

МИНИСТЕРСТВО ЭНЕРГЕТИКИ И ЭЛЕКТРИФИКАЦИИ СССР

Главное производственно-техническое управление

по строительству

Всесоюзный институт по проектированию организации

энергетического строительства

"ОРГЭНЕРГОСТРОЙ"

Тема № 4631

План 1973 - 1974гг.

РАЗРАБОТКА И ПРИМЕНЕНИЕ МЕТОДОВ ПРОЕКТИРОВАНИЯ АСУ

Раздел А.

Применение методов машинного проектирования АСУ

К Н И Г А 3

Техническое задание

на разработку комплекса алгоритмов и программ для

автоматизированного проектирования

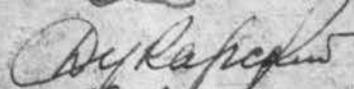
систем организационного управления

Зам. директора института  
начальник СКБ АСУ



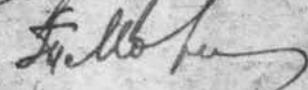
/В.А.Сердюков/

Начальник отдела НИС-7



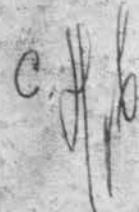
/О.М.Дукарский/

Начальник отдела АСУ-27



/Т.С.Монахова/

Руководитель темы  
главный специалист



/С.П.Никаноров/

Москва - 1974 г.

В настоящем отчете представлено задание на разработку алгоритмов и программы логической части автоматизированной системы проектирования систем организационного управления, выполненное по теме 463I плана ЦО 1973-1974 г.г.

Исполнитель: отделы АСУ-27, НИС-7

Руководитель темы № 463I: главный специалист Никаноров С.П.

Отчет разработан:

Главным специалистом Никаноровым С.П.

Старшим научным сотрудником, к.ф.м.н. Персицем Д.Б.

## А Н Н О Т А Ц И Я

В отчете представлено задание на разработку комплекса алгоритмов и программ логической части автоматизированной системы проектирования систем организационного управления. Задание предназначено для формирования алгоритмов и программ, обеспечивающих работу метода автоматизированного проектирования в его логической части в рамках основанной на нем системы автоматизированного проектирования. В состав определяемого в задании комплекса программ входят программы, обеспечивающие формирование теоретико-множественной модели (определения) системы управления, ее интерпретацию в терминах реальных объектов, и придание этой интерпретации формы проекта, а также внесение в проект изменений.

Вместе с Положением о методе автоматизированного проектирования настоящее задание может играть роль общей части технического задания на разработку системы автоматизированного проектирования.

## СО Д Е Р Ж А Н И Е

В в е д е н и е . . . . .	5
Раздел 1. Основные определения для описания процесса формирования проекта как интерпретированной теории . . . . .	10
1. Операционная схема . . . . .	10
1.1. Аргументы операций . . . . .	10
1.2. Граф операционной схемы . . . . .	11
1.3. Открытая операционная схема . . . . .	12
1.4. Замкнутая операционная схема . . . . .	13
2. T-интерпретации родов структур . . . . .	14
2.1. Операции сборки . . . . .	14
2.2. Операции квазисборки . . . . .	15
3. Термины и обозначения конститuent . . . . .	17
3.1. Конститuent . . . . .	17
3.2. Собственные обозначения конститuent . . . . .	18
3.3. Схемные обозначения конститuent . . . . .	20
3.4. Собственные термины конститuent . . . . .	22
3.5. Схемные термины конститuent . . . . .	22
3.6. Типы обозначений и терминов конститuent в схематическом представлении . . . . .	24
V 4. R-интерпретируемый род структуры . . . . .	26
Раздел II. Функциональная схема работы метода автоматизированного проектирования . . . . .	29
1. Общая схема работы метода . . . . .	29
2. Укрупненная блок-схема работы метода . . . . .	30
3. Функции проектировщика . . . . .	32
Раздел III. Состав комплекса программы . . . . .	33
Раздел IV. Задания на разработку алгоритмов и программы . . . . .	35
1. О диалоговом режиме использования комплекса программ . . . . .	35
2. Описание функций, выполняемых на ЭВМ . . . . .	35
3. Место алгоритмов, представленных в отчетах /3/ и /4/, в комплексе . . . . .	43
Литература . . . . .	44

## В В Е Д Е Н И Е

Настоящее задание является основанием для разработки комплекса алгоритмов и программы, обеспечивающих логическую часть автоматизированного проектирования систем организационного управления.

Задание не определяет алгоритмы во всех деталях, поскольку детализация задания может быть выполнена только в процессе дальнейшей разработки системы автоматизированного проектирования и комплекса программ для нее.

Настоящее задание на программирование вместе с Положением об автоматизированном проектировании может быть использовано как техническое задание на разработку системы автоматизации проектирования.

Для разработки более детального задания на разработку комплекса программ необходимо уточнение класса проектируемых систем и характеристик самой системы проектирования.

В состав задаваемого настоящим документом комплекса программ входят только программы, обеспечивающие формирование абстрактной модели /определения/ системы управления и ее интерпретацию в терминах реальных объектов /так называемую,  $R$ -интерпретацию/, а также придание этой  $R$ -интерпретации формы проекта и внесение в проект изменений.

Алгоритмы, выполняющие  $R$ -интерпретацию, заданы только в общем виде и при конкретизации класса моделей системы управления должны быть уточнены. Например, если система управления определяется как функциональная /т.е. с различием выполняемых ею функций и применяемых для их выполнения методов/, должны быть разработаны алгоритмы, обеспечивающие достаточно хороший выбор методов для выполнения функций управления.

С другой стороны, можно выполнение R - интерпретации в нетривиальных случаях (таких, как нахождение оптимальных решений) исключить из функций ЭВМ, предполагая включение соответствующих конститутант в список конститутант, "R - интерпретация которых не предусмотрена," тем самым представляя решения этих вопросов проектировщику. В этом смысле "объем" возможностей, предоставляемых методом, "линейно" зависит (растет) от "объема" включенных в его состав алгоритмов R - интерпретации.

Аналогично обстоит дело с большинством алгоритмов. Несмотря на то, что речь должна идти о комплексе взаимосвязанных программ, различные группы алгоритмов представляют различные, относительно независимые друг от друга возможности. Поэтому разработка программы <sup>допуска</sup> учитывать как взаимосвязанность программ, так и относительную независимость программ, обеспечивающую возможность различных вариантов работы метода от использования только алгоритмов, представленных в отчете [3], до полного комплекса, предусматриваемого настоящим заданием.

Как уже отмечалось, задание на разработку комплекса алгоритмов и программ дается с той степенью детальности, который соответствует общему уровню разработанности системы автоматизированного проектирования. Более точно, даются только постановки задач без указания входных и выходных форм и количественных характеристик исходных данных.

Контрольный пример должен, по-видимому, представлять собой целую систему тестов, чтобы иметь возможность проверять не только отдельные программы, но и их совместную работу.

Разработка этих элементов (формы, количественные характеристики и контрольный пример), должна быть осуществлена в опытно-поисковом порядке при дальнейших применениях метода в рамках построения

систем автоматизированного проектирования, включающих машинный синтез абстрактных моделей (определений) проектируемых систем.

Программирование комплекса представляет собой довольно трудоемкую и сложную задачу, требующую специальных средств. Представляется целесообразным использовать, с одной стороны, информационно поисковые системы, а с другой, — такие языки, как РЕБАЛ, PL - I, а также модульные системы, ориентированные на преобразования больших массивов.

Общая тенденция в дальнейшем совершенствовании метода состоит, в частности, в том, чтобы создать рамки для использования более или менее произвольных алгоритмических языков и систем программирования, особенно в части выполнения автоматической R-интерпретации абстрактных понятий.

Впрочем для выполнения этой функции могут быть использованы многие имеющиеся программы и комплексы программ: календарного планирования, линейного программирования, вообще, алгоритмы нахождения экстремумов, а также алгоритмы сортировок, выборов, упорядочения и переупорядочения массивов и т.п.

Имея в виду сложность и разветвленность программ, а также работу с большими массивами, предполагается вести программирование на таких ЭВМ, как ЭВМ системы "РЯД" (в частности, ЕС-1020) и М-32. Впрочем, выбор этих машин оправдывается, кроме того (а быть может и главным образом) имеющимися возможностями.

Настоящее задание отражает дальнейшее развитие метода автоматизированного проектирования, описанного в отчетах [1] - [4]. Изложенная в отчетах [1] - [3] <sup>проектирования</sup> схема автоматизированного  систем организационного управления получила здесь свое дальнейшее развитие. В частности более детально разработан процесс построения абстрактного определения проектируемой системы, что дало возможность установить более четкую границу и структуру

общелогической части метода, а также ее отношение к формированию заданий на разработку алгоритмов и программ, обеспечивающих создание проекта. Непосредственно, однако, знакомство с этим материалом не обязательно. Другой характер носит отношение настоящего задания к отчету [4] и к алгоритмам, представленным в отчете [3]. В отчете [4] изложена система операций над абстрактными определениями в виде родов структур и обобщенных родов структур, и реализующие их алгоритмы вместе с входными и выходными формами.

Место, которое занимают представленные в этих отчетах алгоритмы, подробно устанавливается в п. 3 раздела IV настоящего задания.

Содержательное описание операций над родами структур, как и определение самих родов структур, данные в отчете [4], можно рассматривать как составную часть раздела I п. I. настоящего задания.

Во всяком случае, предполагается знакомство с этим материалом. Необходимо также знакомство с понятием регулярного морфизма, определенного в п. 7 отчета [5] (ч. III).

Наконец, общая концепция и содержательное описание процесса проектирования, на который ориентировано настоящее задание, изложено в Положении о методе автоматизированного проектирования систем организационного управления.

Настоящее задание состоит из четырех разделов.

В разделе I, по-существу, излагается теория автоматизированного проектирования, вводящая все необходимые понятия на общепринятом уровне математической строгости, хотя несколько нестандартный характер логических конструкций заставил вести изложение более формально, чем это требуется существом вопроса.

В разделе II излагается в разных вариантах схема проектирования. Изложение ориентировано на формирование представления о работе метода в целом.

Раздел III содержит список функций, выполняемых на ЭВМ; имеется в виду, что такие функции однозначно соответствуют программам требуемого комплекса.

Раздел IV является основным, ибо в нем и изложено задание на программирование (в виде описания машинных функций).

Кроме того, в этом разделе, как уже указывалось, устанавливается связь с алгоритмами, содержащимися в отчетах [3] и [4], а также отмечаются характерные черты режима работы ЭВМ, квалифицирующие его как диалоговый.

# РАЗДЕЛ I

## ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ ДЛЯ ОПИСАНИЯ ПРОЦЕССА ФОРМИРОВАНИЯ ПРОЕКТА КАК ИНТЕРПРЕТИРОВАННОЙ ТЕОРИИ

### I. Операционная схема

#### I.1. Аргументы операций

Пусть  $OP$  - некоторое множество операций над родами структур и  $Op_i \in OP$ , тогда результат выполнения операции  $Op_i$  записывается в виде  $Op_i(x_1, \dots, x_k)$ , где аргументы  $x_k$  могут принимать значения родов структур, соотношений теории множеств (или целых текстов), наборов термов, отображений и т.д. Аргументы, значения которых - роды структур, называются основными, а остальные аргументы - дополнительными.

Число  $k$  для аргумента  $x_k$  называется номером аргумента. Термин "основной (или дополнительный) номер операции" означает номер основного (соответственно, дополнительного) аргумента операции. Таким образом, мы имеем отображение:

$$\rho: OP \rightarrow 2^{\mathbb{Z}^+} \times 2^{\mathbb{Z}^+},$$

где  $S_1(Op) = \rho\pi_1(\rho(Op))$  - множество основных номеров операции  $Op$ , а  $S_2(Op) = \rho\pi_2(\rho(Op))$  - дополнительных. Числа  $\alpha_1 = |S_1(Op)|$ ,  $\alpha_2 = |S_2(Op)|$  и  $\alpha(Op) = |S_1(Op)| + |S_2(Op)|$  называются, соответственно, основной, дополнительной и полной арностями операции.

Множество значений  $k$ -го аргумента операции  $Op$  будет обозначаться через  $Arg_k(Op)$ , а множество всевозможных наборов  $(x_1, \dots, x_n)$ , к которым применима операция  $Op$ , - через  $Arg(Op)$ .

## 1.2. Граф операционной схемы

Графом операционной схемы

$$G = G(D, V, \lambda, \alpha, \mu)$$

назовем пятерку  $\langle D, V, \lambda, \alpha, \mu \rangle$ ,  
удовлетворяющую следующим условиям:

- 1°.  $D$  - конечное множество, называемое множеством дуг;
- 2°.  $V$  - непустое конечное множество, называемое множеством вершин;
- 3°.  $\lambda: D \rightarrow V \times V$ .

Для любого  $v \in V$  положим:

$$D(v) = \text{Im } \lambda \cap (\{v\} \times V)$$

$$D_v = \lambda^{-1}(D(v)) \quad \text{- множество выходящих из вершины } v \text{ дуг;}$$

$$D^{-1}(v) = \text{Im } \lambda \cap (V \times \{v\})$$

$$D_v^{-1} = \lambda^{-1}(D^{-1}(v)) \quad \text{- множество входящих в вершину } v \text{ дуг;}$$

$$B = V \setminus \text{pr}_2(\text{Im } \lambda) \quad \text{- множество начальных вершин;}$$

$$E = V \setminus \text{pr}_1(\text{Im } \lambda) \quad \text{- множество конечных вершин}$$

- 4°.  $|E| = 1$  Через  $e$  обозначим конечную вершину, так что  $E = \{e\}$

- 5°.  $e \in \alpha \subset V$  ( $\alpha$  - множество основных вершин).

Положим  $u_c = V \setminus \alpha$  (множество дополнительных вершин)

- 6°.  $u_c \subset B$ ,  $B \setminus u_c \neq \emptyset$  и  $\lambda|_{D \cap (u_c \times \bar{V})}$  инъективно)

- 7°.  $\mu: D \rightarrow \mathbb{Z}^+$  - упорядочение дуг.

Для любого  $v \in V$  положим  $\mu^v = \mu|_{\mathcal{D}_v^{-1}}$   
 $8^\circ \mu^v: \mathcal{D}_v^{-1} \rightarrow \mathbb{Z}^+_{|\mathcal{D}_v^{-1}|}$  - биекция.  $9^\circ$  Ориентированный граф  $\langle V, \text{Im } \lambda \rangle$  ( $V$  - множество вершин,  $\text{Im } \lambda \subset V \times V$  - множество дуг графа) не имеет ориентированных циклов, или эквивалентно, не существует последовательностей  $d_1, \dots, d_n \in \mathcal{D}$  таких, что  $\rho\pi_2(\lambda(d_i)) = \rho\pi_1(\lambda(d_{i+1}))$  ( $i=1, \dots, n-1$ ) и  $\rho\pi_2(\lambda(d_n)) = \rho\pi_1(\lambda(d_1))$ .

### 1.3. Открытая операционная схема

Открытой операционной схемой  $F = F(G, OP, \nu)$  называется тройка  $\langle G, OP, \nu \rangle$ , удовлетворяющая следующим условиям:

- 1 $^\circ$   $G = G(\mathcal{D}, V, \lambda, \mathcal{U}, \mu)$  - граф операционной схемы;
- 2 $^\circ$   $OP$  - некоторое множество операций над родами структур;
- 3 $^\circ$   $\nu: \mathcal{U} \setminus \mathcal{B} \rightarrow OP$  (операционное заполнение графа);
- 4 $^\circ$  Для любого  $v \in \mathcal{U} \setminus \mathcal{B}$  имеет место равенства

множество:

$$\mu(\mathcal{D}_v^{-1} \cap \lambda^{-1}(\mathcal{U} \times \{v\})) = \mathcal{S}_1 \circ \nu(v)$$

$$\mu(\mathcal{D}_v^{-1} \cap \lambda^{-1}(\mathcal{U}_c \times \{v\})) = \mathcal{S}_2 \circ \nu(v)$$

или, другими словами,

$$\mathcal{S} \circ \nu(v) = \langle \mu(\mathcal{D}_v^{-1} \cap \lambda^{-1}(\mathcal{U} \times \{v\})), \mu(\mathcal{D}_v^{-1} \cap \lambda^{-1}(\mathcal{U}_c \times \{v\})) \rangle.$$

Это означает, что основные вершины соответствуют основным аргументам, а дополнительные - дополнительным. При этом отображение  $\mu$  определяет соответствие между дугами и множествами  $\text{Alg}_k(OP)$  соответствующих операций  $OP$ .

### Г.4. Замкнутая операционная схема

Замкнутой операционной схемой  $C = C(F, \xi, \nabla)$  называется тройка  $(F, \xi, \nabla)$ , удовлетворяющая следующим условиям:

1°  $F$  - открытая операционная схема;

2°  $\xi: V \rightarrow \text{Str} U \left( \bigcup_{\substack{Op \in \mathcal{M} \cup \\ K \in \mathcal{Z}^+(Op)}} \text{Arg}_K(Op) \right)$   
 (полное замыкание операционной схемы,  
 $\text{Str}$  - класс всех рядов структур);

3° Для любого  $v \in V \setminus B (= U \setminus B)$  имеет место соотношение:  $\xi(v) = Op(x_1, \dots, x_n)$ , где  $Op = \nu(v)$ ,

$$n = |\mathcal{D}_v^{-1}|, \quad x_k = [\xi \circ \rho_k \circ \lambda \circ (\mu^v)^{-1}](k), \quad k = 1, \dots, n$$

4°  $\nabla \subset \text{Str}$  (множество базовых родов структур)

5°  $\xi(U \cap B) \subset \nabla$ .

Отображение  $\tilde{\xi} = \xi|_B$  называется базовым замыканием операционной схемы. Очевидно, полное замыкание восстанавливается по базовому однозначно, т.е. если  $C_i = C(F, \xi_i, \nabla)$  ( $i = 1, 2$ )

и  $\tilde{\xi}_1 = \tilde{\xi}_2$ , то  $\xi_1 = \xi_2$  (и, следовательно,  $C_1 = C_2$ ).

## 2. T - интерпретация родов структур

### 2.1. Операции сборки

Операция  $Op(x_1, \dots, x_n)$  над родами структур называется операцией сборки, если для каждого набора значений основных аргументов  $\Sigma_{i_1}, \dots, \Sigma_{i_r}$  из области определения  $Op$  существуют регулярные морфизмы (в смысле [5], гл. III, п. 3):

$$\Sigma \longrightarrow \Sigma_{i_k} \quad (k = 1, \dots, r), \quad \text{где}$$
$$\Sigma = Op(x_1, \dots, x_n), \quad x_{i_k} = \Sigma_{i_k}.$$

Заметим, что регулярный морфизм  $\Sigma_1 \longrightarrow \Sigma_2$  индуцирует естественным образом T - интерпретацию  $\Sigma_2$  в теории рода  $\Sigma_1$  (интерпретацию одной теории в другой мы называем T - интерпретацией). Следующие операции и только они являются операциями сборки (из числа определенных выше <sup>в п. 1</sup> кроме дублирования):

- $Op_I$  - свободное произведение;
- $Op_2$  - прямое произведение;
- $Op_3$  - усиление;
- $Op_5$  - простое расширение базы;
- $Op_7$  - общее расширение базы;
- $Op_9$  - смешанное произведение;
- $Op_{II}$  - конкретизирующее вложение;
- $Op_{I3}$  - расширение родовой структуры.

Операция  $Op_{II}^f$  (конкретизирующее терм-вложение) также индуцирует канонические T - интерпретации основных аргументов в результат операции.

## 2.2. Операция квази-сборки

Операцию усиления  $Op_2$  можно переопределить следующим образом.

Операция 3'. Родовое усиление. Пусть  $\Sigma_1$  - род структуры (с.н. [6], стр. 249) и  $\langle T_1, \dots, T_{2n} \rangle$  - картаж его внутренних термов. Пусть, далее,  $\Sigma_2$  - другой род структуры и  $\rho(\Sigma_2) = \nu$ . Тогда определим  $\Sigma = Op'_3(\Sigma_1, \Sigma_2; \langle T_1, \dots, T_{2n} \rangle)$ ,

положив:

1.  $B_1(\Sigma) = B_1(\Sigma_1)$ ;
2.  $B_2(\Sigma) = B_2(\Sigma_1)$ ;
3.  $G(\Sigma) = G(\Sigma_1)$ ;
4.  $\gamma(\Sigma) = \gamma(\Sigma_1)$ ;
5.  $S(\Sigma) = S(\Sigma_1)$ ;
6.  $R(\Sigma) \{ X_1, \dots, X_n, A_1, \dots, A_m, s \} \equiv R(\Sigma_1) \{ X_1, \dots, X_n, A_1, \dots, A_m, s \} \wedge R(\Sigma_2) \{ T_{i_1}, \dots, T_{i_z}, T_{2n} \}$ ,

Как и в случае операции  $Op'_{11}$ , операция  $Op'_3$  - не является операцией сборки, но индуцирует T-интерпретации аргументов  $\Sigma_1$  и  $\Sigma_2$  в  $\Sigma$ .

Наконец, в случае операции  $Op'_{15}$  (взятие булеана), которая также не является операцией сборки, мы имеем "семейство T-интерпретаций"; именно любой элемент  $\mathfrak{J} \in \mathfrak{J}'$  родовой структуры рода  $\Sigma = Op'_{15}(\Sigma_1, \mathfrak{J}')$  дает T-интерпретацию  $\Sigma_1$  в  $\Sigma$ . Это есть случай переменной интерпретации, определению которой посвящено следующее.

Определение. Пусть  $\Sigma_1$  и  $\Sigma_2$  — два рода структур  
и  $T_1, \dots, T_{P(\Sigma_2)+1}$  внутренние термины рода  $\Sigma_1$ , причем,  
 $T_i \in \bar{G}(B[S])$ , где  $S \in \Phi^{P(\Sigma_1)}$  некоторая схема кон-  
струкции ступени,  $G = G(\Sigma_1)$ . Будем говорить, что термины  
 $T_1, \dots, T_{P(\Sigma_2)}$  являются переменной интерпретацией  $\Sigma_2$   
в  $\Sigma_1$ , если в  $\mathcal{T}_{\Sigma_1}$  имеет место теорема:

$$\forall x_1 \in T_1, \dots, \forall x_{P(\Sigma_2)+1} \in T_{P(\Sigma_2)+1} : R(\Sigma_2) \stackrel{?}{=} x_1, \dots, x_{P(\Sigma_2)+1} \stackrel{?}{=} \Lambda$$

$$\wedge x_{P(\Sigma_2)+1} \in \bar{G}_{x_1, \dots, x_{P(\Sigma_2)}}(S(\Sigma_2)), \quad \text{где}$$

$$\sigma_{x_1, \dots, x_{P(\Sigma_2)}} : \text{дик}_k \mapsto x_k, \text{дик}_k = G(\Sigma_2)^{-1}(z_k), z_k \in B_3(\Sigma_2)$$

Операции  $Op_{11}$ ,  $Op_3$  и  $Op_{15}$  называются  
операциями квазисборки.

### 3. Термины и обозначения конституэнт

#### 3.1. Конституэнты

Конституэнтами рода структуры называются:

1. Род структуры;
2. Основные базисные множества;
3. Вспомогательные базисные множества;
4. Родовая структура;
5. Схема конструкции ступени;
6. Упорядочение базы;
7. Типизация;
8. Внутренние термины;
9. Аксиомы;
10. Теоремы.

Замечание. В п. 8 имеются виду внутренние термины, не включенные в п. 2,3,4. Определение внутренних терминов см. в [6] стр. 243.

Конституэнты 2,3,4,7 - называются внутренними, а остальные (1,5,6) - внешними;

конституэнта 3 называется постоянной, а остальные - переменными;

конституэнты 1+6,9 называются определяющими или исходными, а остальные - производными или выводными.

Конституэнты 1,2,3,4,8 называются понятиями (абстрактными) рода, при этом 1 - фундаментальным понятием.

Таким образом, 2,3,4,8 - внутренние понятия; 2,4,8 - внутренние переменные понятия.

### 3.2. Собственные обозначения конститuent

Каждому типу конститuent присваиваются следующие "стандартные буквы":

1. Род структуры -  $\Phi$  (фундаментальное понятие);
2. Основные базисные множества -  $X$  (от латинского, "икс");
3. Вспомогательные базисные множества -  $C$  (от "constant");
4. Родовая структура -  $P$  (по первой букве);
5. Схема конструкции ступени -  $K$ ;
6. Упорядочение базы -  $U$ ;
7. Типизация -  $B$  - (без мотивировки);
8. Внутренний терм -  $\Pi$  (понятие);
9. Аксиома -  $A$ ;
10. Теорема -  $T$ .

Для дальнейшего фиксируется замкнутая операционная схема  $C(F, \xi, \nabla)$ , причем множества  $V$  и  $\nabla$  предполагаются конечными и упорядоченными. Таким образом, каждый базовый род структуры  $\Sigma \in \nabla$  имеет в  $\nabla$  свой порядковый номер, который называется базовым индексом рода структуры; при этом символ  $\phi_{(i)}$  называется базовым обозначением рода структуры. Далее, каждая вершина  $v \in V$  также имеет свой порядковый номер; который для основной вершины называется родовым индексом; символ  $\phi_j$  называется родовым обозначением рода структуры  $\xi(v)$ , а символ  $\phi_{(i)j}$  - базовым родовым обозначением этого рода; соответственно, символ  $(i)j$  - базовым родовым индексом.

Внутренний индекс конститuent данного рода - это номер по порядку формирования (или написания) конститuent в пределах данного рода-свой для каждого типа (при этом весь текст должен удовлетворять определению формативной конструкции или его подходящего эквивалента; собственно, здесь только важно, чтобы производные конститuent следовали после тех конститuent,

через которые они определяются.

Внутренний индекс, таким образом, может быть только при буквах: X, C, П, А, Т (написание  $X_i$  или  $X.i$ ).

Полный индекс конституэнта, не имеющего внутреннего индекса, совпадает с родовым индексом, а для остальных конституэнт имеет вид: " $i!j$ ", где  $i$  - родовый индекс,  $j$  - внутренний индекс.

Базовый полный индекс - это индекс вида " $(K)i!j$ " или " $(K)i$ " в случае базового рода структуры.

Соответственно, полные базовые и полные внутренние обозначения имеют вид  $G_\alpha$ , где  $G$  - стандартная буква конституэнта, а  $\alpha$  - соответственно ее полный, базовый полный, или внутренний индекс. Если конституэнта не имеет внутреннего индекса, то ее внутреннее обозначение совпадает со стандартной буквой.

Таким образом, при выполнении операций над родами  $\xi$  структур обозначения конституэнт аргументов не наследуются (как это было в отчете [4] и как это делает обычно), хотя, конечно, сами конституэнты аргументов определяют результаты операции и, следовательно, обозначения конституэнт последнего. Поэтому если мы хотим сохранить связь конституэнт аргументов и результата, то это необходимо обеспечить специальным образом, этому вопросу посвящен следующий пункт.

Все введенные в этом пункте "обозначения" и "индексы" называются сквозными или собственными, а обозначения конституэнт - собственными стандартными.

Точно также, как это делается для терминов в п. 3,4, могут быть определены так называемые собственные индивидуальные обозначения, которые являются заменами введенных собственных стандартных обозначений для тех или иных конституэнт на произвольные.

### 3.3. Схемные обозначения конститuent

В настоящем определении вводятся, так называемые схемные (или операционные) обозначения, призванные осуществить связь конститuent различных родов структур, входящих в операционную схему. Таким образом, предполагается, что задана произвольно фиксированная замкнутая операционная схема и ее вершины перенумерованы правильно, т.е. если есть путь из  $v_1$  в  $v_2$ , то  $n(v_1) < n(v_2)$ .

Каждый путь  $t$  графа схемы, начинающийся в основной вершине  $v_1$  и кончающийся в основной вершине  $v_2$ , индуцирует каноническую T-интерпретацию (постоянную или переменную) теории рода  $\xi(v_1)$  в теории рода  $\xi(v_2)$ . Эту интерпретацию мы обозначим через  $J(t)$ . Путь  $t$  можно закодировать последовательностью номеров вершин  $n(v_i)$  и номеров дуг  $\mu(d_j)$ , что будет обозначаться следующим образом:

$$K(t) = n_0 : \mu_1 \cdot n_1 : \mu_2 \cdot n_2 : \dots : \mu_n \cdot n_n.$$

Это выражение назовем кодом интерпретации. Если звено  $\mu_i \cdot n_i$  дает переменную интерпретацию, то он заменяется на  $\mu_i \cdot n_i \cdot x$ . Через  $\varphi_\alpha$ , где  $\alpha$  - код T-интерпретации, будем обозначать образ теории рода  $\xi(v_1)$  в теории рода  $\xi(v_2)$ , т.е. множество тех внутренних конститuent рода  $\xi(v_2)$ , которые перенесены без изменений (с точностью до обозначений) в результате выполнения операций из рода  $\xi(v_1)$ . Относительно каждой операции этому понятию можно дать точное определение. Далее, если  $i$  - внутренний индекс конститuentи рода  $\xi(v_1)$ ,  $G$  - ее стандартная буква,  $\alpha = K(t)$ -код T-интерпретации, то  $G_{\alpha|i}$  - схемное обозначение конститuent, а  $\alpha|i$  - схемный индекс. Для конститuent, не имеющих внутреннего индекса, схемный индекс совпадает с кодом T-интерпретации и, следовательно, схемное обозначение имеет вид  $G_\alpha$ .

Главной интерпретацией рода  $\xi(v_1)$  называется интерпретация  $J(t)$ , где  $v_1$  - начало пути  $t$ , а  $e$  - его конец ( $e$  - конечная вершина графа схемы).

Естественно считать, что  $G_\alpha, G_{\alpha|i} \in \Phi_\alpha$ , так что если  $G_{n\alpha|i} = [J(t)](G_{n\alpha|i})$  - образ конституэнты  $G_{n\alpha|i}$  при T-интерпретации  $J(t)$ , то можно считать  $G_{\alpha|i} = G_{n\alpha|i}$ , где  $\alpha|i$  - схемный индекс конституэнты  $G_{n\alpha|i}$ .

Пусть  $G_{mi}$  - полное (сквозное) обозначение конституэнты рода  $\Phi_m$ ,  $\mathcal{J}'$  - множество путей с началом в начальных основных вершинах графа. Схемы и с концом - в вершине  $\bar{v}$ , где  $n(\bar{v}) = m$ ,  $\mathcal{J}''$  - множество путей с началом в вершине  $\bar{v}$  и с концом в конечной вершине  $e$ . Положим

$$\bar{J}(G_{mi}) = \{G_{e|j} \mid \exists t \in \mathcal{J}': [J(t)](G_{e|j}) = G_{mi} \text{ (начало пути } t)\};$$

$$\bar{\bar{J}}(G_{mi}) = \{G_{n|i} \mid \exists t \in \mathcal{J}'': [J(t)](G_{mi}) = G_{n|i} \text{ (} n = n(e)\text{)}\}.$$

Элементы множества  $\bar{\bar{J}}(G_{mi})$  называются главными T-интерпретациями конституэнты, а элементы множества  $\bar{J}(G_{mi})$  ее главными <sup>инт</sup>антиинтерпретациями.

Можно определить, наконец, понятие "сокращенные обозначения" и "сжатые обозначения". Но такое определение совершенно аналогично соответствующим конструкциям, вводимым в следующих пунктах 3, 4 для терминов.

### 3.4. Собственные термины конститuent

В этом пункте вводятся собственные термины конститuent в отличие от схемных терминов, вводимых в следующем пункте, аналогично тому, как это было сделано выше для обозначений.

Каждый конститuentе  $G_{m|i}$  рода структуры  $\Phi_i$  сопоставляются стандартные внутренний и полный термины. Внутренний термин есть сочетание внутреннего индекса (если таковой существует) и названия типа конститuenta (например, пятая теорема, первое основное базисное множество, схема конструкции ступени и т.д.). Полный термин  $Term(G)$  это сочетание внутреннего термина  $term(G)$  и внутреннего термина  $term(\Phi_m)$  в родительном падеже; разумеется,  $Term(\Phi_m)$  совпадает с  $term(\Phi_m)$  (например, первое основное базисное множество седьмого рода структуры).

Далее, конститuentам типов  $\Phi, X, \mathbb{E}, P, \Pi$  могут присваиваться индивидуальные внутренние термины  $ind(G)$ . Тогда полный индивидуальный термин  $Ind(G)$  есть сочетание внутренних индивидуальных терминов  $ind(G)$  и  $ind(\Phi_m)$ ; впрочем, один из них за исключением индивидуального, может быть и стандартным.

### 3.5. Схемные термины конститuent

В настоящем определении вводится "схемная" терминология. Она индуцирована собственной терминологией и схемным индексом. Полный схемный (стандартный  $STerm(\Phi_m, t)$  или индивидуальный  $SInd(\Phi_m, t)$ ) термин рода структуры  $\Phi_m = \xi(v)$  относительно  $T$ -интерпретации  $J(t)$  (или относительно пути  $t$ , где  $v$  - начало пути) является сочетанием собственных соответственно, стандартных или индивидуальных терминов родов структур, отвечающих вершинам дуг, образующих путь  $t$ . причем начиная со второго все термины ставятся в родительном падеже.

Каждый полный схемный термин рода структуры (относительно  $J(t)$ ) можно заменить сокращенным, который обозначается через  $\overline{STerm}(\varphi_m, t)$ . Пусть для некоторых пар  $(\varphi_m, t)$  введены сокращенные термины. Тогда скатым термином  $CSTerm(\varphi_m, t)$  называется сочетание, полученное заменой "отрезков" полного схемного термина их сокращениями. При этом отрезки пути, соответствующие заменяемым отрезкам терминов, не должны иметь общие "дуги и замена имеется ввиду максимальной, т.е. "заменяемый путь" нельзя расширить. Ясно, что данным множеством  $\overline{ST}$  сокращений вида  $\overline{STerm}(\varphi_m, t)$  и парой  $(\varphi_{m_0}, t_0)$  скатый термин  $CSTerm(\varphi_m, t)$ , вообще говоря, не определяется однозначно. Однако, если заменяемые отрезки множества не имеют общих дуг, то однозначность имеет место.

Формирование схемной терминологии для остальных конститuent осуществляется аналогичным образом. Именно, полный схемный (стандартный или индивидуальный) термин конститuent  $G_{mli}$  (относительно  $t$ ) образуется сочетанием собственного внутреннего термина и поставленного в родительном падеже соответствующего термина рода структуры.

Скатые термины  $CSTerm(G_{mli}, t)$  конститuent  $G_{mli}$  образуются как за счет скатых терминов родов структур  $CSTerm(\varphi_m, t)$ , так и за счет вводимых сокращений для некоторых полных схемных терминов самих конститuent. При этом однозначность обеспечивается условием неналожимости сокращаемых, отрезков во всем множестве сокращений (и для родов структур, и для отдельных конститuent, "встречающихся на пути").

Точно также, как это делается для терминов, вводятся понятия скатых и сокращенных схемных обозначений (см. п. 3.3).

### 3.6. Типы обозначений и терминов конститuent в схематическом представлении

В предыдущем пункте введены восемь типов терминов и обозначений для конститuent. Названия этих типов образуются с помощью следующей таблицы:

Признаки	Схемность	Стандартность	Полнота
Значения признаков	Собственный	Стандартный	Полный
	Схемный	Индивидуальный	Сжатый (внутренний)

Типы обозначений или терминов характеризуются тремя признаками (схемность, стандартность, полнота), каждый из которых принимает два значения. Название типа образуется из названий определяющих его значений признаков. При этом со словом "собственный" должно употребляться слово "внутренний" (но не "сжатый"), а со словом "схемный" - слово "сжатый". Понятие "сокращенный" можно рассматривать как частный случай понятия "сжатый".

Заметим, что изложенное представление типов обозначений и терминов конститuent призвано играть роль некоторого методического приема, не более.

Ниже представлено распределение определений этих типов по пунктам:

- а) собственные стандартные обозначения - п. 3.2;

- б) собственные индивидуальные обозначения - нет (ссылка в п. 3.2 на аналогично с п. 3.4);
- в) схемные стандартные полные обозначения - п. 3.3;
- г) схемные краткие (сокращенные) обозначения - нет (ссылка в п. 3.3. на п. 3.5);
- д) схемные индивидуальные полные обозначения - нет (определяются естественным образом, аналогично определению терминов, того же типа в п. 3.5);
- е) собственные термины - п. 3.4;
- ж) схемные термины - п. 3.5.

#### 4. R-интерпретируемый род структуры

R-интерпретируемый род структуры называется род структуры вместе со следующими данными:

1. Частичное отображение  $Alg(\Sigma)$  множества внутренних термов в множество алгоритмов.

При этом алгоритм формирует R-интерпретацию соответствующего внутреннего термина по R-интерпретации тех внутренних термов, через которые он определяется или выражается; таким образом, более точно, алгоритм сопоставляется выражению вида:

$$\Pi_i = L,$$

где  $L$  - терм, содержащий в качестве свободных переменных внутренние термы, интерпретация, которых определяет R-интерпретацию термина  $\Pi_i$ .

2. Подмножество  $RInt(\Sigma)$  множества конститuent данного рода структуры, которое называется абстрактным содержанием проекта, причем внутренние термы из множества  $RInt$  даны вместе с указанными выше выражениями вида:

$$\Pi_i = L.$$

3. Подмножество  $SInt$  множества всех конститuent данного рода  $\Sigma$ , элементы которого называются конститuentами, предназначенными для специальной R-интерпретации.

4. Для каждой конститuent  $G \in SInt$  - подмножество множества свободных переменных (внутренних термов), входящих в определяющее конститuentу  $G$  выражение (разметка формулы или термина). Термы этого подмножества называются разрешающими. (Имеется в виду что R-интерпретация конститuent из  $SInt$  может

(и должна) быть выполнена проектировщиком, с помощью R-интерпретации только разрешающих термов).

Пусть дан  $\mathcal{R}$ -интерпретируемый род структур. Определим релевантное множество конститuent  $Rel$  индуктивным образом:

1.  $RInt \subset Rel$  ;
2. Если  $G \in Rel \cap SInt$  и  $G'$  - свободная переменная входящая в определяющее  $G$  соотношение, причем  $G'$  - разрешающий терм, то  $G' \in Rel$ . ;
3. Если  $G \in Rel \setminus SInt$  и  $G'$  - свободная переменная входящая в определяющее  $G$  соотношение, то  $G' \in Rel$  ;
4. Других элементов, кроме определенных в п.п. 1,2,3, множество  $Rel$  не содержит.

В дальнейшем будет предполагаться, что ориентированный граф на множестве  $Rel$ , определенный ниже, не имеет циклов и петель. Дуга графа  $\Gamma$  по определению соединяет конститuentу  $G_1$  с конститuentой  $G_2$ , если либо  $G_1 \in SInt$  и тогда  $G_2$  - разрешающий терм для  $G_1$  (т.е. для выражения, определяющего  $G_1$ ), либо  $G_1 \notin SInt$  и тогда  $G_2$  - какая-либо из свободных переменных в выражении, определяющим конститuentу  $G_1$ .

Пусть  $Z \subset Rel$ . Определим индуктивным образом множество  $Aut(Z)$  элементы которого называются автоматически R-интерпретируемыми конститuentами на базе Z.

1. Если  $G \in (Rel \cap D(Alg) \setminus SInt)$  ( $D(Alg)$  - область определения отображения  $Alg$ ) и каждая из свободных переменных выражений, определяющего  $G$ , либо принадлежит  $Z$  либо принадлежит  $Aut(Z)$ , то  $G \in Aut(Z)$ .

2. Других элементов, кроме определенных в п. 1 множество не содержит.

Определим теперь по индукции понятие общей конституэнты  $n$ -ой очереди.

1. Общая конституэнта 1-ой очереди (или исходная конституэнта) - это конституэнта из  $Rel$ , которая либо является базисным множеством, либо родовой структурой, либо принадлежит множеству  $Synt$  и в этом случае не имеет разрешающих термов.

2. Общая конституэнта  $n$ -ой очереди - это конституэнта из множества  $Rel \setminus Aut(Z)$  (где  $Z$  - множество конституэнт 1-ой, ...,  $n-1$ -ой очередей), которая либо принадлежит  $Synt$  и в этом случае ее разрешающие термы принадлежат множеству  $Aut(Z) \cup Z$ , либо не принадлежат  $Synt$  и в этом случае все свободные переменные определяющего ее выражения принадлежат множеству  $Aut(Z) \cup Z$ .

Общие конституэнты  $n$ -ой очереди ( $n > 1$ ), не принадлежащие множеству  $Synt$ , называются конституэнтами, R-интерпретация для которых не предусмотрена (или предзаданиями на программирование или просто задачами). Оставшиеся конституэнты (т.е. принадлежащие  $Synt$  и все общие конституэнты 1-ой очереди) называются конституэнтами  $n$ -ой очереди.

Раздел II. Функциональная схема работы метода  
автоматизированного проектирования.

I. Общая схема работы метода.

Весь процесс проектирования, использующий данный метод, может быть разбит на три этапа:

1. Формирование абстрактного определения проектируемой системы  $\{K$  - интерпретируемо  $\}$ , т.е. главного рода структуры/;

2. Интерпретация этого определения в терминах реальных объектов /т.е.  $R$ -интерпретации главного рода структуры/;

3. Придание интерпретированному определению формы проекта.

На каждом из этих этапов происходит чередование операций, выполняемых на ЭВМ, и операций, выполняемых проектировщиком.

В приведенную ниже в п.2 укрупненную блок-схему включены только те функции (как для ЭВМ, так и для проектировщика), называемые основными, выполнение которых необходимо для получения проекта.

Кроме таких функций методом предусмотрено выполнение еще двух типов функций: вспомогательных и внесения изменений. Вспомогательные функции призваны обеспечить реальную выполнимость основных функций проектировщика. Конечно, проектировщиком непосредственно используются из вспомогательных только машинные операции, но выполнение последних, в свою очередь, требует выполнения определенных вспомогательных функций проектировщиком:

Функции внесения изменений позволяют перепроектировать только ту часть проекта, которая требует изменений, а не повторять на ЭВМ весь процесс с начала до конца.

Вспомогательные функции не включены в блок-схему ввиду неопределенности порядка их следования, а функции изменения ввиду их простоты (на данном уровне разработанности) и изолированности.

Наконец, заметим, что рассматриваемая схема носит иллюстративный характер и предназначена для обеспечения восприятия всего комплекса программы в целом и определения его места во всей функциональной структуре работы метода.

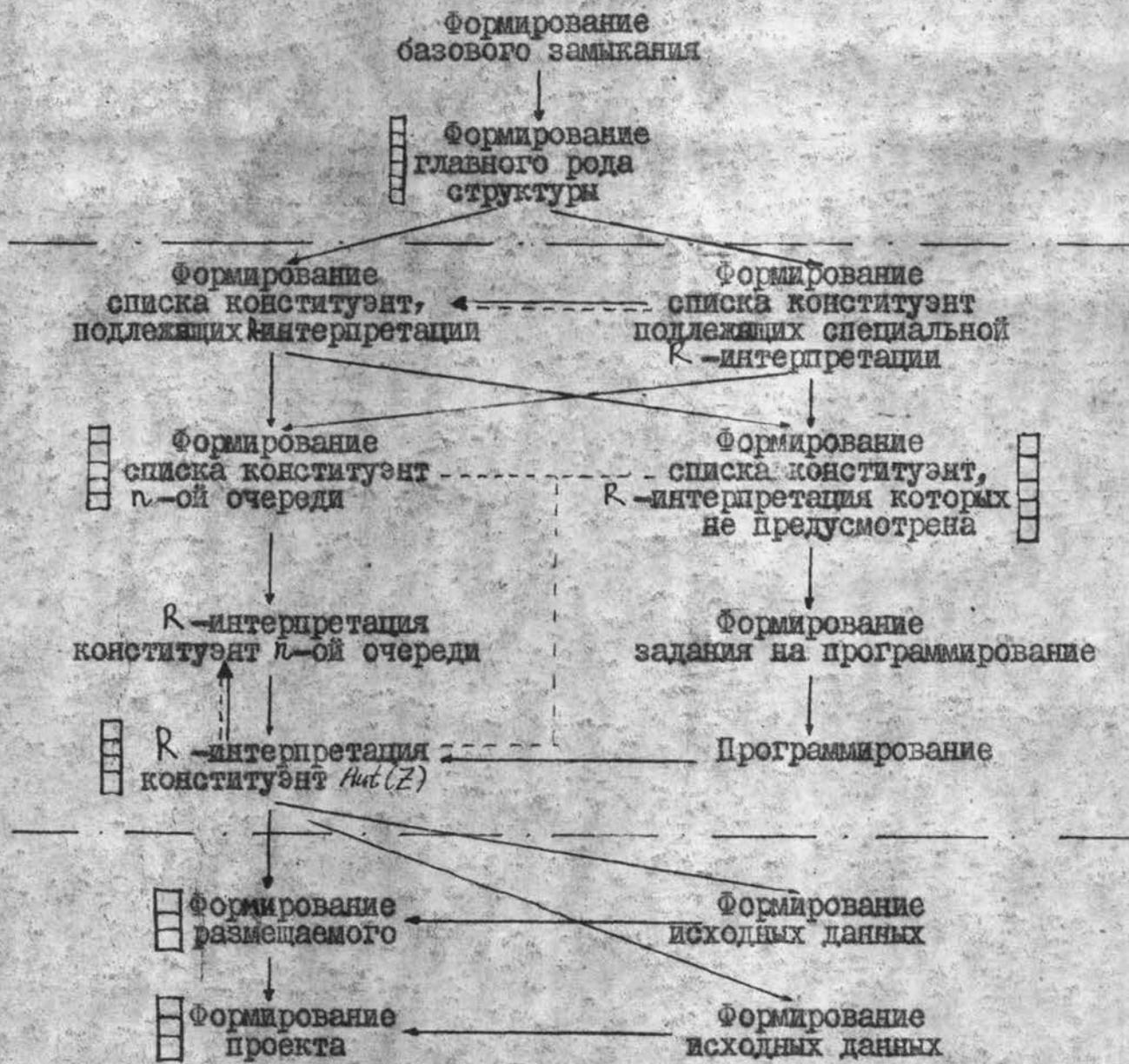
## 2. Укрупненная блок-схема работы метода

В приводимой ниже схеме приняты следующие условные обозначения:

-  - функция ЭВМ;
-  - функция проектировщика;
-  - соединение операций, которые выполняются одновременно (параллельно);
-  - возможная последовательность операций;
-  - основная последовательность операций;
-  - границы указанных выше в п. I трех этапов.

## 3. Функции проектировщика

Приводимые ниже формулировки функций проектировщика предназначены для более полной ориентации программиста в вопросах назначения функций ЭВМ и реализующих их алгоритмов и программ, а так же для формирования входных и выходных форм; знаком отмечены вспомогательные функции.



Укрупненная блок-схема работы метода

- Ф.1. Формирование базового замыкания операционной схемы, главный род структуры которой есть абстрактное определение проектируемой системы.
- Ф.2.<sup>н</sup>) Оснащение замкнутой операционной схемы индивидуальными и сокращенными обозначениями и терминами.
- Ф.3. Формирование R-интерпретируемого главного рода структуры (библиотека алгоритмов представляется методом и постоянна для данного класса проектируемых систем).
- Ф.4. R-интерпретация конститuent первой очереди.
- Ф.5. R-интерпретация конститuent n-ой очереди ( $n > 1$ ).
- Ф.6. Формирование заданий на разработку алгоритмов и программ, обеспечивающих R-интерпретацию конститuent, R-интерпретация которых не предусмотрена.
- Ф.7. Формирование исходных данных для представления R - интерпретации в виде проекта.
- Ф.8. Формирование исходных данных для размещения проекта на материальном носителе (в памяти ЭВМ или в виде текста).
- Ф.9. Формирование исходных изменений.
- Ф.10. Формирование производных изменений.

### Раздел III. СОСТАВ КОМПЛЕКСА ПРОГРАММ.

Состав основных алгоритмов и программ определяется в настоящем техническом задании путем перечисления функций автоматизированного проектирования, выполняемых на ЭВМ. Описания этих функций /приведенные в разделе 1У/, составленные в языке определений, изложенных в разделе 1, представляют собой исходные данные для разработки входных и выходных форм человеко-машинного процесса проектирования, алгоритмов и программ. В приводимом ниже списке знаком \* отмечены функции, не включенные в укрупненную блок-схему работы метода, приведенную в разделе П.

- M1 Ввод специальных базовых родов структур.
- M2 Формирование главного или промежуточного рода структуры.
- M3\* Формирование стандартных терминов и обозначений.
- M4\* Ввод сокращенных и индивидуальных терминов и обозначений.
- M5\* Раскрытие конститuent данного рода структуры через составляющие их конститuentы с помощью индивидуальных терминов и обозначений.
- M6\* Формирование сжатых терминов и обозначений.
- M7\* Формирование T-интерпретаций и антиинтерпретаций /конститuent/, удовлетворяющих заданным условиям.
- M8 Формирование списка конститuent  $n$ -ой очереди главного рода структуры.
- M9 Формирование R-интерпретации конститuent главного рода структуры /т.е. множества  $Aut(z)$ /.

- MI0      Формирование списка конститунт, R-интерпретация которых не предусмотрена.
- MI1      Распределение R-интерпретированных конститунт по разделам проекта.
- MI2      Формирование проекта.
- MI3 \*    Формирование элементов основных входов, требующих внесения изменений.
- MI4 \*    Выбор способа внесения изменений.
- MI5 \*    Формирование измененной части проекта.

## Раздел 1У. ЗАДАНИЕ НА РАЗРАБОТКУ КОМПЛЕКСА АЛГОРИТМОВ И ПРОГРАММ.

### 1. О диалоговом режиме использования комплекса программ.

В настоящем разделе приводятся описания машинных функций, которые и играют роль заданий на разработку алгоритмов и программ.

В описаниях содержатся указания на использование тех или иных машинных операций при выполнении функций проектировщиком. Вместе с изложенной в разделе II схемой работы метода эти описания дают возможность рассматривать совокупность требуемых программ как единый комплекс, обеспечивающий диалоговые элементы в работе метода.

Для обеспечения диалогового режима помимо чередования функций машины и проектировщика в схеме работы метода должен быть предусмотрен вопросный вид выходных машинных форм для машинных функций М2, М8, М10, а также функции М13. Более точно, эти формы должны состоять из двух частей: заполненной и незаполненной. Функция проектировщика состоит в ответах на вопросы, т.е. в заполнении незаполненных граф машинных документов.

### 2. Описания функций, выполняемых на ЭВМ.

#### М1. Ввод специальных базовых родов структур.

Операция применяется при выполнении функции Ф1 и заключается в пополнении массива базовых родов структур новыми. При этом коды базовых родов структур должны указывать, являются ли они /роды структур/ специальными или нет.

## М2. Формирование главного или промежуточного рода структуры

Операция заключается в построении главного или промежуточного рода структуры по заданному базовому замыканию операционной схемы и указанию соответствующей вершины. Результат представляется либо в стандартных собственных обозначениях, либо с помощью стандартной собственной терминологии, либо в индивидуальных обозначениях, либо в индивидуальной терминологии, либо, наконец, в любом сочетании этих типов представлений. При этом индивидуальные обозначения и термины могут быть использованы только, разумеется, после их введения, т.е. выполнения функции М4.

Операция используется, главным образом, при выполнении функции  $\Phi 1$  (формирование промежуточных родов структур как главных-в базовых замыканиях соответствующих схем), а также функций  $\Phi 2$ ,  $\Phi 3$  и  $\Phi 7$ .

## М3. Формирование стандартных терминов и обозначений

Операция формирования стандартных терминов и обозначений конституэнт (из заданного множества конституэнт) по заданным индивидуальным и сокращенным (или скатым) терминам или обозначениям. Операция носит вспомогательный характер и предназначена для формирования входных данных других машинных функций  $M_i$  в тех случаях, когда эти данные необходимо представить в стандартных обозначениях или терминологии.

## М4. Ввод сокращенных и индивидуальных терминов и обозначений

Помимо формирования массивов индивидуальных и сокращенных терминов и обозначений конституэнт и их Т-интерпретаций необходимо предусмотреть возможность пополнения этих массивов. Результат выполнения операции используется при выполнении машинных операций,  $M_2$ ,  $M_5$ ,  $M_7$  -  $M_{10}$  когда требуется представить выходные данные в индивидуальных и скатых обозначениях и терминах, а также при выполнении функции  $M_3$ .

м5. Раскрытие конституэнт с помощью индивидуальных терминов и обозначений

Операция заключается в замене стандартных обозначений "внутренних понятий" (стр. 17), фигурирующих в записи аксиом, теорем и внутренних термов данного рода структуры, их индивидуальными эквивалентами, а также в аналогичной замене стандартной терминологии индивидуальной.

Операция, таким образом, позволяет формировать "дисциплины", т.е. определения одних понятий через другие в более или менее традиционном виде.

Операция применяется при выполнении функций  $\Phi\Phi.4, \Phi.5, \Phi6$

м6. Формирование скатых терминов и обозначений

По заданным сокращенным обозначениям <sup>и терминам</sup> некоторых Т-интерпретаций конституэнт формируются скатые обозначения и термины любых Т-интерпретаций любых конституэнт операционной схемы.

Операция используется при выполнении функций  $\Phi\Gamma - \Phi5$  для представления результата выполнения операции м7 в скатых терминах и обозначениях.

м7. Формирование Т-интерпретаций и антиинтерпретаций

а) По заданному пути графа схемы, снабженной базовым замыканием, и конституэнте рода структуры, отвечающего началу (или концу) пути, построить ее Т-интерпретацию (соответственно, антиинтерпретацию);

б) то же для класса конституэнт (при заданном пути графа, схемы). Например, построить Т-интерпретацию всех конституэнт данного рода или для некоторого подмножества конституэнт данного рода построить их антиинтерпретацию;

в) то же для класса путей при заданном множестве конститuent. Например, построить все главные интерпретации всех конститuent данного рода структуры, или все главные антиинтерпретации данного подмножества множества всех конститuent данного рода структуры, в частности фиксированной конститuent;

г) для данного подмножества множества конститuent данного рода структуры построить антиинтерпретации этих конститuent, соответствующие одному и тому же пути графа и удовлетворяющих условию максимальности этого пути (относительно выдвинутых требований).

Результаты выполнения операций представляются в стандартных терминах и обозначениях или по специальному указанию - в слатых терминах и обозначениях с помощью операции м6.

Операции используются при выполнении функций ф1-ф5.

### м8. Формирование списка конститuent n-ой очереди главного рода структуры

Понятие конститuent n-ой очереди (п.ж.1) определено при описании функций ф4, ф5.

Операция заключается в построении списка конститuent n-ой очереди по базовому замыканию операционной схемы, списку конститuent (с указанием тех их свободных переменных, интерпретация которых возможна на ЭВМ), требующих специальной вычислительной интерпретации, а при  $n \geq 2$  еще должен быть задан список конститuent, R-интерпретация которых выполнена (этот список содержит конститuent всех очередей до n-1 включительно, но не исчерпывается ими).

Операция используется для выполнения функций  $\phi_4$  и  $\phi_5$ .

м9. Формирование R-интерпретаций конституэнт <sup>главного</sup> рода структуры

Операция заключается в выполнении по специальным алгоритмам R-интерпретации конституэнт, R-интерпретация свободных переменных (или аргументов) которых задана. К специальным алгоритмам относится, например, теоретико - множественные операции: пересечение, объединение, прямое произведение и т.п., а также алгоритмы нахождения экстремумов. Вообще, в этот список могут быть включены любые алгоритмы, которые решают задачу построения одних множеств по другим, при условии, что и те и другие множества представлены в главном роде структуры внутренними терминами (включая базовые множества и родовую структуру). В частности, сюда включаются разработанные на основе результата выполнения операции м10 алгоритмы (и программы), позволяющие строить R-интерпретации или соответствующих конституэнт.

Операция используется при выполнении функций м12, а также может применяться при выполнении функций  $\phi_4$ - $\phi_8$ .

м10. Формирование списка конституэнт, R-интерпретация которых не предусмотрена

В общем случае для выполнения этой операции должны быть заданы:

- а) главный род структуры;
- б) список конституэнт, R-интерпретация которых выполнена на ЭВМ;
- в) список всех конституэнт, подлежащих R-интерпретации;
- г) список конституэнт, подлежащих специальной R-интерпретации, с указанием свободных переменных (или аргументов), R-интерпретация которых предполагается выполненной.

Искомый список конститунт, (т.е. конститунт), R-интерпретация которых не предусмотрена), состоит из конститунт, принадлежащих списку в), но не принадлежащих ни списку г), ни списку б), и, кроме того, не принадлежащих к числу тех, R-интерпретация которых может быть выполнена с помощью операции м9.

Операция используется для выполнения функции ф6.

Примечание: функции м8, м9, м10 выполняются одновременно.

### м11. Распределение R-интерпретированных конститунт по разделам проекта

Для выполнения операции должны быть заданы:

- а) список конститунт Главного рода структуры, R-интерпретация которых должна составить содержание проекта;
- б) список имен уровней иерархии разделов проекта;
- в) распределение конститунт списка а) по разделам проекта или правила распределения;
- г) список названий разделов проекта или правило формирования списка (из числа воспринимаемых ЭВМ); результатом выполнения операций является как-либо закодированный текст проекта, т.е. "размещаемое".

### м12. Формирование проекта

Входными данными для выполнения операции являются:

- а) указание носителя проекта (магнитная лента ЭВМ или широкая печать);
- б) размещаемое (результат выполнения операции м11);
- в) список имен уровней иерархии частей носителя;
- г) распределение разделов проекта по частям носителя.

Результатом выполнения операции является проект в виде текстового документа или в виде записи на магнитной ленте.

Примечание: операции M11 и M12 могут выполняться одновременно.

### Описание функций внесения изменений (Ф9, Ф10, M13, M14, M15)

Разработка метода в части предоставления специальных возможностей для внесения изменений (дабы избежать "ненужного перепроектирования") находится в начальной своей стадии и может, по-видимому, осуществляться только по мере освоения метода. Здесь возникают два важных вопроса, которые на настоящем этапе в принципе решены быть не могут:

1. Внесение каких изменений потребует практика проектирования с использованием настоящего метода.

2. Внесение каких изменений реально осуществимо на ЭВМ при существующем состоянии программирования (включая возможности ЭВМ).

Поэтому ниже дается только общая предполагаемая схема внесения изменений, которая может служить исходным пунктом для работы в этом направлении.

*A priori* внесения изменений могут потребовать следующие данные:

1. Граф схемы.
2. Открытая схема.
3. Список базовых определений.
4. Базовые замкания.
5. Индивидуальная терминология и обозначения.
6. Сокращенная терминология и обозначения.
7. Список конститuent, подлежащих R-интерпретации.
8. Список конститuent, подлежащих специальной R-интерпретации.

9. Свободные переменные (входящие в конститутанты списка п.8), R-интерпретация которых необходима для специальной R-интерпретации соответствующих конститутант.
10. R-интерпретация конститутант n-ой очереди.
11. Список имен уровней иерархии разделов проекта.
12. Распределение конститутант списка п.7 по разделам проекта.
13. Список названий разделов проекта.
14. Носитель проекта.
15. Список имен уровней иерархии частей носителя.
16. Распределение разделов проекта по частям носителя.

Таким образом мы имеем 16 входов в машинные операции, в которые можно вносить изменения.

Назовем эти входы основными входами. Каждый основной вход снабжается правилом внесения изменений. В ответ на заданное "исходное" изменение ЭВМ выделяет те данные всех основных входов, которые должны подвергнуться "производным" изменениям. Затем, вычисляя объем необходимых изменений и сравнивая его с объемом полного перепроектирования, ЭВМ выносит решение о внесении изменений, но в режиме внесения изменений или в режиме перепроектирования.

В первом случае проектировщик получает с ЭВМ список тех элементов основных входов, которые требуют внесения изменений.

Во втором случае предлагается весь процесс проектирования повторить сначала.

После внесения проектировщиком всех необходимых изменений ЭВМ формирует только измененную часть проекта.

### 3. Место представленных в отчетах [3], [4] алгоритмов в рассматриваемом комплексе

В отчете [3] представлены две группы алгоритмов: алгоритмы, реализующие некоторые теоретико-множественные операции, и алгоритмы подстановки кортежей (в двух вариантах).

Алгоритмы теоретико-множественных операций — это алгоритмы выполнения функции М9, т.е. алгоритмы, реализующие R-интерпретацию конститuent R-интерпретируемого (главного) рода структуры. Конечно, это только небольшая (но необходимая) часть алгоритмов, обеспечивающих выполнение этой функции, хотя с другой стороны номенклатура этих алгоритмов, если не ограничиваться классом систем, будет всегда неполна даже с теоретической точки зрения. В то же время для некоторых классов систем, повидимому, окажется достаточным и упомянутых выше алгоритмов.

Алгоритм подстановки кортежей (отчет [3], (задача 1<sup>0</sup>, лист 53) может быть использован при разработке алгоритмов и программ выполнения функций М.2 (формирование главного или промежуточного рода структуры), рассматривая выражение  $Op(x_1, \dots, x_n)$  как кортеж знаков, и М.8 (формирование списка конститuent n-ой очереди) при  $n = 1$ , рассматривая соотношения, которые определяют конститuent, как кортежи свободных переменных.

Другие задачи, связанные с алгоритмом подстановки кортежей и представленные в отчете [3] (задачи 2<sup>0</sup> - 6<sup>0</sup>, лист 53) могут быть использованы в операциях М.7 (формирование T-интерпретаций и антиинтерпретаций, удовлетворяющих заданным условиям), например, для конститuent "род структуры" и М.8 при  $n > 1$ .

В отчете [4] приведены алгоритмы, реализующие операции над родами структур и обобщенными родами структур. Эти алгоритмы играют роль основных алгоритмов выполнения функции М.2.

## ЛИТЕРАТУРА

1. "Разработка методов проектирования АСУ капитального строительства Минэнерго СССР". Раздел А. Часть 1. "Метод машинного проектирования АСУ" (отчет по теме 4903, арх.номер ОМ-149957). "Оргэнергострой", М., 1972 г.
2. "Разработка методов проектирования АСУ капитального строительства Минэнерго СССР". Раздел А. Часть 2. "Математические модели для машинного проектирования целевых АСУ", (отчет по теме 4903, арх.номер ОМ-149506). "Оргэнергострой", М., 1972 г.
3. "Разработка методов проектирования АСУ капитального строительства Минэнерго СССР". Раздел А. Часть 3. "Комплекс алгоритмов для машинного проектирования целевых АСУ" (отчет по теме 4631, арх.номер ОМ-156378) "Оргэнергострой", М., 1973 г.
4. "Разработка методов проектирования АСУ капитального строительства Минэнерго СССР". Раздел А. Часть 4. "Формальное проектирование системы управления для Главэнергостройпрома", (отчет по теме 4631, арх.номер ОМ-156399), "Оргэнергострой", М., 1973 г.
5. "Разработка и применение методов проектирования АСУ". Раздел А. Применение методов машинного проектирования АСУ. Регулярные морфизмы родов структур и связанные с ними категории. (отчет по теме 4631 арх.номер ОМ-156 647 ), "Оргэнергострой", М., 1974 г., часть III.
6. Н.Бурбаки. Теория множеств. "Мир", М., 1965.