = 20 \vee 40$	Иметь шероховатость $Rz = 20 \vee 40$
A_{18}	$x_{40}^0 = . \ell$	Длина ℓ поверхности вычисляется алгоритмически (перед значением ℓ стоит точка)
A_{19}	$x_{57}^0 = . \ell^*$	Расстояние ℓ^* до крайнего торца детали вычисляется алгоритмически (перед значением ℓ^* стоит точка)
A_{20}	$x_{58}^0 = 1 \vee 2$	Иметь код 1 или 2 координации относительно крайнего торца детали
A_{21}	$x_{51}^0 = 0X$	Иметь базой B_y ось OX детали
A_{22}	$x_{52}^0 = 0$	Не иметь координирующего размера по оси OZ детали
A_{23}	$x_{68}^0 = 2 \vee 3 \vee 5$	Иметь допустимое отклонение от концентричности, перпендикулярности или параллельности поверхности
A_{24}	$x_{31}^0 = 1 \vee 2 \vee 5$	Являться наружной поверхностью
A_{25}	$x_{31}^0 = 1 \vee 4$	Быть открытой
A_{26}	$x_{31}^0 = 2 \vee 5$	Быть полуоткрытой слева

Условия A_{11}, A_{12}, A_{23} предназначены для выяснения того, какая схема установки может быть принята для достижения заданной точности относительного взаимного положения поверхностей.

Наконец, условия $A_{24} - A_{26}$ с учетом всех остальных условий позволяют выбрать элементарные планы, относящиеся к обработке поверхностей наружного или внутреннего контуров левой или правой стороны детали.

В соответствии с выделенными логическими условиями общее число возможных правил экстраполяции (а следовательно, и классов ЭПО) составит 2^{20} . На самом деле при анализе входных ситуаций проверяется выполнимость только тех комбинаций условий, которые не противоречат логике правдоподобных рассуждений. Как оказалось, максимальное число условий в каждой осмысленной комбинации (2.6) не превышает 10, а число реально используемых правил для каждой группы поверхностей не превышает нескольких десятков.

Наибольшей трудоемкостью отличается разработка правил выбора ЭПО цилиндрических поверхностей (d_{01}). В качестве примера приведем одно из них (g_{κ}^2):

$$(\forall d_{01}^{\wedge} \times d_{01}^{\vee}) [(A_{24} \wedge A_{26} \wedge \bar{A}_{19} \wedge \bar{A}_{18} \wedge \bar{A}_{12} \wedge A_{13} \wedge A_{16} \wedge A_{18} \wedge \bar{A}_{19} \wedge A_{23}) \Rightarrow (d_{01}^{\wedge} \bar{z}_1 h_a) \wedge (d_{01}^{\vee} \bar{z}_1 h_b) \wedge \dots \wedge (d_{01}^{\wedge} \bar{z}_1 h_c)],$$

где $h_a, h_b, \dots, h_c \in MR^{\wedge}$ - множество альтернативных планов обучающей выборки, отвечающих условиям данного правила (κ -го класса ситуации).

Описанная логическая схема выбора ЭПО не охватывает всего многообразия факторов, влияющих на построение структуры технологического процесса. Однако она является вполне достаточной для того, чтобы компенсировать исходную неопределенность при последующем сравнительном анализе вариантов проектирования.

2.3. Разработка граф-схем алгоритмов выбора элементарных планов обработки поверхностей

Вторая группа G_2^D правила экстраполяции может быть задана в виде табличных или блочных алгоритмов.

Табличные алгоритмы значительно повышают адаптивные свойства моделей проектирования. Однако их использование требует повышенных затрат машинного времени.

Блочный (граф-схемный) способ представления алгоритмов позволяет сократить время поиска решений до минимума, но при корректировке моделей часто приводит к большим изменениям программы. В случае выбора ЭЮ применение блочных алгоритмов полностью себя оправдывает, так как параметрический анализ входных ситуаций рассматривается как устойчивый и объективно сложившийся процесс.

Для каждого класса поверхностей характерен свой набор логических условий выбора ЭЮ. Некоторые из наборов являются общими. В соответствии с этим множество MA'' логических условий можно разбить на следующие подмножества:

$$MA'' = \bigcup_{\nu=1}^7 MA''_{\nu}, \quad \bigcap_{\nu=1}^7 MA''_{\nu} \neq \emptyset, \quad (2.7)$$

где MA''_1 - условия выбора ЭЮ цилиндрических поверхностей;

MA''_2 - условия выбора ЭЮ конусов, сфер, фасонных поверхностей и многогранников; MA''_3 - условия выбора ЭЮ пазов;

MA''_4 - условия выбора ЭЮ отверстий; MA''_5 - условия выбора ЭЮ плоскостей, лысок и уступов; MA''_6 - условия выбора ЭЮ прорезей (окон); MA''_7 - условия выбора ЭЮ канавок.

Разработка блочных алгоритмов производится на основе составления логических функций (2.6) с учетом разбиения (2.7).

Положим, что на каждом шаге алгоритма выбора ЭЮ поверхностей ν -й группы $|\nu=1-7|$ проверяется выполнимость одного логического условия $A_q \in MA''_{\nu}$. Задача заключается в том, чтобы установить такую последовательность проверки условий из MA''_{ν} , которая обеспечивает построение граф-схем с минимальным числом дуг.

Формальные методы построения оптимальных граф-схем весьма трудоемки, поэтому ограничимся практическими мерами. Избегая перебора возможных вариантов упорядочения множества MA''_{ν} , оценим прежде частоту повторения $|MA_q|$ каждого условия относительно соответствующего множества логических функций и организуем не возрастающую по величине α последовательность проверки условий. Тем самым мы сохраняем естественную тенденцию к достижению в процессе экстраполяции конечных решений как можно раньше, оставляя проверку "уникальных" условий на последние шаги процесса.

При одинаковой частоте повторения отдельных условий в первую очередь выбирается то из них, которое относится к параметру с наибольшим числом граничных значений. Если в таких условиях оказалось несколько, то для определенности следует выбрать условие с наименьшим индексом.

Упорядоченные последовательности проверки условий множеств MA_{2j}'' представлены ниже:

$$MA_1'' = \{A_{13}, A_{14}, A_{24}, A_{23}, A_{26}, A_9, A_8, A_{17}, A_{16}, A_{15}, A_{18}, A_{19}, A_{20}, A_7, A_{10}, A_{11}, A_{12}, A_{25}\};$$

$$MA_2'' = \{A_{13}, A_{14}, A_{17}, A_{16}, A_{24}, A_{25}, A_{26}\};$$

$$MA_3'' = \{A_{13}, A_{14}, A_{22}, A_{21}, A_{24}, A_{17}, A_{23}\};$$

$$MA_4'' = \{A_{15}, A_{14}, A_{22}, A_{21}, A_{17}, A_{23}\};$$

$$MA_5'' = \{A_{13}, A_{14}, A_{22}, A_{21}\};$$

$$MA_6'' = \{A_{13}, A_{19}, A_{22}, A_{21}\};$$

$$MA_7'' = \{A_{13}, A_{14}, A_{21}, A_{25}, A_{26}, A_{24}, A_{17}, A_{23}\}$$

Пример построения граф-схемы алгоритма по одной из указанных последовательностей изображен на рис. 2.2. Нумерация висячих вершин граф-схемы соответствует изменению переменной "к" в предполагаемых подмножествах MA^k альтернативных ЭПО

($k=1, K, v=\bar{v}$). Взаимная согласованность выбираемых ЭПО достигается с помощью правил (2.4), описывающих все допускаемые сочетания существующих опорных планов с планами обработки поверхности v -й группы.

На заключительной стадии проектирования при подготовке к печати окончательного варианта технологического процесса опорный план дополняется необходимыми слесарными и гальваническими операциями, ссылками на используемые инструкции, а планы обработки поверхностей - переходами по обработке фасок, скруглений, технологических канавок и подрезаемых торцов.

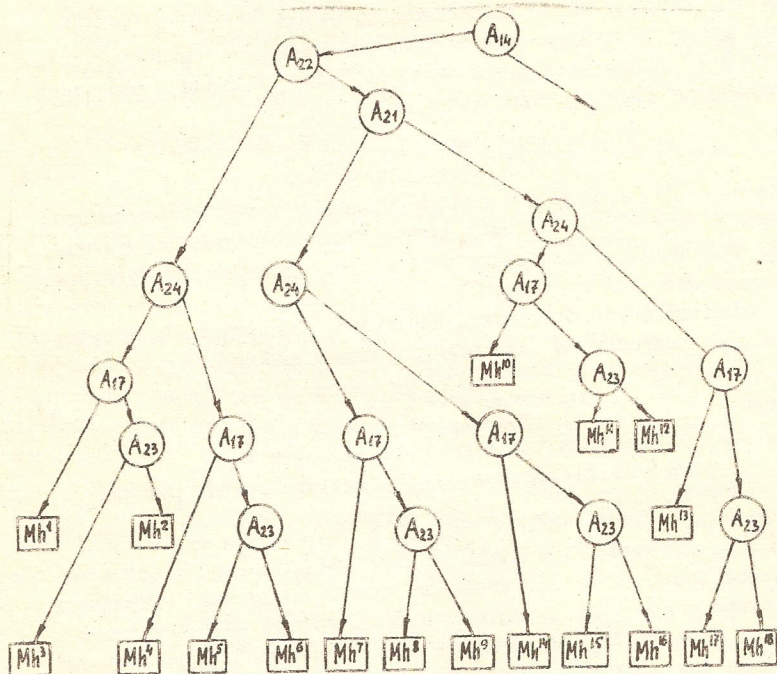


Рис. 2.2. Граф-схема выбора элементарных планов обработки "пазов"

Глава 3. ОПТИМИЗАЦИЯ СТРУКТУРЫ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА

3.1. Определение величин уступок

При решении задачи оптимального технологического проектирования методом последовательных уступок в качестве оптимальных могут выступать лишь эффективные решения.

Решение T_r является эффективным, если по отношению к любому другому решению T_r' справедливо $W(T_r) \leq W(T_r')$, либо среди неравенств $W_2(T_r) \leq W_2(T_r')$, ($r=1,2,3$) всегда найдется хотя бы одно строгое.

Располагая множеством MT обучающих примеров проектирования технологических процессов, каждый из которых рассматривает-

ся как оптимальный по глобальному критерию W , можно решить эту задачу следующим образом. Вначале определим величину уступки ΔW_1 , находя наибольшую разность между заданным и расчетным значениями первого критерия W_1 :

$$\Delta W_1 = \max_{T^* \in MT} [W_1(T^*) - W_{10}], \quad (3.1)$$

где $W_1(T^*)$ - число операций в обучающем технологическом процессе $T^* \in MT$; W_{10} - минимальное число операций, получаемое при машинном проектировании того же технологического процесса (по условиям той же детали).

Отсюда можно определить допустимое относительное отклонение $\Delta \bar{W}_1$ критерия W_1 для всей обучающей выборки:

$$\Delta \bar{W}_1 = \frac{\Delta W_1}{W_{1, \max}}, \quad (3.2)$$

где $W_{1, \max}$ - максимальное значение критерия W_1 .

С учетом (3.2) критерий W_1 рассчитывается по формуле

$$W_1 = W_{10}(1 + \Delta \bar{W}_1), \quad (3.3)$$

что дает возможность в каждом конкретном случае определять некоторое множество Ω_1 решений для поиска среди них оптимальных по второму критерию. Исходя из этого, находим величину уступки ΔW_2 как наибольшую разность между заданным и расчетным значениями критерия W_2 :

$$\Delta W_2 = \max_{T^* \in MT} [W_{20} - W_2(T^*)], \quad (3.4)$$

где $W_2(T^*)$ - оценка качественного содержания комплексного плана обработки детали, соответствующего обучающему технологическому процессу $T^* \in MT$; W_{20} - максимальное значение качества, получаемое при машинном проектировании того же технологического процесса (по условиям той же детали).

В результате определяем допустимое относительное отклонение $\Delta \bar{W}_2$ критерия W_2 для всей обучающей выборки:

$$\Delta \bar{W}_2 = \frac{\Delta W_2}{W_{2, \max}}, \quad (3.5)$$

где $W_{2, \max}$ - максимальное значение критерия W_2 .

Отсюда следует

$$W_2 = W_{20}(1 - \Delta \bar{W}_2), \quad (3.6)$$

что дает возможность в каждом конкретном случае определять некоторое множество Ω_2 решений для поиска среди них оптимальных

по третьему критерию и т.д.

Далее на основании анализа значений всех частных критериев производится выбор весовых коэффициентов для принятия окончательных решений по глобальному критерию оптимальности.

3.2. Выбор метода решения задачи по критерию минимума числа операций

Оптимизация структуры проектируемого процесса по критерию W , интерпретируется в виде задачи о нахождении покрытия с минимальным весом некоторого графа $G = (M_d, M_h, R)$, вершинами которого являются элементарные поверхности и допустимые планы их обработки соответственно из множеств M_d и M_h , а дугами - отношения между ними (типа γ_i - "иметь имя") из множества R . Покрытием графа G называется такое множество дуг $R' \subset R$, при котором любая вершина графа инцидентна по крайней мере одной дуге из R' .

Представляя граф G в виде булевой матрицы $\|y_{ij}\|$ взаимосвязи ЭОП и ЭНО, можно определить покрытие как такой набор единиц, при котором каждая строка и каждый столбец матрицы содержит не менее одного элемента из этого набора, где

$$y_{ij} = \begin{cases} 1, & \text{если поверхность } \hat{a}_i \text{ может быть обра-} \\ & \text{ботана по плану } h_j, \quad i = \overline{1, L}, j = \overline{1, M}; \\ 0 & \text{в противном случае.} \end{cases} \quad (3.7)$$

Пусть также задана матрица $\|z_{ej}\|$, где

$$z_{ej} = \begin{cases} 1, & \text{если план } h_j \text{ имеет в своем составе} \\ & \text{операцию } c_e, \quad j = \overline{1, M}, e = \overline{1, N} \\ 0 & \text{в противном случае.} \end{cases} \quad (3.8)$$

Требуется найти все подмножества H_p множества M_h , которые будучи покрытиями матрицы $\|y_{ij}\|$ имеют в своем составе минимальное число операций $p = \overline{1, P}$.

Введем булевый вектор $X = (x_1, x_2, \dots, x_j, x_M)$ такой, что

$$x_j = \begin{cases} 1, & \text{если план } h_j \text{ войдет в какое-либо покрытие;} \\ 0 & \text{в противном случае.} \end{cases} \quad (3.9)$$

Обозначим через x_j $y_{ij} = 1$ пригодность для обработки поверхности \hat{a}_i плана h_j , выбираемого в покрытие. Тогда условием выполнимости обработки данной поверхности при использовании планов, определяемых вектором X , будет

$$\bigvee_{j=1}^M (x_j \wedge y_{ij}) = 1. \quad (3.10)$$

Распространяя это условие на все элементы множества M_i , получим

$$\bigwedge_{i=1}^L (\bigvee_{j=1}^M (x_j \wedge y_{ij})) = 1. \quad (3.11)$$

Поступая аналогично, обозначим через $x_j \wedge z_{je} = 1$ входимость операции C_e в состав выбираемого в покрытие плана y_j . Тогда условием входимости операции C_e в один из планов вектора X будет

$$\bigvee_{j=1}^M (x_j \wedge z_{je}) = 1. \quad (3.12)$$

В результате критерий W_1 примет вид:

$$W_1 = m \wedge \sum_{e=1}^N (\bigvee_{j=1}^M (x_j \wedge z_{je})), \quad (3.13)$$

где $1 < W_1 \leq N$.

Перейдем к разработке последовательного алгоритма решения поставленной задачи.

3.3. Последовательный алгоритм определения оптимальных покрытий

При разработке алгоритма решений задачи воспользуемся свойством выпуклости множества всех покрытий матрицы $\|y_{ij}\|$: если некоторое подмножество M_p элементарных планов из M_n является столбцовым покрытием строк заданной матрицы, то и любое его расширение, получаемое добавлением в M_p нового плана, также будет покрытием.

Это свойство позволяет в какой-то степени упростить задачу и свести ее решение к двум этапам:

1) нахождению нижней границы решения, образованного простыми (безызыточными) покрытиями;

2) выбору среди простых покрытий оптимальных по критерию W_1 .

Рассмотрим первый этап, подразумевая под простыми покрытиями такие подмножества планов из M_n , которые являются покрытиями, но перестают быть ими при удалении любого плана.

Будем считать, что на i -м шаге алгоритма должна быть построена некоторая матрица Π_i^* , фиксирующая все покрытия строк

y_1, y_2, \dots, y_i заданной матрицы $\|y_{ij}\|$. Обозначим через $X_j(i)$ булевый вектор, описывающий одно из возможных покрытий строки y_i при $x_j \wedge y_{ij} = 1$, а через Π_i - матрицу, описывающую все покрытия строки y_i . Тогда описание

текущей ситуации на i -м шаге поиска решений можно записать в виде:

$$P_i^* = P_i^* \cup P_i, \quad i=1, 2, \dots, 6. \quad (3.14)$$

Редуцирование текщих ситуаций будем производить с помощью следующего универсального правила: если в матрице P_i^* имеется такая пара вектор-строк X_m и X_n , что X_m поглощает X_n , т.е.

$X_n \subseteq X_m$, то вектор-строка X_n вычеркивается. Это правило универсально в том смысле, что требования, предъявляемые к покрытию со стороны вектора X_m , полностью удовлетворяются при соблюдении требований вектора X_n . Следовательно, вектор оказывается излишним.

действие указанных процедур продемонстрируем на конкретном примере, используя фрагмент детали, изображенный на рис. 3.1. Допустим, в этом случае соответствие между M_d и M_H можно представить в виде графа G , показанного на рис. 3.2а.

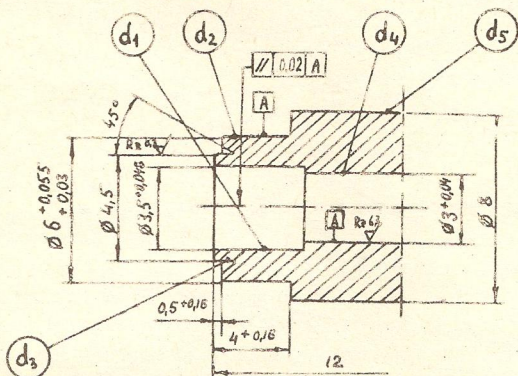


Рис. 3.1. Фрагмент конструкции обрабатываемой детали

Для заданного графа построим матрицу $\|y_{ij}\|$, в которой все планы упорядочены по предпочтению согласно их приоритетам от худших к лучшим:

$$\|y_{ij}\| = \begin{matrix} & h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 & h_9 & h_{10} & \left. \begin{matrix} d_1 \\ d_2 \\ d_3 \\ d_4 \\ d_5 \end{matrix} \right\} \\ \begin{matrix} 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{matrix} \end{matrix}$$

и определим все простые покрытия.

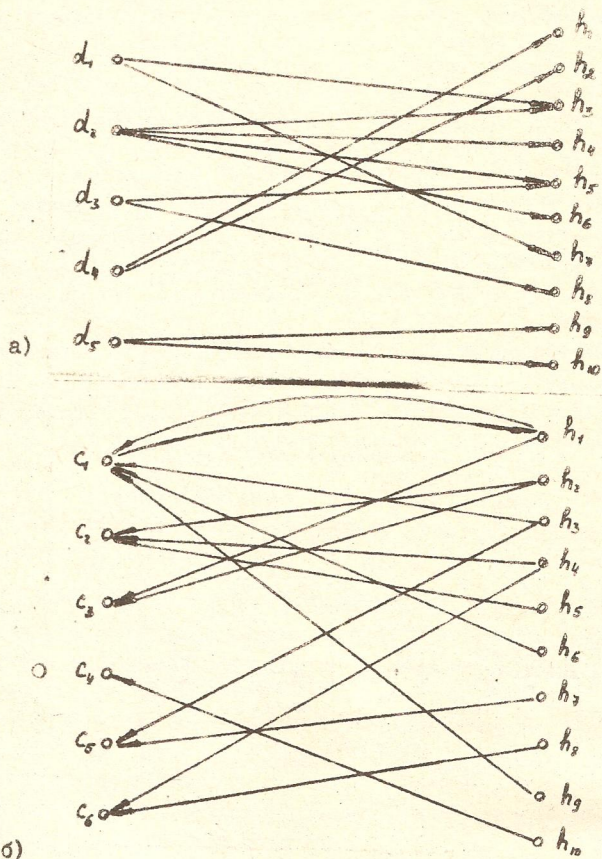


Рис. 3.2. Исходные данные для решения задачи по критерию минимума числа операций: а) граф соответствия между элементами детали и планами их обработки; б) граф соответствия между планами обработки и технологическими операциями

1. $i=1$; $\Pi_1^* = \Pi_0^* \vee \Pi_1$. Так как Π_0^* равно пустому множеству, то $\Pi_1^* = \Pi_1$, где

$$\Pi_1 = \begin{bmatrix} h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 & h_9 & h_{10} \\ \left[\begin{array}{cccccccccc} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{array} \right] & X_3(i) \\ X_7(i) \end{bmatrix}$$

2. $\bar{L} = L + 1$; $\Pi_2^* = \Pi_1^* \vee \Pi_2$, где

$$\Pi_2 = \begin{array}{cccccccccccc} h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 & h_9 & h_{10} & & \\ \left[\begin{array}{cccccccccccc} 0 & 0 & I & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & I & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & I & 0 & 0 & 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & 0 & I & 0 & 0 & 0 & 0 & 0 & \end{array} \right. \begin{array}{l} X_3(2) \\ X_4(2) \\ X_5(2) \\ X_6(2) \end{array} \end{array}$$

Производя поразрядную дизъюнкцию каждой строки матрицы Π_1^* с каждой строкой матрицы Π_2 , получим все возможные покрытия строк y_1 и y_2 матрицы $\|y_{ij}\|$:

$$\Pi_2^* = \begin{array}{cccccccccccc} h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 & h_9 & h_{10} & & \\ \left[\begin{array}{cccccccccccc} 0 & 0 & I & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \\ 0 & 0 & I & I & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \\ 0 & 0 & I & 0 & I & 0 & 0 & 0 & 0 & 0 & 0 & \\ 0 & 0 & I & 0 & 0 & I & 0 & 0 & 0 & 0 & 0 & \\ 0 & 0 & I & 0 & 0 & 0 & I & 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & I & 0 & 0 & I & 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & I & 0 & I & 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & 0 & I & I & 0 & 0 & 0 & 0 & \end{array} \right. \begin{array}{l} X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \\ X_7 \\ X_8 \end{array} \end{array}$$

3. Применяя правило редуцирования матрицы Π_2^* , вычеркиваем векторы X_2, X_3, X_4 и X_5 , поглощающие X_1 . Оставшиеся векторы X_6, X_7 и X_8 являются простыми покрытиями строк y_1 и y_2 .

4. Проверяем условие $\bar{L} = L$. Если оно не выполняется, то возвращаемся к п. 2 алгоритма. В противном случае осуществляется переход к последующему этапу решения задачи, связанному со взвешиванием каждого из найденных покрытий матрицы $\|y_{ij}\|$.

Продолжая процесс, нетрудно убедиться, что для рассматриваемого примера результирующая матрица Π_5^* будет содержать 20 простых покрытий:

$$\Pi_5^* = \begin{array}{cccccccccccc} h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 & h_9 & h_{10} & & \\ \left[\begin{array}{cccccccccccc} I & 0 & I & 0 & I & 0 & 0 & 0 & I & 0 & 0 & \\ I & 0 & I & 0 & I & 0 & 0 & 0 & 0 & I & 0 & \\ 0 & I & I & 0 & I & 0 & 0 & 0 & I & 0 & 0 & \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \\ 0 & I & 0 & 0 & 0 & I & I & I & 0 & I & 0 & \end{array} \right. \begin{array}{l} X_1 \\ X_2 \\ X_3 \\ \dots \\ X_{20} \end{array} \end{array}$$

т.е. для обработки поверхностей заданной конструкции могут быть выбраны следующие подмножества планов:

$$H_1 = \{h_{11}, h_3, h_5, h_9\}; H_2 = \{h_1, h_3, h_5, h_{10}\};$$

$$H_3 = \{h_2, h_3, h_5, h_9\}, \dots, H_{20} = \{h_2, h_6, h_7, h_8, h_{10}\};$$

определяемые соответственно векторами $X_1, X_2, X_3, \dots, X_{20}$.

На втором этапе решения задачи анализируются отношения (типа γ_p - "состоять из ..."), существующие между элементарными планами и операциями (рис. 3.2).

Введем булевый вектор $\Lambda = (\lambda_1, \lambda_2, \dots, \lambda_e, \dots, \lambda_N)$ такой, что

$$\lambda_e = \begin{cases} 1, & \text{если операция } C_e \text{ входит в какой-либо из} \\ & \text{планов рассматриваемого покрытия } H_p, p = \overline{1, P}, \\ 0 & \text{в противном случае.} \end{cases} \quad (3.15)$$

Очевидно, каждому покрывающему вектору X_p можно поставить в соответствие вектор Λ_p , число единиц в котором определяет вес данного покрытия. Отсюда вытекает процедура для вычисления составляющей λ_e вектора Λ_p , сводимая к поразрядной конъюнкции вектора X_p с вектор-строкой z_e матрицы $\|z_{ej}\|$:

$$\lambda_e = \bigvee_{j=1}^M (x_j \wedge z_{ej}) = X_p \wedge z_e. \quad (3.16)$$

В рассматриваемом примере матрица $\|z_{ej}\|$ имеет вид:

$$\|z_{ej}\| = \begin{array}{cccccccccccc|l} & h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 & h_9 & h_{10} & \\ \cdot & \text{I} & 0 & \text{I} & 0 & 0 & \text{I} & 0 & 0 & \text{I} & 0 & C_1 \\ \cdot & 0 & \text{I} & 0 & \text{I} & \text{I} & 0 & 0 & 0 & 0 & 0 & C_2 \\ \cdot & \text{I} & \text{I} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_3 \\ \cdot & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \text{I} & C_4 \\ \cdot & 0 & 0 & \text{I} & 0 & 0 & 0 & \text{I} & 0 & 0 & 0 & C_5 \\ \cdot & 0 & 0 & 0 & \text{I} & 0 & 0 & 0 & \text{I} & 0 & 0 & C_6 \end{array}$$

Отобразим производимые действия такой последовательно-стью шагов:

1. Выбрать ℓ -ю вектор-строку матрицы $\|z_{ej}\|$.
2. Вычислить значение λ_e вектора Λ_p согласно (4.16).
3. Проверить $\ell = N$. Если условие выполняется, то вычислить сумму $\sum_{e=1}^N \lambda_e$; иначе перейти к п. 1.

Произведя подобные действия со всеми покрытиями, получим множество Ω_1 , состоящее из 5 оптимальных по критерию W_1 подмножеств планов $H_1 = \{h_1, h_3, h_5, h_9\}$, $H_2 = \{h_2, h_3, h_5, h_9\}$, $H_3 = \{h_1, h_3, h_8, h_9\}$, $H_{13} = \{h_1, h_5, h_7, h_9\}$, $H_{15} = \{h_2, h_5, h_7, h_9\}$, каждый из которых предопределяет синтез логической схемы процесса обработки заданного фрагмента детали на основе использования минимального числа операций.

Иная блок-схема решения задачи приведена на рис. 3.3.

Она включает также заключительный этап, связанный с описанием комплексных планов $H_p \in \Omega$, обработки детали, например:

$$H_p = (d_1, \bar{z}_1, h_3) \wedge (d_2, \bar{z}_1, h_3) \wedge (d_3, \bar{z}_1, h_3) \wedge (d_4, \bar{z}_1, h_3) \wedge (d_5, \bar{z}_1, h_3).$$

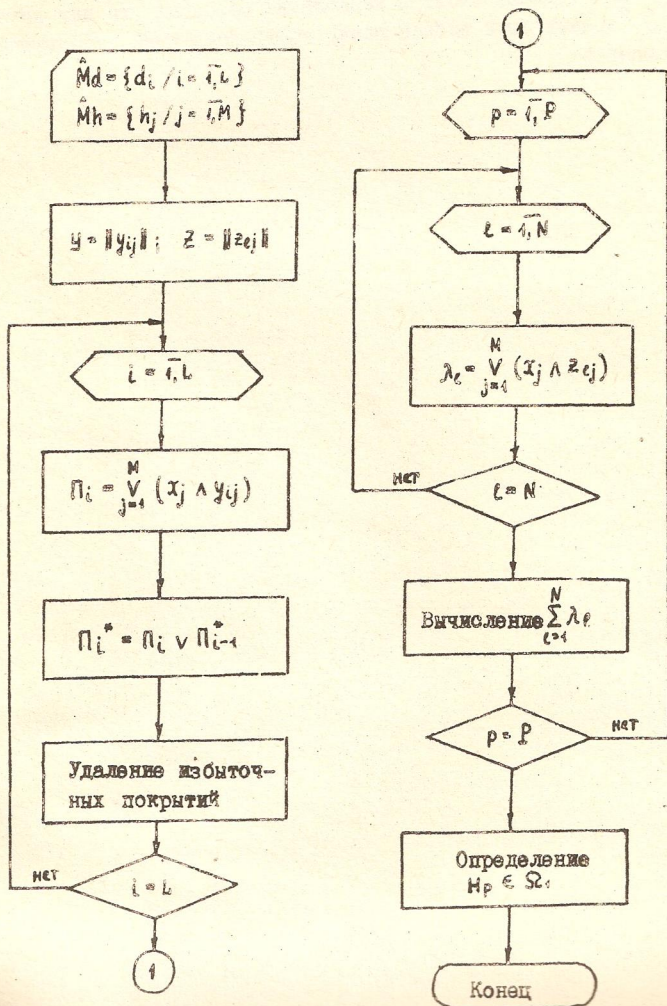


Рис. 3.3. Алгоритм решения задачи по критерию W , минимума числа операций*

С этой целью реализуется процедура $X_p \wedge y_i$ ($i=1, 2, \dots, 4$) поразрядной конъюнкции покрывающих векторов $X_p \in \mathcal{P}_1$, с каждой вектор-строкой матрицы $\|y_i\|$. Если окажется, что в конъюнкции $X_p \wedge y_i$ единицы стоят в нескольких разрядах, то для поверхности d_i должен быть выбран план, соответствующий наибольшему номеру разряда.