

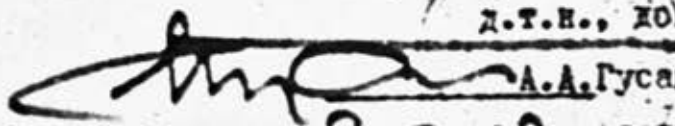
Государственный комитет Совета Министров СССР
по делам строительства

Центральный научно-исследовательский и проектно-
экспериментальный институт автоматизированных
систем в строительстве
ЦНИИМАСС

УДК: 681.322.06:721.011:69.003:65.014.011.56

№ гос. регистрации
Инв. №

"Утвержден"
Директор ЦНИИМАСС
д.т.н., доцент


А.А. Гусев
3003 1977 г.

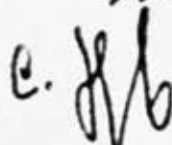
Технический проект экспериментальной
системы пакетов прикладных программ
автоматизированного проектирования
систем организационного управления
/логико-интерпретационный блок про-
ектирования/

Всего томов 3. Том 3, книга 2

Шифр 15-2-76


И.о. зам. руководителя отделения АСУС  И.М. Левин

Руководитель темы,
зав. отделом ИО АСУС



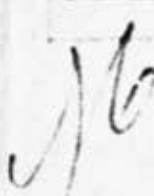
С.П. Никаноров

Ответственный исполнитель,
с.н.с., к.ф.-м.н.



Д.Б. Персиц

Москва - 1977 г.



АКАДЕМИЯ НАУК УССР
ВЫЧИСЛИТЕЛЬНЫЙ ЦЕНТР ОДЕССКОГО ОТДЕЛЕНИЯ
ИНСТИТУТА ЭКОНОМИКИ

ЭКСПЕРИМЕНТАЛЬНАЯ СИСТЕМА ППП АВТОМАТИЗИРОВАННОГО
ПРОЕКТИРОВАНИЯ СИСТЕМ ОРГАНИЗАЦИОННОГО УПРАВЛЕНИЯ

Д 4. СОДЕРЖАНИЕ ЗАДАЧ ПРОЕКТИРОВАНИЯ
И АЛГОРИТМЫ ИХ РЕШЕНИЯ

на 142 листах

"Согласовано"
Руководитель темы
С. П. НИКАНОРОВ
Ст. научн. сотрудник к. ф.-м. н.
Д. Б. ПЕРСИЦ

Ответственный исполнитель
Ст. инженер
А. В. АИЗЕНШТАТ
Ст. инженер
Б. А. ЗАКС
Зав. отделом моделирования
и системной техники
Г. Я. ПОРТНОВ

45-8-76
ТПЗм2

ОГЛАВЛЕНИЕ

	Стр
1.4. Содержание задач проектирования и алгоритмы их решения	2
3. Перевод представления, Т- и антиинтерпретация (ППП переводе представлений).....	4
3.1. Постановка задачи (содержание и метод решения)..	4
3.2. Алгоритмы управления функционированием ППП (УПП).	10
3.3. Описание алгоритмов модулей тела пакета	14
3.4. Характеристики информационных массивов ППП и связь с другими ППП	51
3.5. Ограничения и возможности расширения ППП	56
3.6. Средства программного контроля и контроля результатов	57
4. Перепроектирование и внесение изменения (ППП внесения изменения)	58
4.1. Постановка задачи (содержание и метод решения)..	58
4.2. Алгоритмы управления функционированием ППП (УПП).	61
4.3. Описание алгоритмов модулей тела пакета.....	69
4.4. Характеристики информационных массивов ППП и связь с другими ППП.....	130
4.5. Ограничения и возможности расширения ППП	142
4.6. Средства программного контроля и контроля результатов	143

15.2.78

ТПТЗм2

3. Перевод представлений, T- и антиинтерпретации
(ППП перевода представлений)

3.1. Постановка задачи (содержание и метод решения).

Одной из особенностей математического аппарата построения главного рода структуры является то, что при выполнении операций над родами структур, дополнениями и отождествляющими отображениями применяются стандартные обозначения конститuent. Поэтому обозначения конститuent результата операции отличаются от обозначения тех конститuent аргументов, из которых они произошли. Для того, чтобы правильно проинтерпретировать конститuentу главного рода структуры, проектировщику необходимо знать, из каких конститuent базисных родов и дополнения получена данная. Эта необходимость возникает и во многих других случаях (при внесении изменений в базисные рода структур и дополнения в операционную схему и др.).

Вышеизложенное говорит о необходимости представить проектировщику специальные средства для ретроспективного анализа конститuent родов структур, т.е. средства для определения истории этих конститuent.

Нужно также предоставить возможность использования индивидуальных обозначений и терминов конститuent, которые могут быть применены проектировщиком для семантической идентификации конститuent.

Полезно предусмотреть возможность выдачи на печать выражения отдельных конститuent.

Все указанные средства, а также многие другие сервисные средства предоставляются проектировщику ППП перевода

15-2-76
ТПЗар2

представлений.

Ниже будут определены $2^6=64$ вида представлений конститuent. Каждое представление характеризуется шестизначным двоичным числом $\alpha_1 \alpha_2 \alpha_3 \alpha_4 \alpha_5 \alpha_6$. Значения признаков, соответствующих каждому разряду, приведены в таблице.

	α_1	α_2	α_3	α_4	α_5	α_6
O :	Имя	Обозначение	Собственное	Стандартное	Полное	Базовое
I :	Выражение	Термин	Схемное	Индивидуальное	Сжатое (внутреннее)	Простое

Для разъяснения таблицы введем следующие определения.

Определение 3.1.1. Представление конститuent называется простым внутренним, если оно не содержит базового индекса рода структуры или дополнения, содержащего данную конститuentу, а также не выражает соответствие этой конститuentу вершине графа операционной схемы.

Определение 3.1.2. Полным схемным кодом называется знакосочетание вида

$$n_0 \rightarrow (M_1) n_1 \rightarrow (M_2) n_2 \rightarrow \dots \rightarrow (M_z) n_z,$$

где n_i ($i = 0, \dots, z$) - номера вершин графа операционной схемы

такие, что каждая дуга $n_{i-1} n_i$ ($i = 1, \dots, z$) принадлежит множеству дуг графа (т.е. одна из вершин n_{i-1} и n_i является аргументом в операции, соответствующей другой вершине);

M_i - номер аргумента, которым является одна из вершин n_{i-1} и n_i для операции, соответствующей другой вершине; стрелки \rightarrow - указывают направление (аргумент \rightarrow результат).

Определение 3.1.3. Кратким схемным кодом называется условное обозначение, вводимое проектировщиком для идентификации схемного кода.

Определение 3.1.4. Т-интерпретацией $f_{\alpha, \beta}$ называется отображение множества конституэнт β -ого аргумента операции α на множество конституэнт результата операции, определяющее процесс перехода конституэнт аргумента в конституэнты результата.

Определение 3.1.5. Антиинтерпретацией $f_{\alpha, \beta}^{-1}$ называется обратное к $f_{\alpha, \beta}$ отображение (вообще говоря, неоднозначное) множества конституэнт β -ого результата операции α на множество конституэнт аргумента операции, определяющее историю конституэнт результата.

Определение 3.1.6. Т- и антиинтерпретация вдоль пути, определяемого схемным кодом, называется композиция отображений $f_{\alpha, \beta}$ или $f_{\alpha, \beta}^{-1}$, отвечающих последовательным дугам пути.

Каждое из 64 представлений, введенных выше, имеет вид

$$\mathcal{B}(K) [\mathcal{J}_\delta]$$

где K - полный или краткий схемный код, причем n_2 - номер вершины, соответствующей рассматриваемой конституэнте; \mathcal{B} - внутреннее простое представление конституэнты вершины n_0 , из которой данная конституэнта получается в результате Т- или антиинтерпретации вдоль пути, определяемого схемным кодом K ;

\mathcal{J}_δ - базовый индекс рода структуры или дополнения, соответствующего n_0 .

Примечание. Базовый индекс \mathcal{J}_δ отсутствует в представлении, если $\alpha_6 = 1$.

При $\alpha_3 = 0$ \mathcal{B} есть имя конституэнты, а при $\alpha_1 = 1$ \mathcal{B} есть выражение конституэнты.

Имя конституэнты может быть представлено обозначением ($\alpha_2 = 0$) или термином ($\alpha_2 = 1$).

При $\alpha_1 = 1$ и $\alpha_2 = 0$ B есть выражение, в котором имена всех конститuent представлены обозначениями, а при $\alpha_1 = 1$ и $\alpha_2 = 1$ - терминами.

Обозначения и термины могут быть стандартными ($\alpha_4 = 0$) и индивидуальными ($\alpha_4 = 1$).

Обозначения конститuent $\Phi_i, E_i, P_i, X_i, C_i, D, M, \Pi_i, K_i, A_i, T_i$ называются стандартными. Стандартные термины получают заменой натуральных чисел их порядковыми названиями, а стандартных букв - некоторыми словосочетаниями:

- Φ - род структуры;
- X - основное базисное множество;
- C - вспомогательное базисное множество;
- D - родовая структура;
- M - тип родовой структуры;
- Π - терм;
- K - тип термина;
- A - аксиома;
- E - дополнение;
- Γ - отображение.

Например, стандартным термином для A_1 будет "первая аксиома", для Π_2 - "второй терм" и т.д.

Индивидуальные обозначения и термины вводятся проектированием по своему усмотрению и несут ~~смысловую~~ ^{смысловую} нагрузку.

Представление конститuent может быть собственным ($\alpha_3 = 0$) и схемным ($\alpha_3 = 1$). При $\alpha_3 = 1$ указание схемного кода K обязательно, причем, если $\alpha_3 = 0$, то указывается полный схемный код, а если $\alpha_3 = 1$, то указывается краткий схемный код.

При $\alpha_3 = 0$ и $\alpha_4 = 0$ приводится "вырожденный" схемный код $K = \kappa_0$. При $\alpha_3 = 0$ и $\alpha_4 = 1$ схемный код вообще не приводится.

Поле (J_g) присутствует в представлении конституэнт лишь при $\alpha_6 = 0$.

Пакет прикладных программы перевода представлений осуществляет перевод конституэнт из одного представления в другое. На данном этапе разработки метода машинного проектирования систем организационного управления представляется нецелесообразным реализовать все 2^{12} типов перевода (каждое представление и каждое представление).

Сделаем некоторые ограничительные предположения, упрощающие работу пакета.

1) Задаваемое представление может быть одним из следующих:

001001
000001
001011

2) Искомое представление удовлетворяет следующим условиям:

- а) $(\alpha_1=1) \wedge (\alpha_2=1) = 0$ (т.е. логично);
- б) $\alpha_3 = 0$;
- в) $(\alpha_2=1) \wedge (\alpha_4=0) = 0$;
- г) $\alpha_5 = 0$.

В задаче перевода представлений можно выделить следующие функции:

- 1) ввод в БД индивидуальных терминов и обозначений (осуществляется СУБД);
- 2) ввод в БД сокращений схемных кодов (осуществляется СУБД);
- 3) анализ задания на перевод представлений (осуществляется УПН пакета перевода представлений);
- 4) Т- и антиинтерпретация конституэнт в соответствии со

схемным кодом (осуществляется модулем Т-и антиинтерпретации тела пакета перевода представления);

5) перевод из представления 000001 в другие собственные представления (осуществляется различными модулями тела пакета, в зависимости от вида перевода);

6) контроль процесса перевода представления (осуществляется всеми модулями и УПП пакета перевода представления).

1222

ТЛЗм2

3.2. Алгоритм управления функционированием ППП перевода представлений.

Управление функционированием ППП перевода представлений производит управляющая программа пакета (УПП), представленная ниже.

Работа ППП перевода представлений начинается с ввода в ЭВМ управляющего оператора перевода представлений. ГУП производит расшифровку содержания управляющего оператора и передачу управления УПП перевода представлений. УПП вызывает модуль ввода задания на перевод представлений (который вводит форму 9 в МОЗУ) и заменяет в ней все сокращенные и обобщенные схемные коды полными. В результате работы модуля получается массив МТПШ. Затем происходит выделение множества конститuent, представленных к переводу. УПП обрабатывает каждую запись массива МТПШ и по ней ставит в соответствие выделенному множеству конститuent множество конститuent, полученных в результате перевода представлений. При этом формируются строки выходного массива ППК. По окончании обработки всех строк массива МТПШ происходит распечатка массива ППК (форма IO). На этом работа пакета заканчивается и управление передается ГУПу.

Блок-схему УПП пакета перевода представлений см. на рис. 3.2.1.

В блок-схеме приняты следующие обозначения:

- МОВЗ - модуль ввода задания на перевод представлений ;
- МОТА - модуль Т- и антиинтерпретации по схемному коду ;
- МОЯ1 - модуль перевода 000011 - 040111 ;
- МОЯ2 - модуль перевода 000011 - 100111 ;
- МОВЗ - модуль нахождения базового индекса ;
- ППК - массив результатов перевода конститuent ;
- МТПШ - массив требований на перевод представлений конститuent ;
- ℓ - число строк массива МТПШ, увеличенное на 1.
- КС - множество конститuent, полученное из множества, указанного в поле Ф.9 "Статус конститuent" выделением конститuent типов, приведенных в поле Ф.9 "Тип конститuent" ;

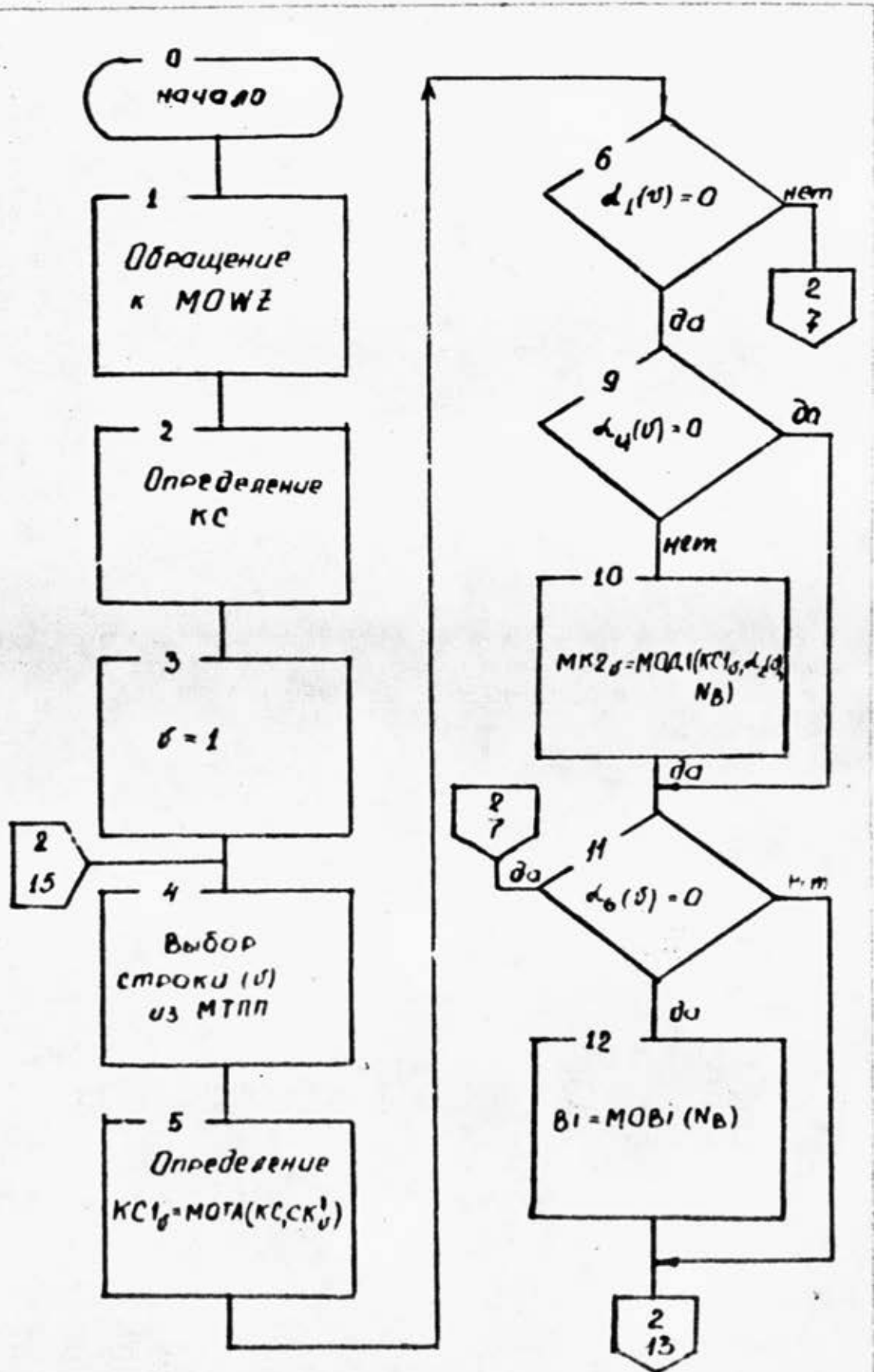
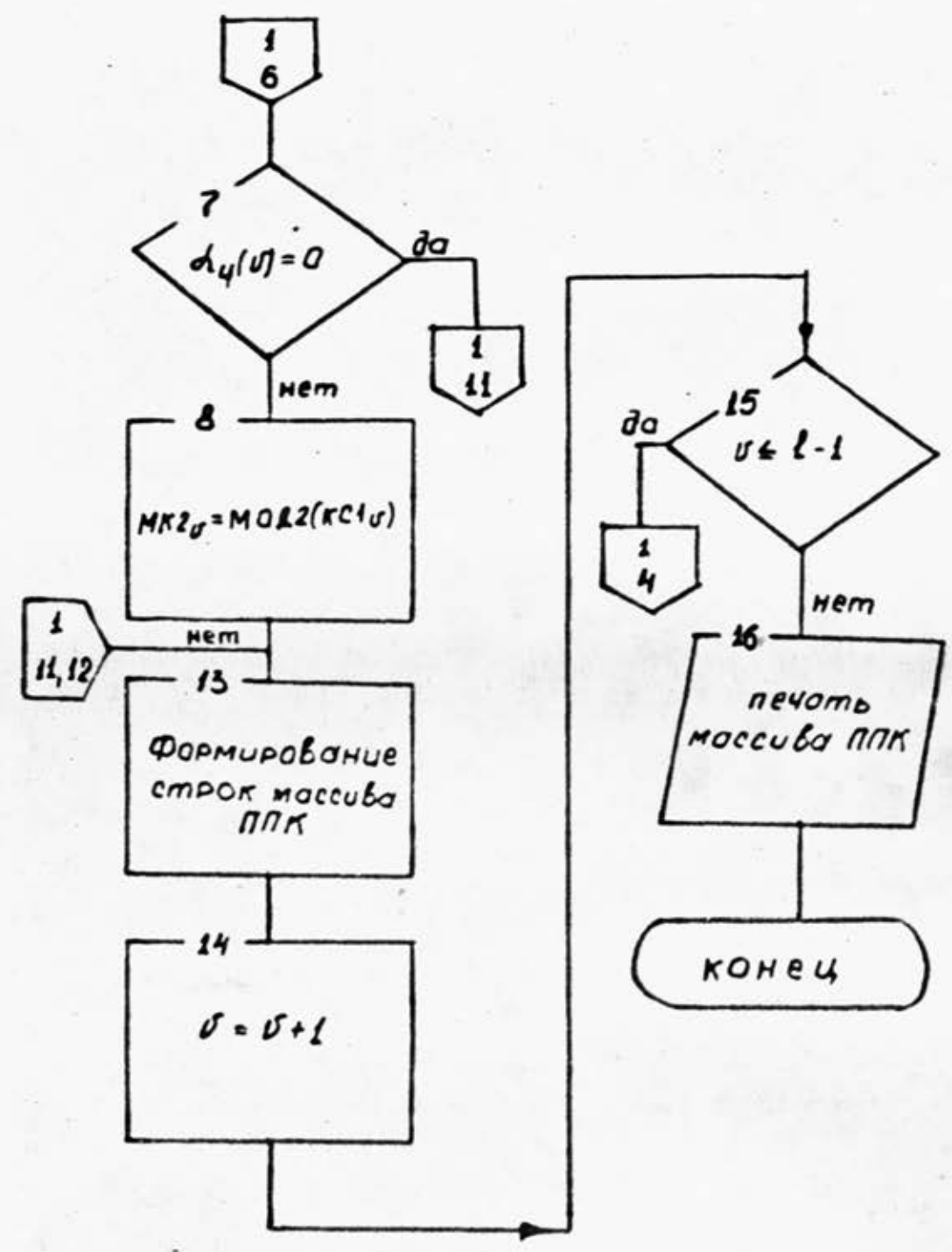


Рис. 3.2.1

(Вмест 1)



15.2.26
Т.П.З.м.2

Рис 3 2 1

(Лист 2)

- CK'_i - полный схемный код, соответствующий i -ой строке массива МТПШ;
- d_i - значение i -того знака в коде представления конститuent, соответствующего i -ой строке массива МТПШ;
- N_i - номер вершины, соответствующей результату перевода конститuent;
- Bi - значение базового индекса.

В блок-схеме приведены следующие блоки.

Бл.01. Обращение к модулю MOU_2 .

При обращении к модулю происходит ввод задания на перевод представлений конститuent. В результате работы модуля формируется массив требований на перевод представлений конститuent (МТПШ).

Бл.02. Определение множества конститuent КС.

Из множества, указанного в поле Ф.9 "Статус конститuent" выделяются конститuentы типов, приведенных в поле Ф.9 "Тип конститuent".

Бл.04. Выбор очередной строки массива МТПШ.

Бл.05. Получение КС K'_i множества имен конститuent с помощью модуля Т- и антиинтерпретации по схемному коду.

Исходные данные для модуля:

- множество конститuent КС;
- полный схемный код CK'_i .

В результате работы модуля формируется множество K'_i , полученное из множества КС Т- и антиинтерпретацией по схемному коду CK'_i .

Бл.10. Получение множества MOU_2 с помощью модуля перевода $OOOOII \rightarrow O_{n_2}OIII$.

Модуль MOU_2 осуществляет замену стандартных имен конститuent множества K'_i их индивидуальными терминами (если $\alpha_i \cdot \beta_i = 1$) или индивидуальными именами (если $\alpha_i \cdot \beta_i = 0$).

В результате работы модуля получаем множество MOU_2 .

Бл.08. Получение множества MOU_2 с помощью модуля перевода $OOOOII \rightarrow IOOIII$.

Исходные данные для работы модуля:

- множество конститuent K'_i .

15-2-86

ТПтЗин2

Модуль МОД2 осуществляет замену в выражениях конституэнт КСИ⁰ стандартных имен конституэнт их индивидуальными именами.

В результате работы модуля образуется множество выражений $МХ2^0$.

Бл.12. Блок нахождения базового индекса с помощью модуля $МОВ_1$.

Бл.13. Формирование строк массива ШК^{*}) соответствующих строке (v^0) МТШ.

Номер требования и номер схемного кода берутся из строки (v^0) МТШ. Стандартное имя конституэнты выбирается из множества КС. В качестве номера вершины выступает последний элемент схемного кода.

Поле "Базовый индекс" заполняется в случае $\alpha_e(i^2) = 1$ базовым индексом рода структуры или дополнения, соответствующего вершине v^0 .

Поле "Внутреннее простое представление" заполняется элементами множества $МХ2^0$.

Бл.16. Печать массива ШК.

Массив ШК распечатывается по Ф.Ю.

3.3. Описание алгоритмов модулей тела пакета.

В тело пакета входят следующие модули:

- модуль одношаговой Т-интерпретации ;
- модуль одношаговой антиинтерпретации ;
- модуль Т- и антиинтерпретации по схемному коду ;
- модуль расшифровки обобщенного схемного кода ;
- модуль ввода задания на перевод представлений ;
- модуль перевода $0000II \rightarrow 0000III$;
- модуль перевода $0000II \rightarrow 1000III$;
- модуль нахождения базового индекса.

В данном пункте описаны алгоритмы этих модулей.

*) Структура записи массива ШК описана в п.3.4. данного документа.

3.3.1. Модуль одношаговой Т-интерпретации.

Модуль предназначен для получения одношаговой Т-интерпретации конститuent, т.е. интерпретации по схемному коду $\gamma_0 \rightarrow (M) \gamma_1$.

Исходными данными для работы модуля являются:

σ - отождествляющее отображение, представленное в форме:

$$\sigma = \left\{ \begin{array}{l} \alpha_1, \alpha_2, \dots, \alpha_\lambda \\ \beta_1, \beta_2, \dots, \beta_\lambda \end{array} \right\}$$

n - натуральное число ($n \leq \alpha_\lambda$).

Блок-схема модуля представлена на Рис. 3.3.1.

Через m обозначим число $\beta_i \in \sigma$, которое определяется соответствием: $\alpha_i \rightarrow \beta_i$ ($n = \alpha_i$).

В блок-схеме модуля приведены следующие блоки:

Блок 3. Значение величины m находится по формуле $m = \beta_i$ (в случае, когда $n = \alpha_i$).

Блок 9. Значение величины m определится по формуле

$$m = \beta_i - \Delta \frac{\beta_i \beta_{i-1}}{\alpha_i - \alpha_{i-1}}, \text{ где } \Delta = \alpha_i - n;$$

(в случае, когда $n < \alpha_i$).

Блок 7. Величине m присваивается отрицательное значение (в случае, когда $n > \alpha_\lambda$).

3.3.2. Модуль одношаговой антиинтерпретации.

Модуль предназначен для получения одношаговой антиинтерпретации конститuent, т.е. интерпретации по схемному коду $\gamma_0 \leftarrow (M) \gamma_\lambda$.

Исходными данными для работы модуля является:

σ - отождествляющее отображение, представленное в форме:

$$\sigma = \left\{ \begin{array}{l} \alpha_1, \alpha_2, \dots, \alpha_\lambda \\ \beta_1, \beta_2, \dots, \beta_\lambda \end{array} \right\}$$

n - натуральное число ($n \leq \beta_\lambda$).

Блок-схема модуля представлена на Рис. 3.3.2.

Через m обозначим число $\alpha_i \in \sigma$, которое определяется соответствием: $\alpha_i \rightarrow \beta_i$ ($n = \beta_i$).

В блок-схеме модуля приведены следующие блоки:

Блок 3. Значение величины m определится по формуле:

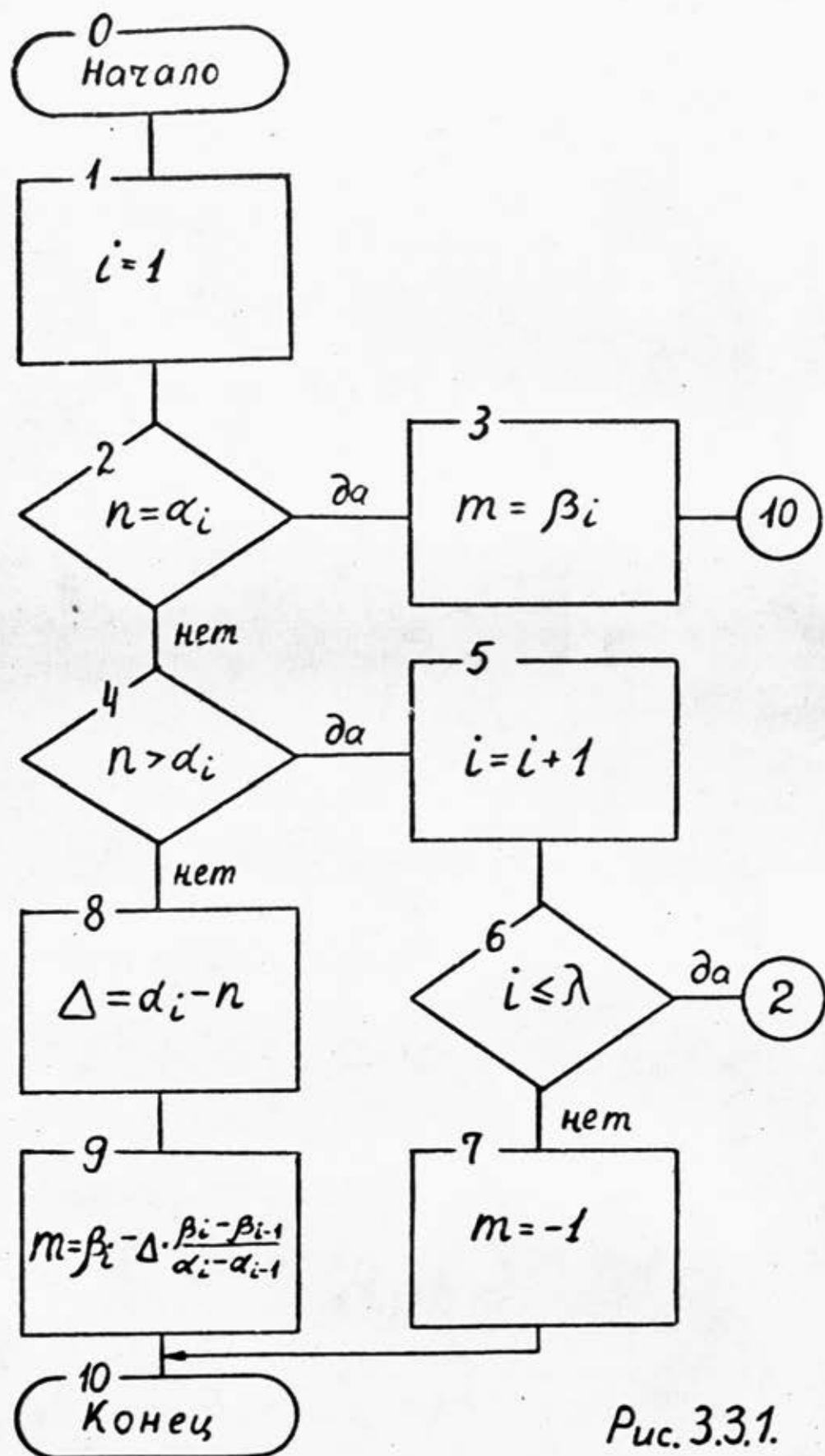


Рис. 3.3.1.

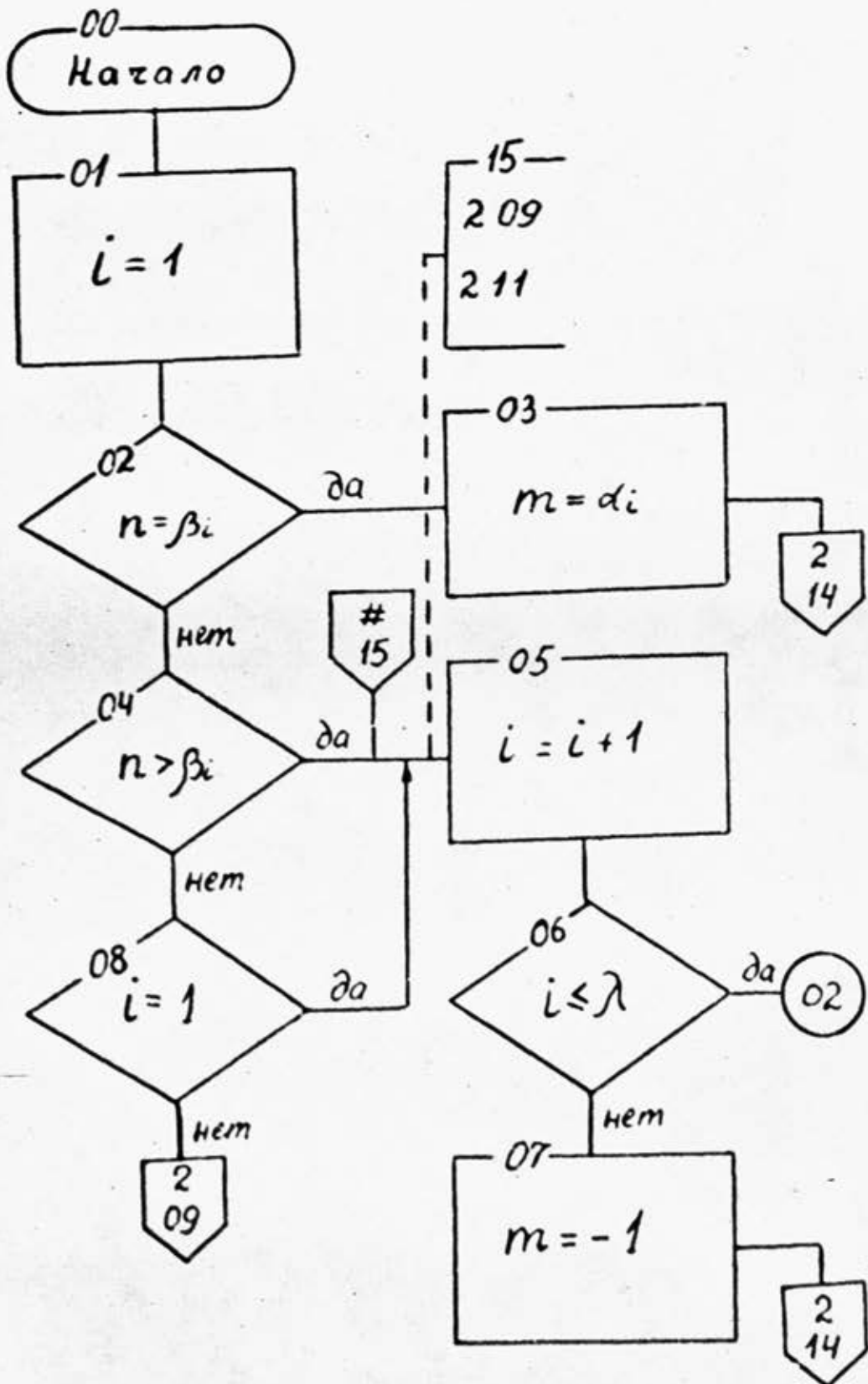
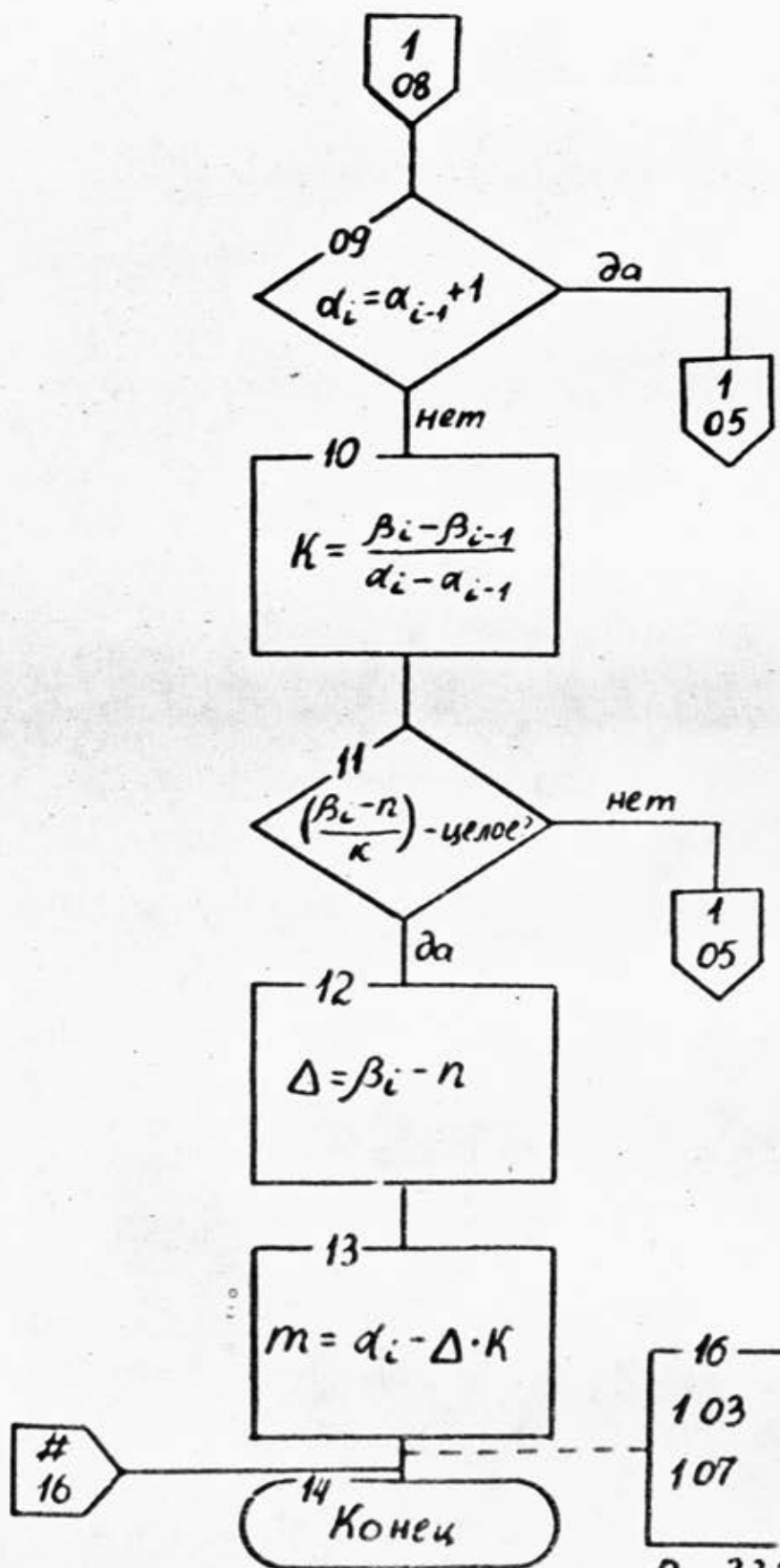


Рис. 3.3.2 (лист 1)



$m = d_i$ (в случае, когда $n = \beta_i$).

Блок 7. Величине m присваивается отрицательное значение (в случае, когда $n > \beta_i$).

Блок 13. Величина m находится по формуле:

$$m = d_i - \Delta \kappa \quad , \quad \text{где } \Delta = \beta_i - n \quad \text{и} \quad \kappa = \frac{\beta_i - \beta_{i-1}}{d_i - d_{i-1}}$$

(в случае, когда $n < \beta_i$).

3.3.3. Модуль Т- и антиинтерпретации по схемному коду.

Исходными данными для работы модуля являются:

- КС множество имен конституэнт, подлежащих Т- и антиинтерпретации по схемному коду;

- полный схемный код, представленный в виде

$$B_0 K_1 M_1 V_1 \dots K_N M_N V_N.$$

Результатом работы модуля является множество КС I имен конституэнт, полученное из КС Т- и антиинтерпретацией по данному схемному коду.

Блок-схема модуля представлена на Рис. 3.3.3.

В блок-схеме приняты следующие обозначения:

КС - множество j имен конституэнт, подлежащих Т- и антиинтерпретации по схемному коду;

UV - модуль одношаговой Т-интерпретации;

AV - модуль одношаговой антиинтерпретации;

FORST1 - подпрограмма, заменяющая стандартные обозначения конституэнт их порядковыми номерами в роде структуры;

FORST2 - подпрограмма, определяющая по порядковому номеру конституэнт их стандартные обозначения;

R - признак, различающий направление стрелки:

$$R = \begin{cases} 0, & \text{если } K_i = ' \rightarrow ' \\ 1, & \text{если } K_i = ' \leftarrow ' \end{cases}$$

В блок-схеме приведены следующие блоки.

Бл. 2,6,7. Блоки запоминания очередных значений

$$V_{i-1}, M_i, V_i.$$

Бл. 3,5. Блоки определения признака R:

$$R = \begin{cases} 0, & \text{если } K_i = ' \rightarrow ' \\ 1, & \text{если } K_i = ' \leftarrow ' \end{cases}$$

Бл. 8. Определение номера величины операции

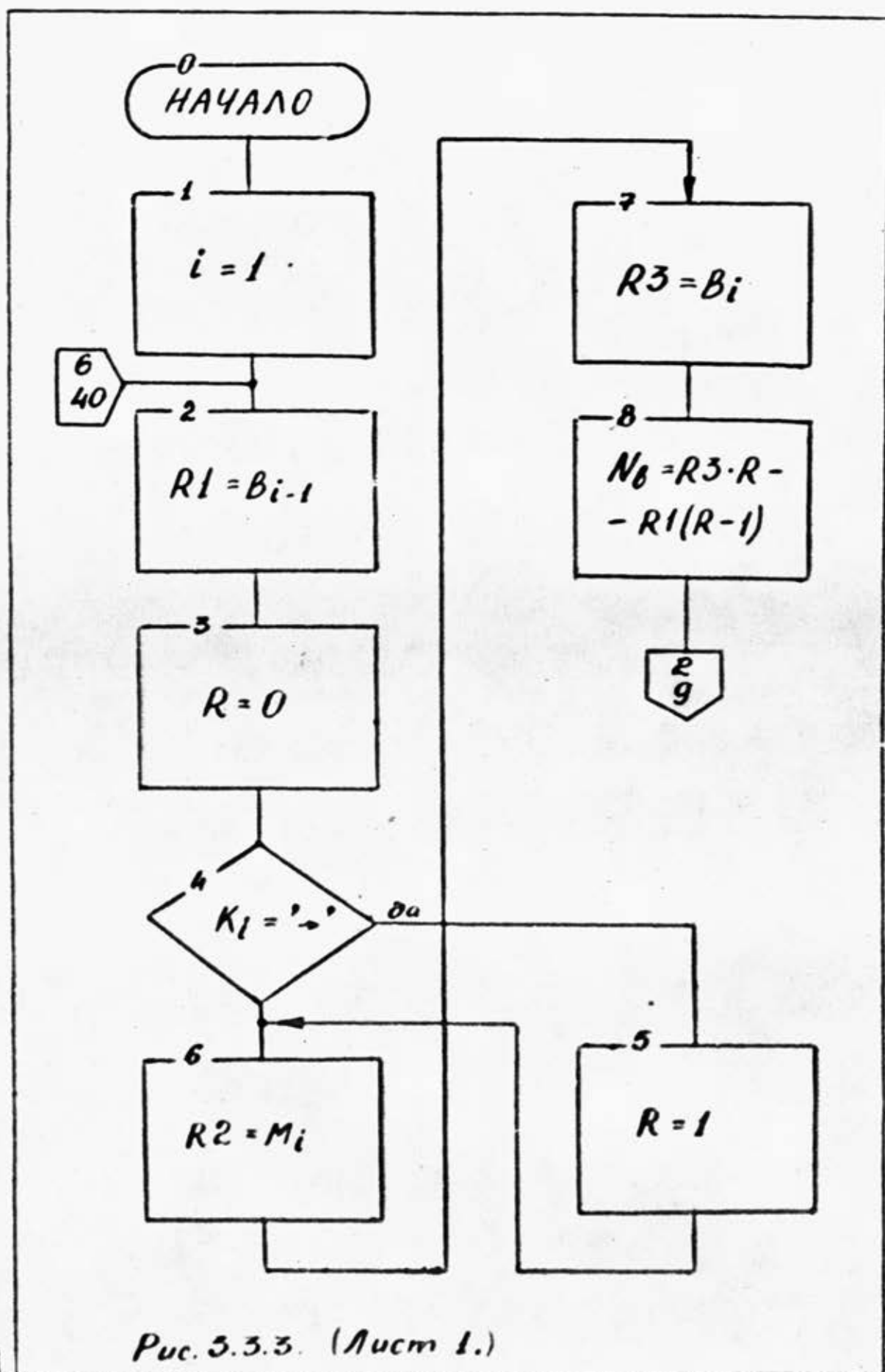


Рис. 3.3.3. (Лист 1.)

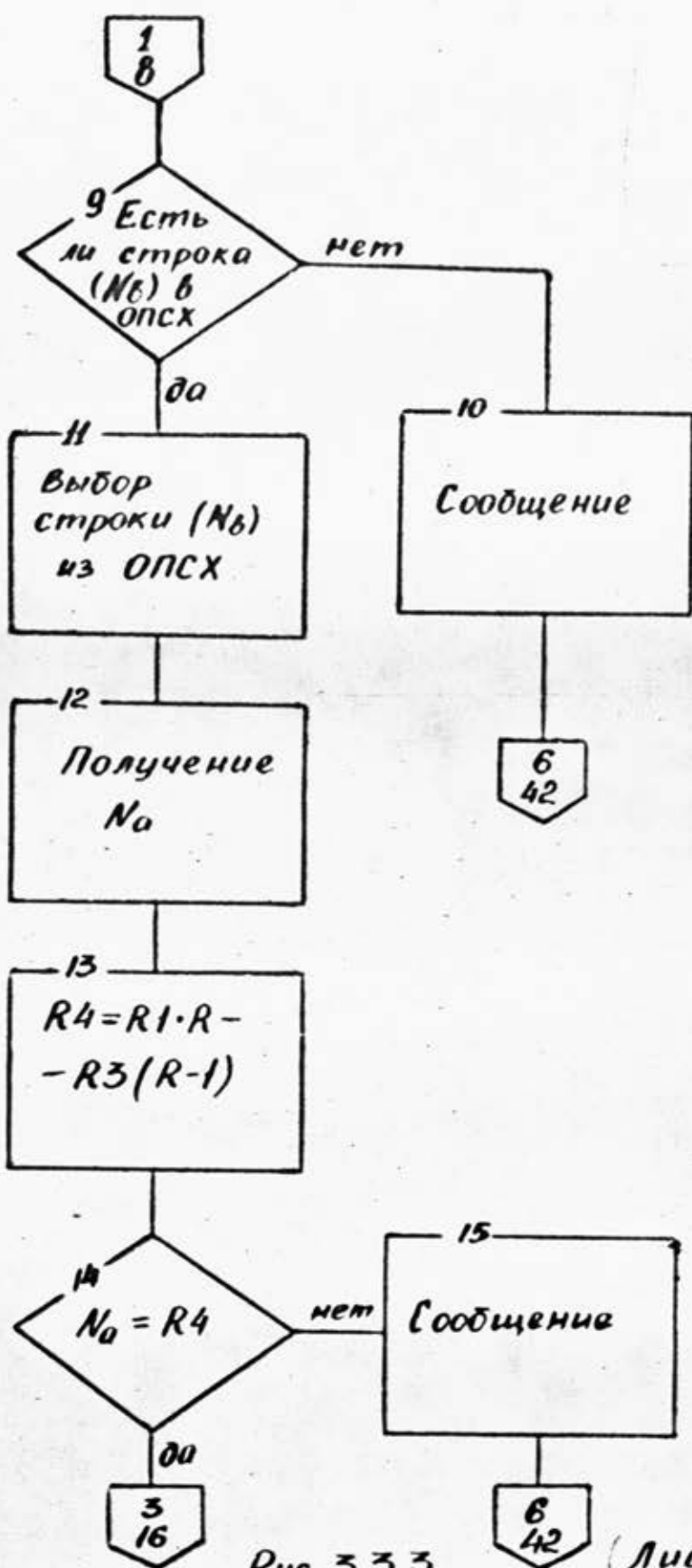


Рис. 3.3.3.

(Лист 2.)

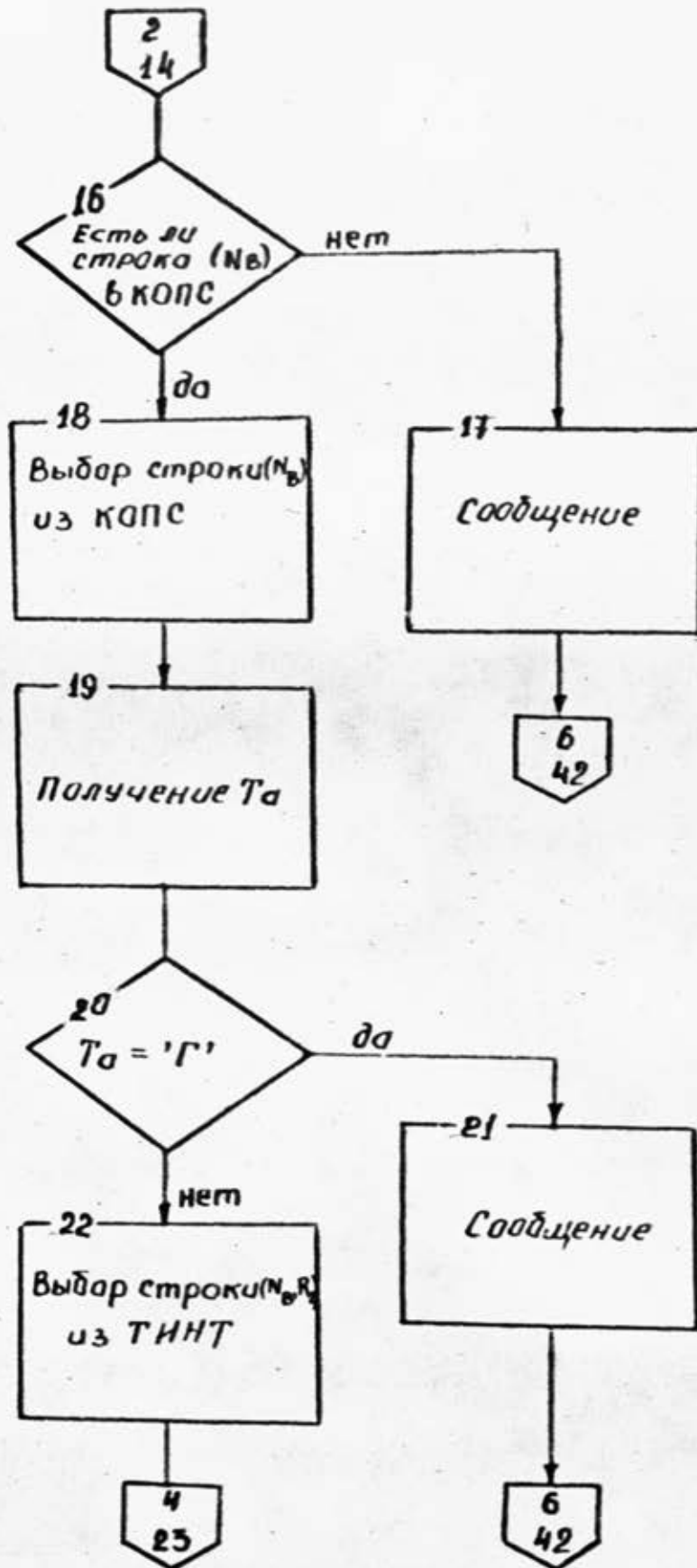


Рис. 33.3 (Лист 3)

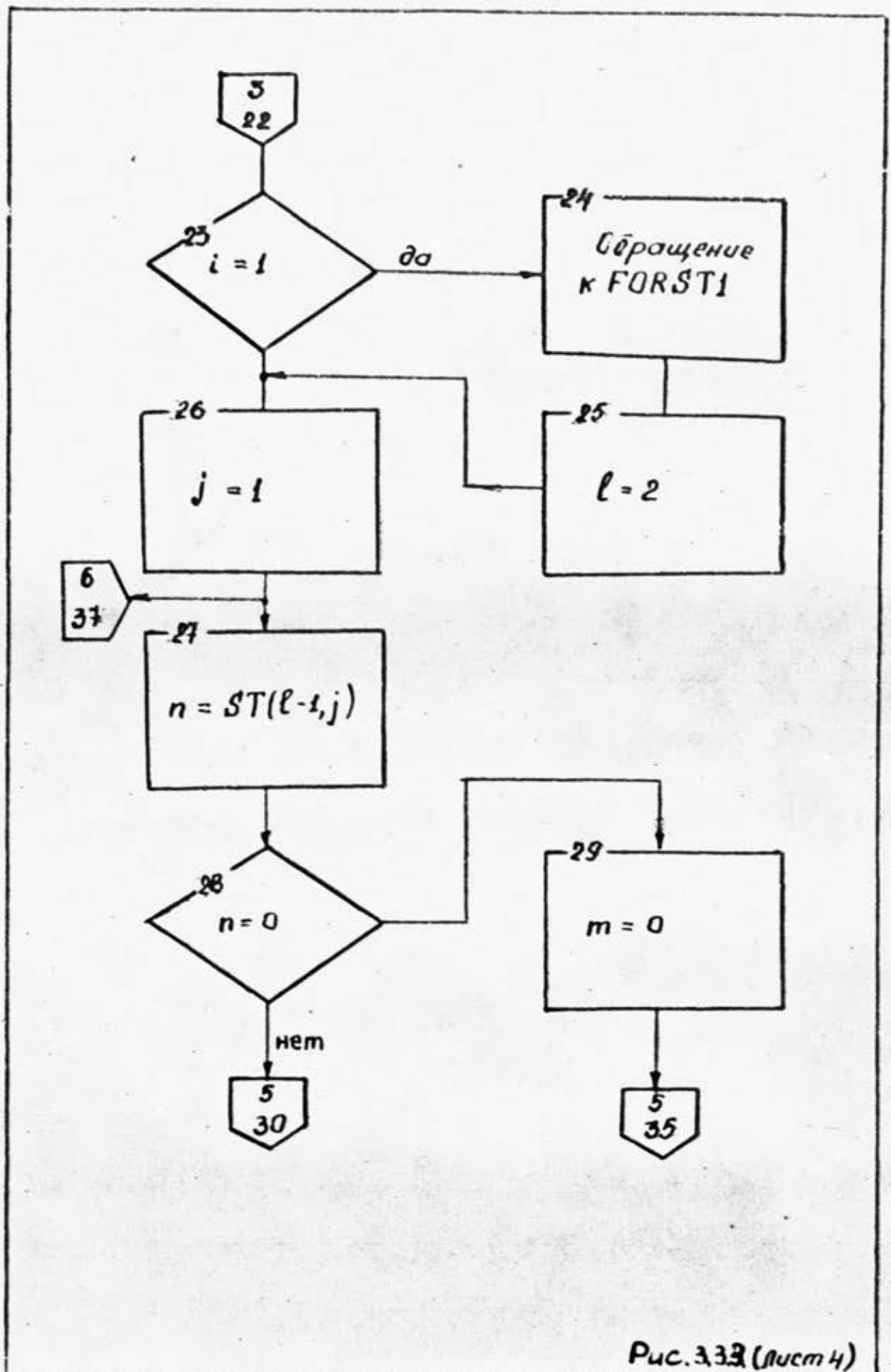
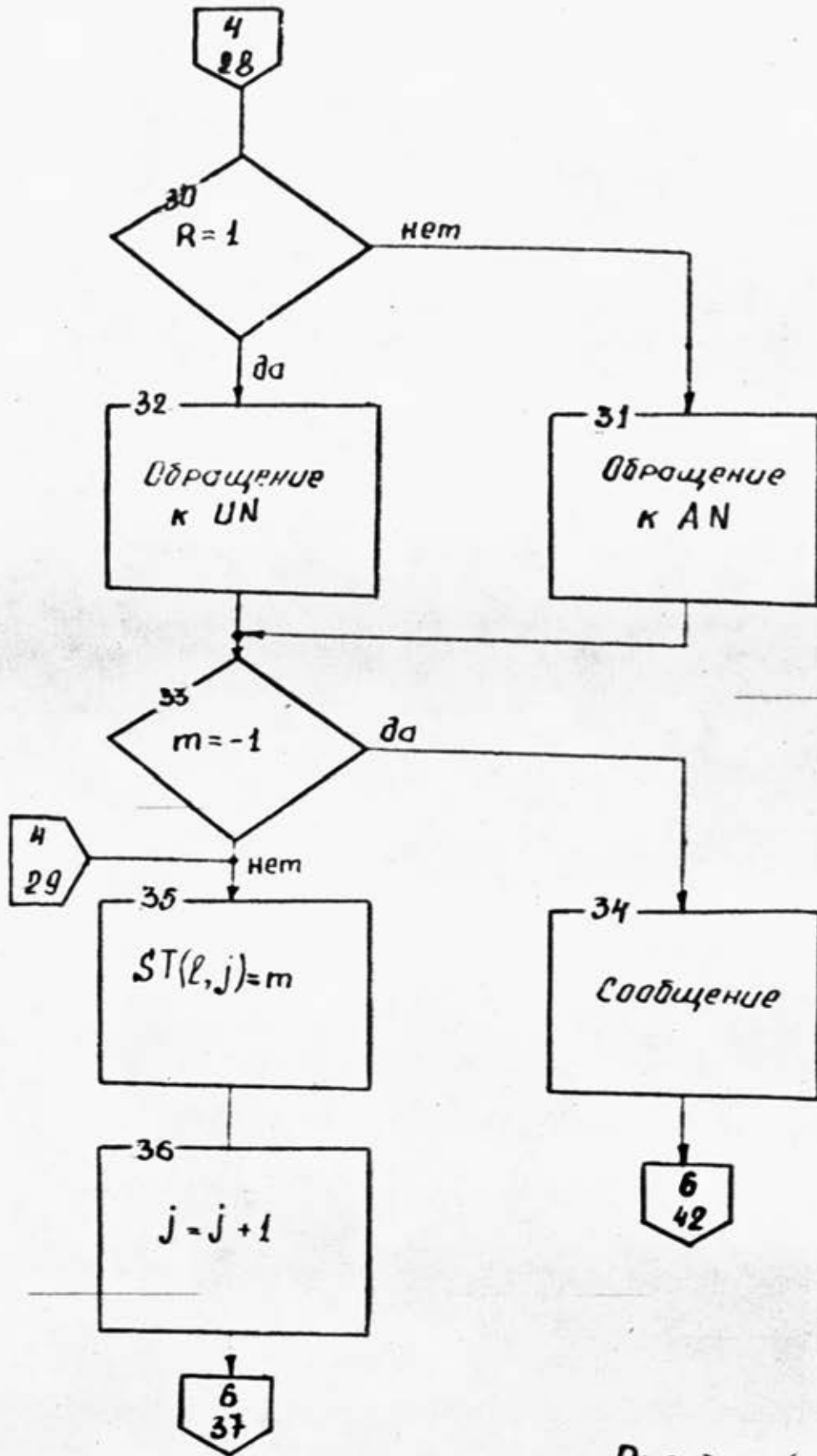
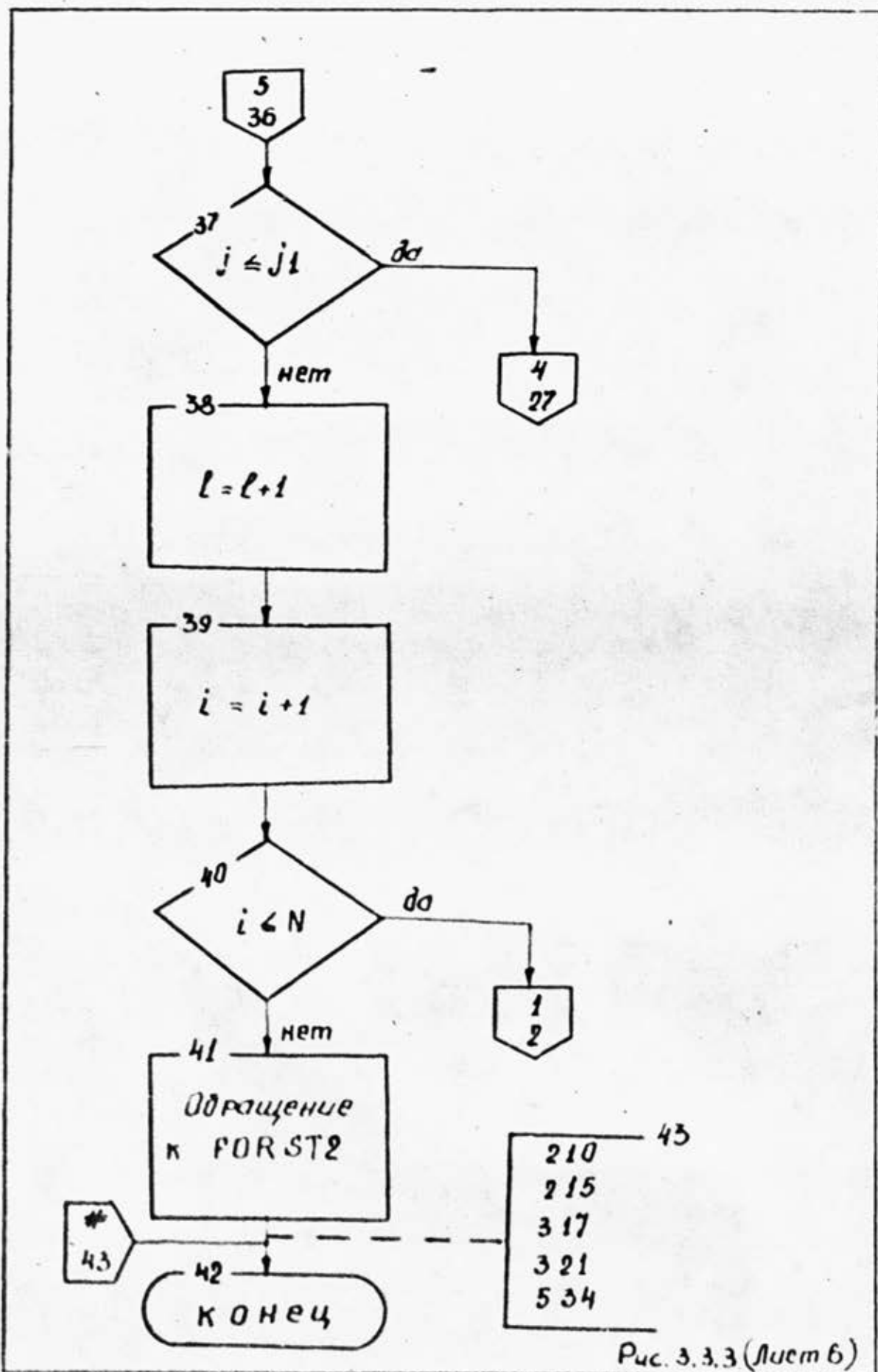


Рис. 332 (лист 4)



15-2-76
ТНТ3м2

Рис. 3.3.3. (Лист 5)



$$N_6 = \begin{cases} R_3, & \text{если } R = 1 \\ R_1, & \text{если } R = 0 \end{cases}$$

Бл. 13. Определение номера вершины аргумента операции

$$R_4 = \begin{cases} R_1, & \text{если } R = 1 \\ R_3, & \text{если } R = 0 \end{cases}$$

Бл. 10. Выдача сообщения об ошибке в случае, когда вершина с номером N_6 в ОПСХ отсутствует ("Вершина N_6 отсутствует")

Бл. 11. Выбор строки ОПСХ, соответствующей вершине N_6 .

Бл. 12. Нахождение R_2 -ого аргумента в строке ОПСХ, соответствующей вершине N_6 (N_d).

Бл. 15. Выдача сообщения об ошибке в случае, когда вершина N_6 не является аргументом операции ("Вершина N_6 не является аргументом").

Бл. 17. Выдача сообщения об ошибке в случае, когда операция не проведена - вершина с номером N_6 в КОПС отсутствует ("Операция не произведена").

Бл. 18. Выбор строки КОПС, соответствующей вершине N_6 .

Бл. 19. Нахождение R_2 -ого аргумента в строке КОПС, соответствующего вершине N_6 (T_a).

Бл. 21. Выдача сообщения об ошибке в случае, когда аргументом операции является отождествляемое отображение.

Бл. 22. Выбор строки ТИИТ, соответствующей вершине N_6 и аргументу R_2 .

Бл. 24. Обращение к подпрограмме FORSt I.

Исходные данные для работы подпрограммы:

- множество имен КС;
- признак R ;
- запись строки (N_6, R_2) из ТИИТ.

Подпрограмма FORSt I производит преобразование множества КС по формулам соответствия, приведенным в п. 1.3. данного документа.

В результате работы подпрограммы получаем множество чисел, соответствующих множеству имен КС - $St(l, j)$ ($l=1; j=1, j_1$).

Бл. 31, 32. Обращение к модулям UN или AN.

Исходными данными для работы модулей являются:

- число n , определяемое в бл. 27;
- запись строки (N_6, R_2) из ТИИТ.

Бл.34. Выход сообщения об ошибке в случае, когда в результате одношаговой Т- или антиинтерпретации получено отрицательное число.

Бл.41. Обращение к подпрограмме FORSt 2.

Исходные данные для работы подпрограммы:

- $(\ell - 1)$ -ая строка множества $ST - ST(\ell - 1, \kappa)$;
- признак R ;
- запись строки $(N\ell, R2)$ из ТИИТ.

Подпрограмма FORSt 2 производит преобразование множества ST по формулам соответствия, приведенным в п.1.3. данного документа.

В результате работы подпрограммы получаем множество стандартных обозначения конститuent КСИ.

3.3.4. Модуль расшифровки обобщенного схемного кода.

Исходными данными для работы модуля является:

- обобщенный схемный код, представленный в виде: $B_0 Z B_N$.
- Блок-схема модуля представлена на Рис. 3.3.4.

В блок-схеме приняты следующие обозначения:

B_0, B_N - номера вершин операционной схемы;

N_2 - модуль нижнего замыкания;

W_2 - модуль верхнего замыкания;

СХК - массив схемных кодов;

$WJK(z)$ - оператор выделения записей из СХК, последний элемент которых равен z ;

$WJON(z)$ - оператор выделения записей из СХК, первый элемент которых равен z ;

SZEP - оператор формирования записей множества СХК;

Z - признак, определяющий направление обобщенного схемного кода;

K - признак, определяющий направление стрелки в расшифрованном схемном коде;

Γ - граф, полученный из ОПСХ путем нижнего замыкания;

МВ - список номеров вершин, полученных в результате верхнего замыкания;

M - число вершин, полученных в результате верхнего

замыкания.

В блок-схеме приведены следующие блоки.

Бл.2,5. Блоки определения признака K :

$$K = \begin{cases} \rightarrow, & \text{если } Z = \Rightarrow \\ \leftarrow, & \text{если } Z = \Leftarrow \end{cases}$$

Блок 8. Получение графа Γ с помощью модуля нижнего замыкания ($NZ(R1)$).

Исходными данными для работы модуля являются:

- операционная схема;
- номер вершины $R1$, по которой производится нижнее замыкание.

Бл.10. Выдача сообщения в случае, когда вершина $R2$ в графе Γ отсутствует.

Бл.11. Получение списка вершин графа Γ с помощью модуля верхнего замыкания по вершине $R2$ ($WZ(R2)$).

Исходными данными для работы модуля являются:

- граф Γ ;
- номер вершины $R2$, по которой производится верхнее замыкание.

Бл.16. Определение M - порядкового номера MB_{i-e} -ого аргумента строки (MB_i) в графе Γ .

Бл.17. Определение значения SXK_j :

$$SXK_j = \begin{cases} MB_{i-e} \cdot X_j(N) \cdot MB_i, & \text{если } K = \rightarrow \\ MB_i \cdot X_j(N) \cdot MB_{i-e}, & \text{если } K = \leftarrow \end{cases}$$

Бл.25. Обращение к подпрограмме $WJVK$.

Исходные данные для работы подпрограммы:

- множество записей SXK_j ($j = \overline{1, J1}$);
- значения MB_e .

Подпрограмма из множества SXK выделяет записи, последний элемент которых равен MB_e , запоминает их и вычерчивает из множества SXK .

В результате работы подпрограммы получаем множество записей MK_p ($p = \overline{1, \bar{p}}$, где \bar{p} - число записей множества MK).

Бл.26. Обращение к подпрограмме $WJVN$.

Исходные данные для работы подпрограммы:

- множество записей SXK_j ($j = \overline{1, J1}$);
- значения MB_e .

Подпрограмма из множества SXK выделяет записи, первым

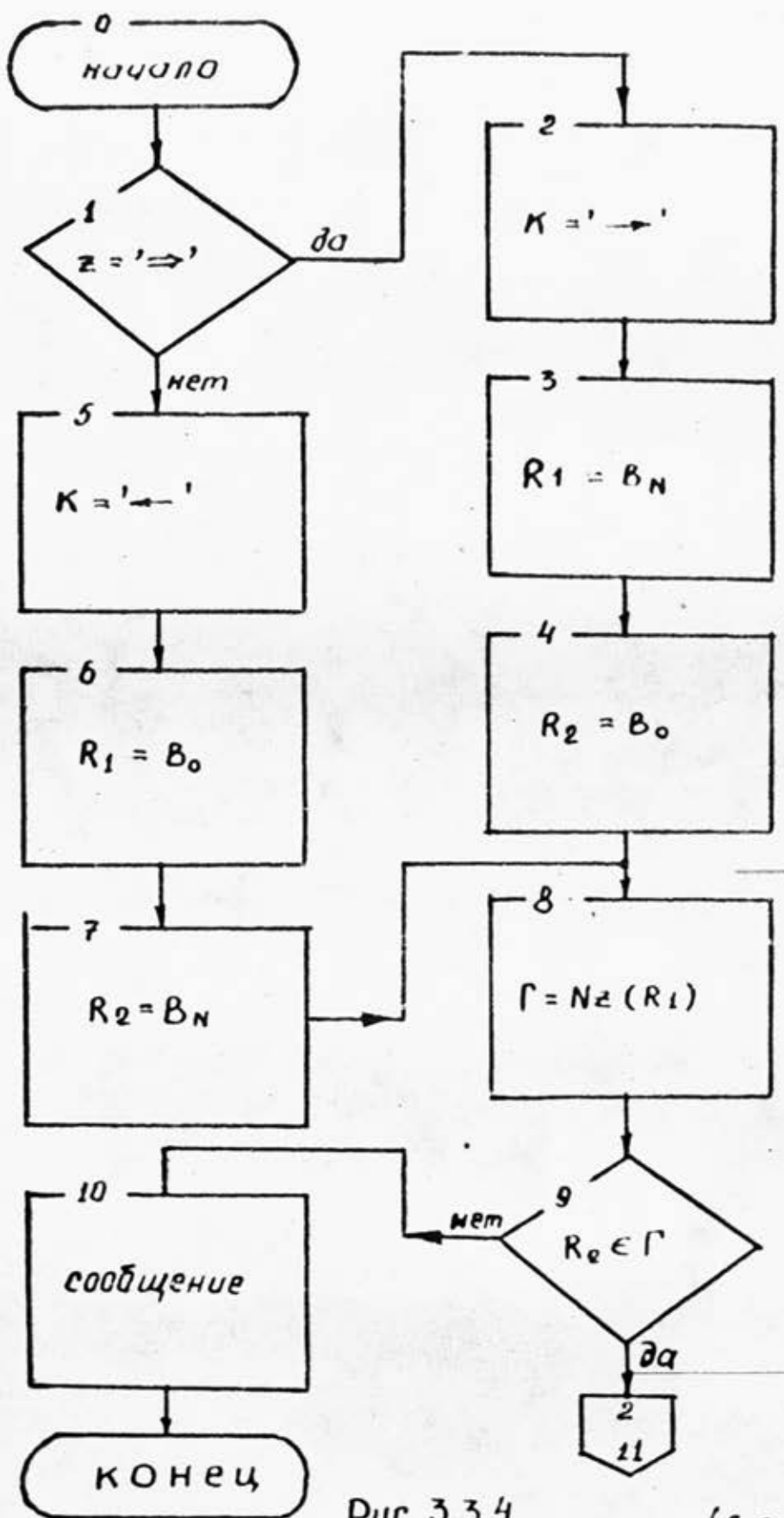


Рис. 3.3.4

(лист 1)

15-2-8
ТПТЗки2

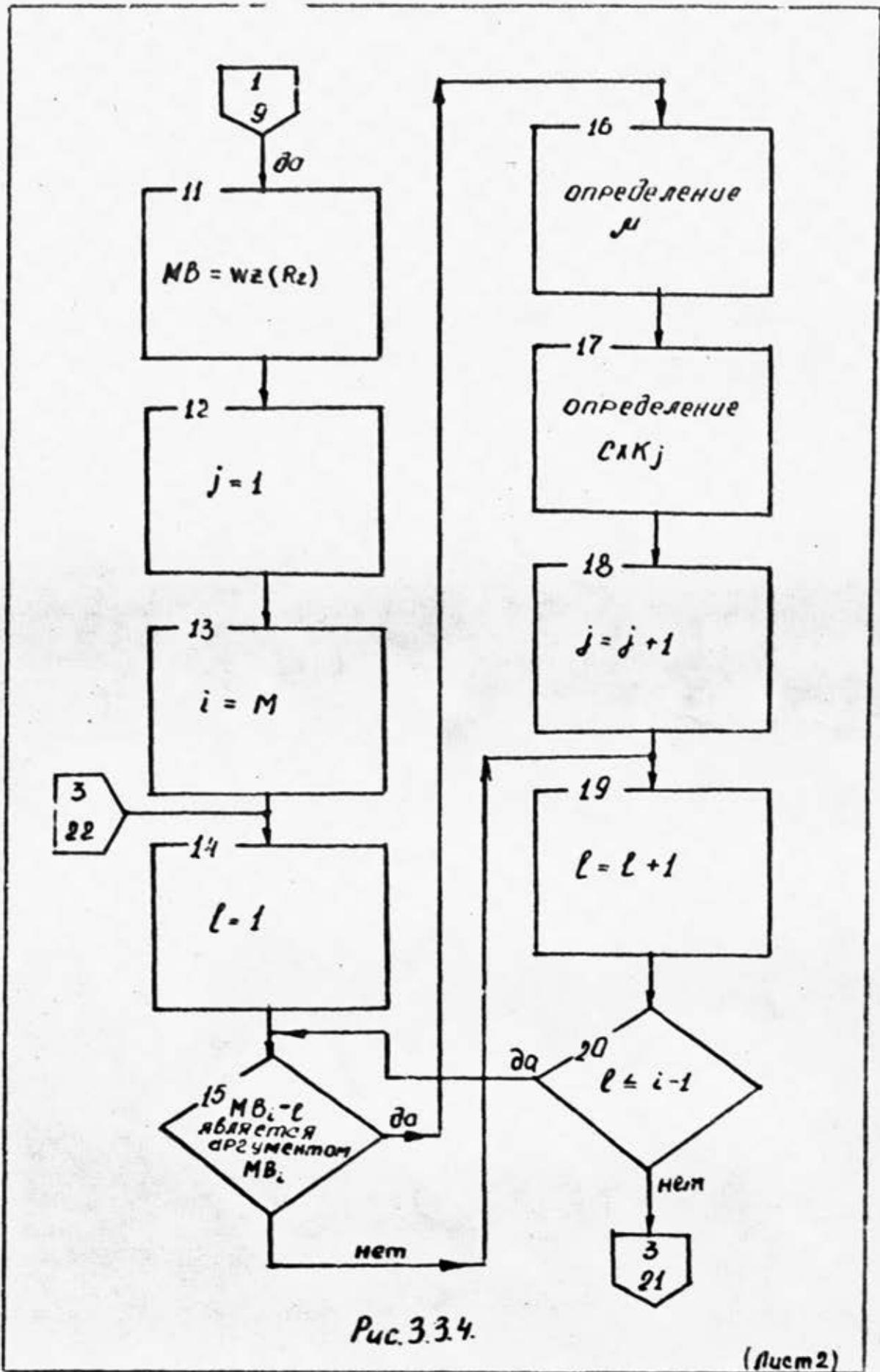


Рис.3.34.

752-Ж
ТПГЗМ²

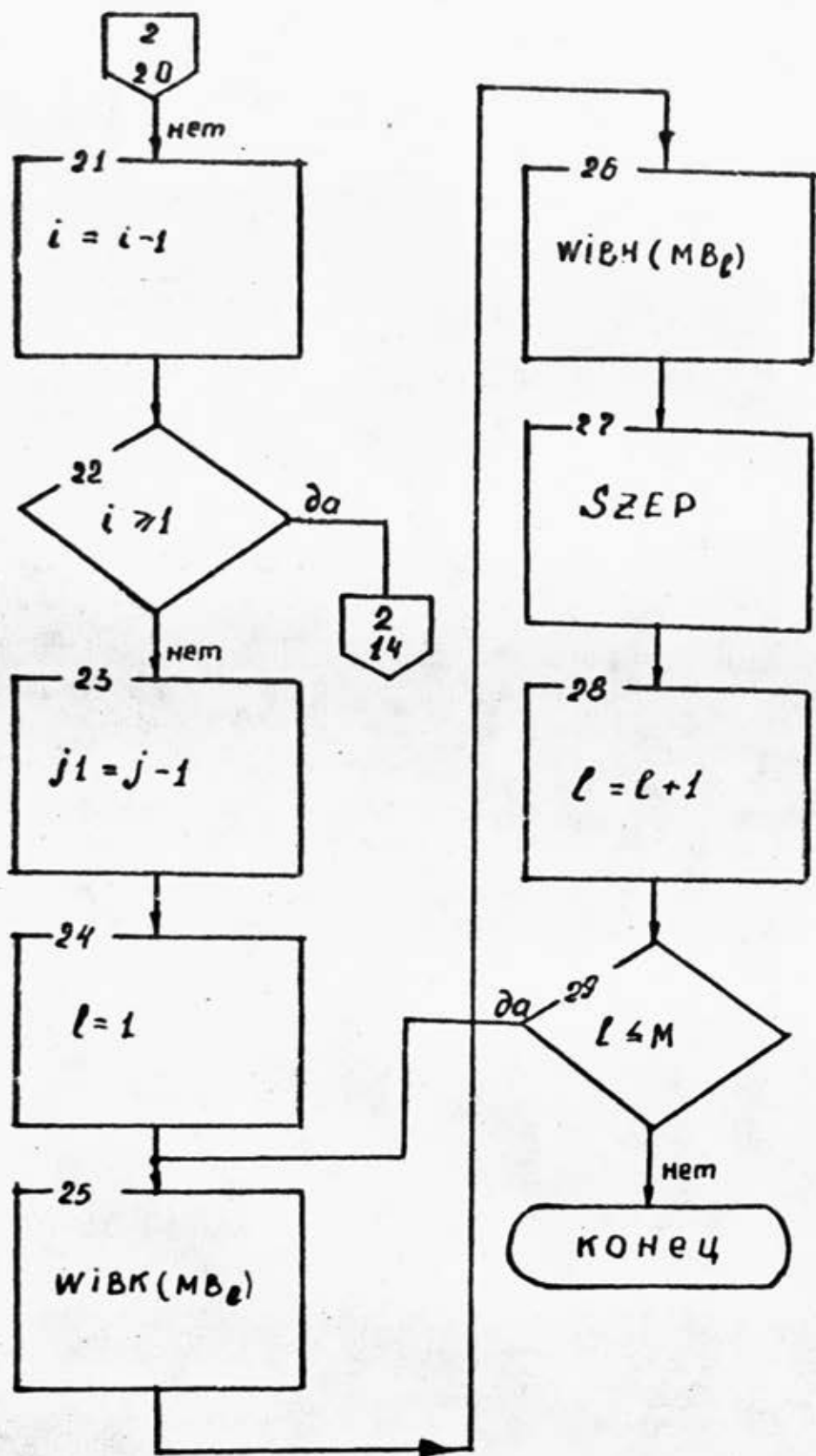


Рис. 3.3.4

элементом которых является KB_e , запоминает их и вычерчивает из множества СХК.

В результате работы подпрограммы получаем множество записей MNp' ($p' = \overline{1, \tilde{p}}$, где число \tilde{p} - число записей множества МН).

Бл.24. Обращение к подпрограмме SZEP.

Исходные данные для работы подпрограммы:

- множество записей МКр ($p = \overline{1, \tilde{p}}$);
- множество записей MNp' ($p' = \overline{1, \tilde{p}}$).

Подпрограмма формирует очередную запись множества СХК путем следующего сцепления каждой записи множества МК с каждой записью множества МН:

$SXK_j = MKp \& MNp'$, где $p = \overline{1, \tilde{p}}$, $p' = \overline{1, \tilde{p}}$
 MNp' - часть записи MNp' , не содержащая первый элемент записи.

В результате работы подпрограммы получаем p' записей СХК ($p' = \overline{1, \tilde{p}}$).

3.3.5. Модуль ввода задания на перевод представлений.

Модуль предназначен для ввода задания на перевод представлений конститuent.

Исходными данными для работы модуля является входная форма Ф.9.

В результате работы модуля получается массив требований на перевод представлений конститuent (МТП).

Блок-схема модуля приведена на Рис. 3.3.5.

В блок-схеме приняты следующие обозначения:

L - номер листа Ф.9;

$Kost(L)$ - количество строк в L -том листе Ф.9;

$List$ - суммарное количество листов в Ф.9;

NOM, NOM' - порядковый номер задания соответственно в Ф.9 и записи результата;

$KOPR, KOPR'$ - код представления соответственно в Ф.9 и в записи результата;

CK, CK' - схемный код соответственно в Ф.9 и в записи результата;

d - признак кода в Ф.9

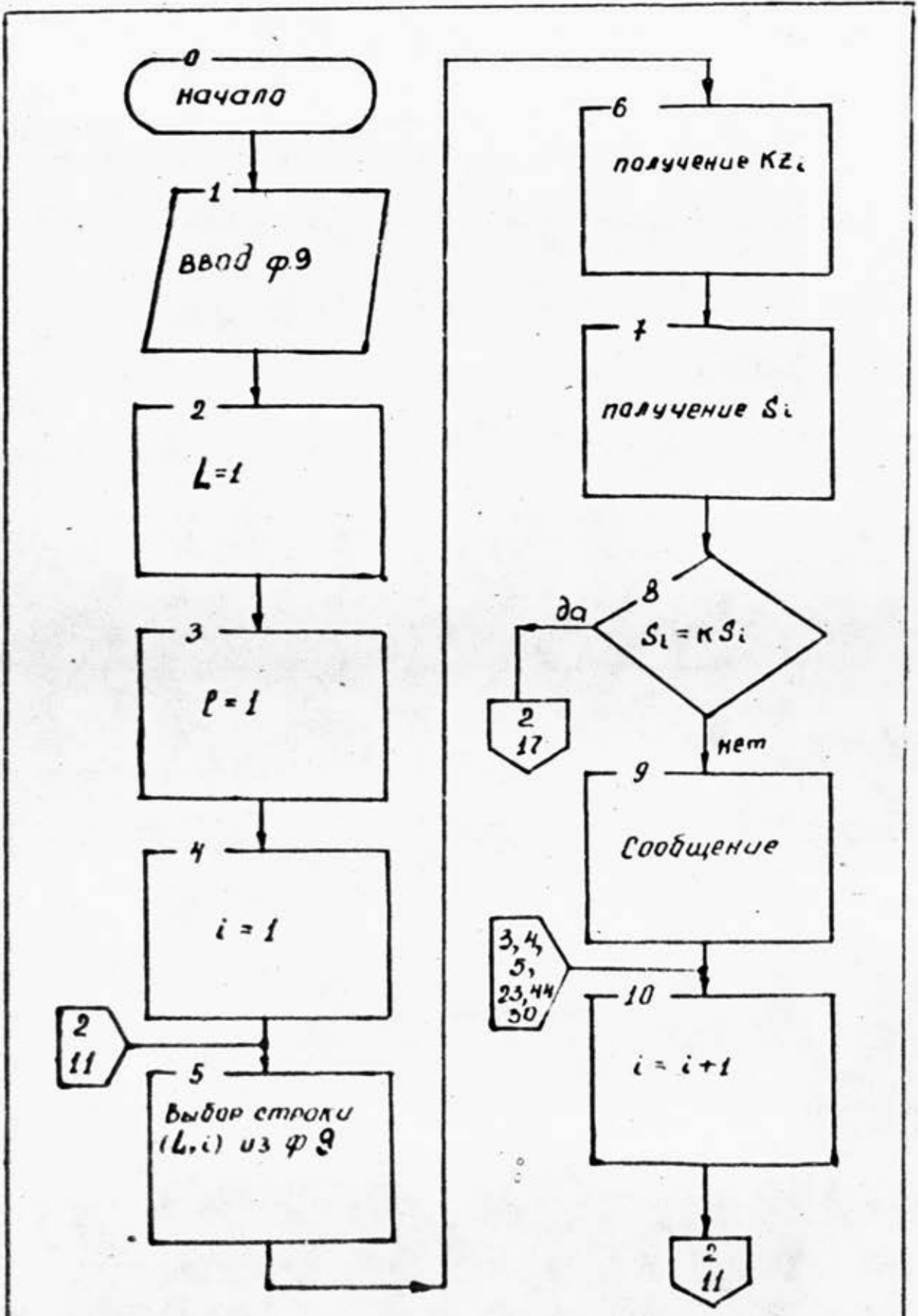


Рис. 3.3.5

(Лист 1)

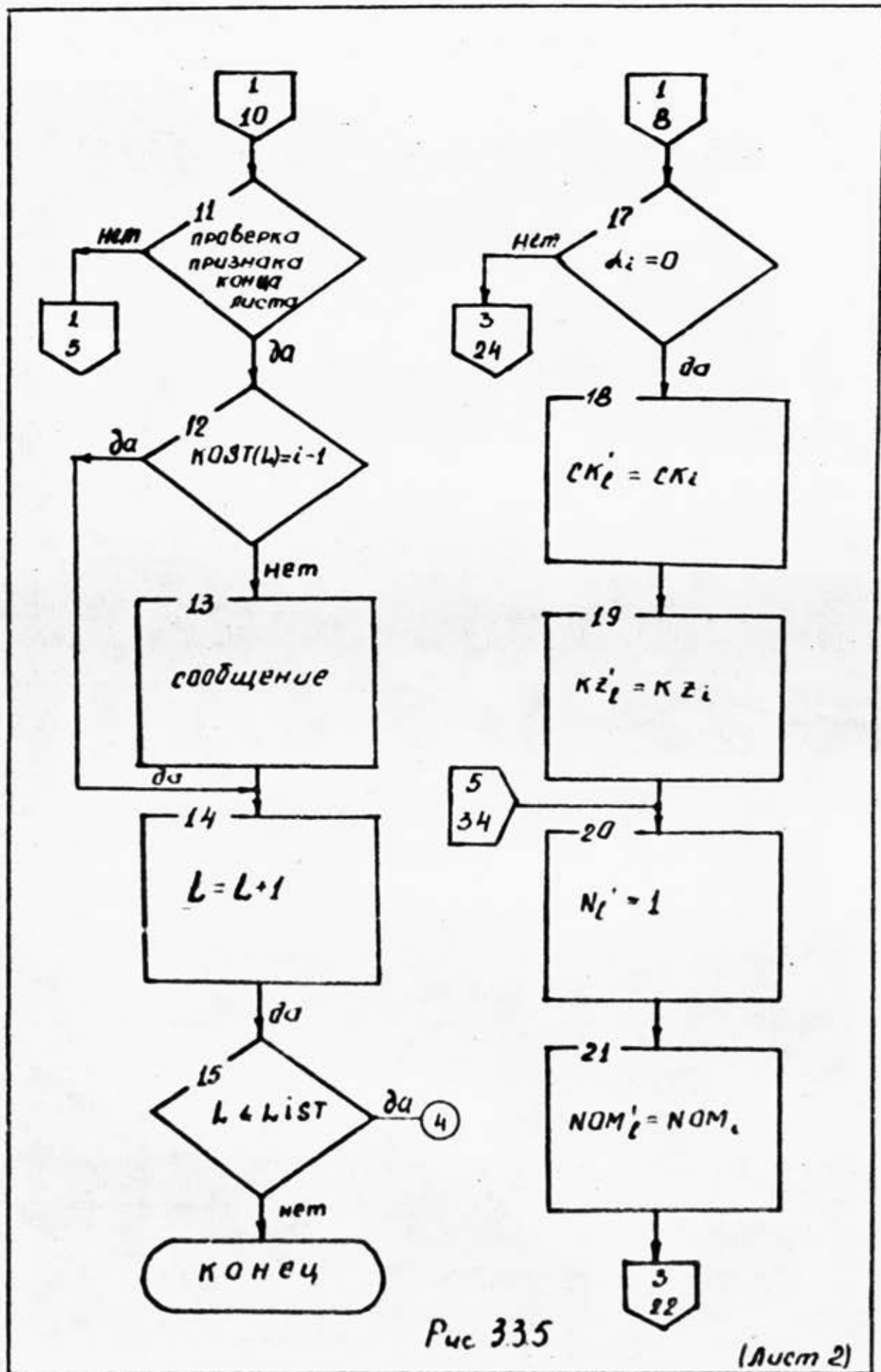


Рис 335

(Лист 2)

УБ-2-78

ТПЗЕН2

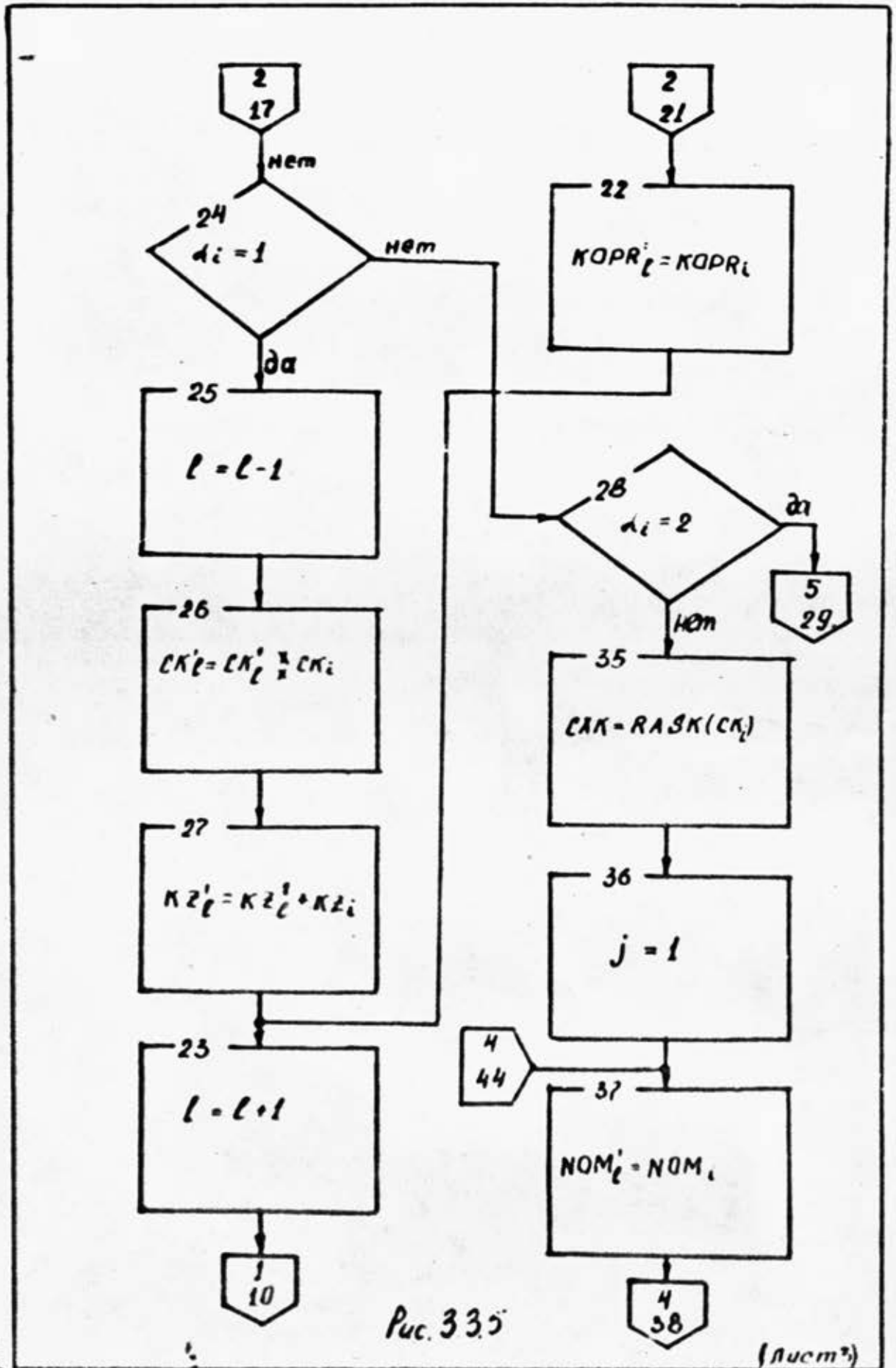


Рис. 335

(Лист 3)

753-36
ТПЗВН2

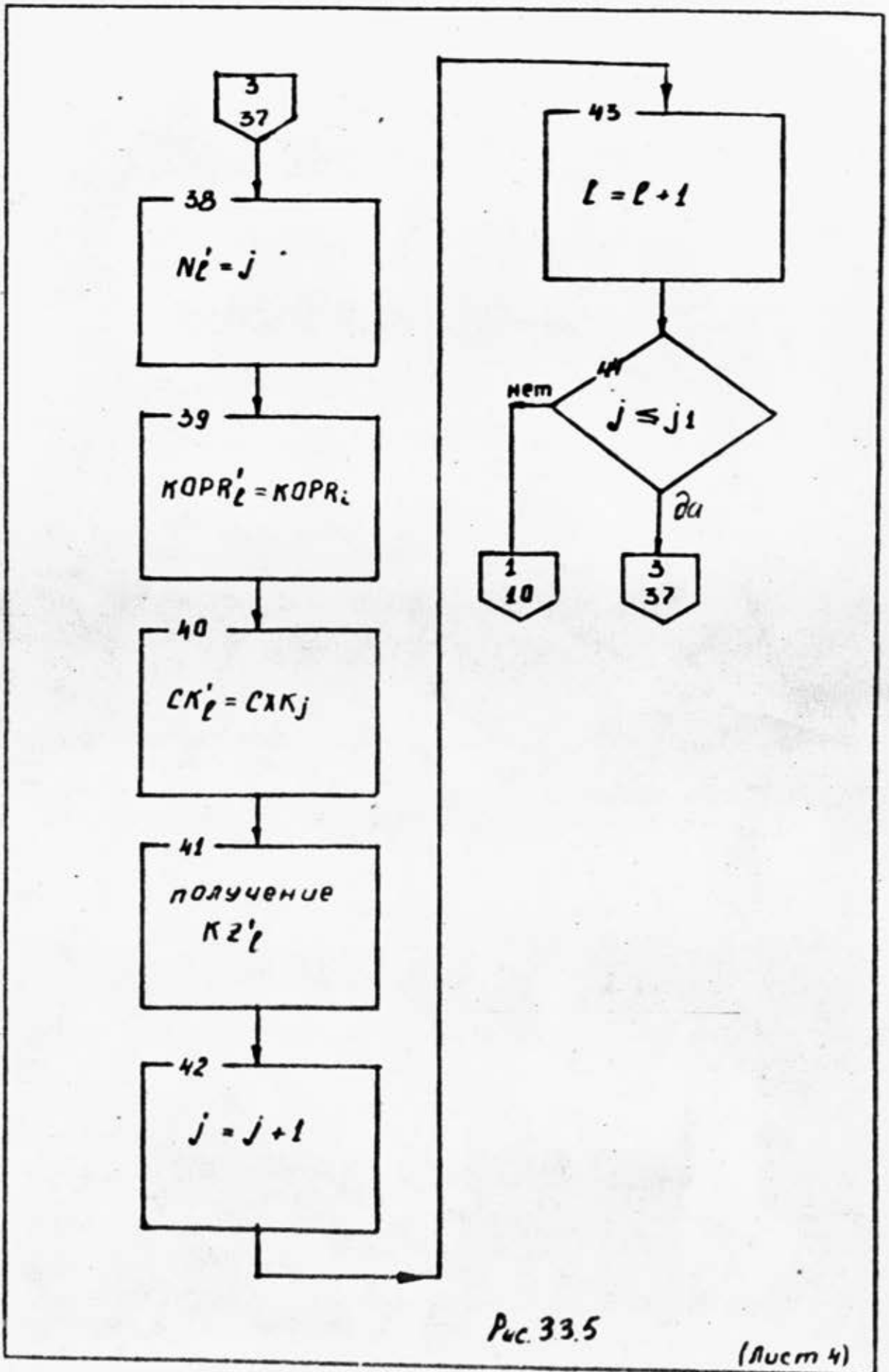


Рис. 33.5

(Лист 4)

15-2-96
ТПТЗКМ2

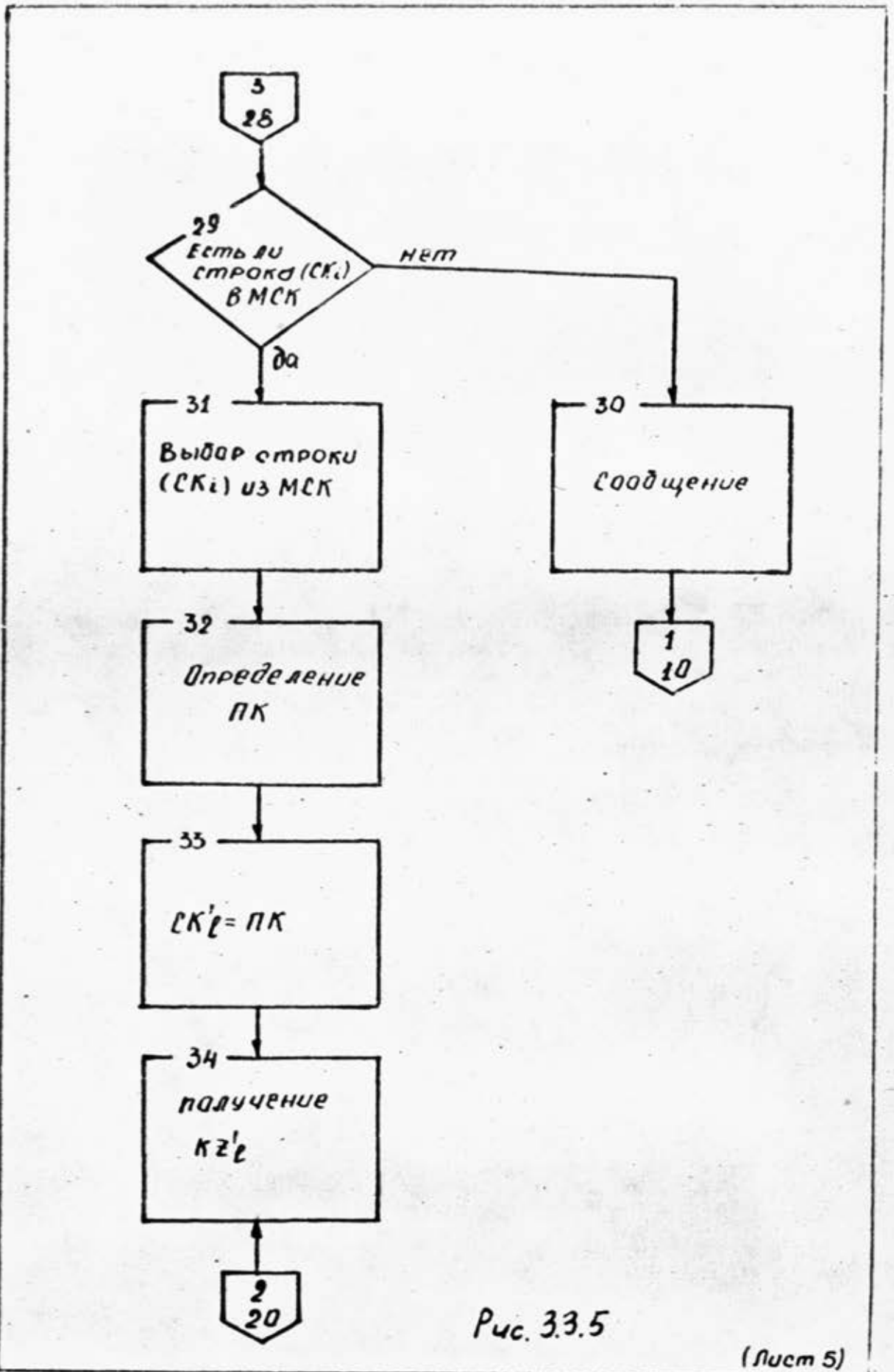


Рис. 33.5

15-2-74

ТТЗЕН2

$$d = \begin{cases} 0, & \text{если } SK_i - \text{ полный схемный код;} \\ 1, & \text{если } SK_i - \text{ продолжение полного схемного} \\ & \text{кода;} \\ 2, & \text{если } SK_i - \text{ краткий код;} \\ 3, & \text{если } SK_i - \text{ обобщенный код} \end{cases}$$

KS - контрольная сумма (для проверки правильности исходной информации);

KZ, KZ' - количество звеньев в схемном коде;

N' - порядковый номер схемного кода;

MCK - массив сокращенных схемных кодов;

PK - полный схемный код в MCK ;

$RASK(SK)$ - модуль расшифровки обобщенного схемного кода SK ;

SXK - множество j схемных кодов, полученных в результате расшифровки обобщенного схемного кода.

В блок-схеме приведены следующие блоки:

Бл.6. Количество звеньев в схемном коде SK_i определяется подсчетом направленных стрелок в коде.

Бл.7. Получение контрольной суммы строки $(L, i) - S_i$.

Бл.9. Выдача сообщения в случае, когда контрольная сумма S_i не совпадает с заданной суммой KS_i строки (L, i) Ф.9.

Бл.13. Выдача сообщения в случае, когда количество просмотренных строк в листе Ф.9 не совпадает с заданным количеством строк листа.

Бл.18-22, 26, 27. Формирование e -ой строки записи результата.

Бл.30. Выдача сообщения в случае, когда строка MCK , соответствующая сокращенному схемному коду SK_i , отсутствует.

Бл.31. Выбор строки MCK , соответствующей сокращенному схемному коду SK_i .

Бл.32. Получение полного схемного кода PK из строки MCK , соответствующей сокращенному схемному коду SK_i .

Бл.33. Формирование схемного кода e -й строки SK_e записи результата.

Бл.34. Определение количества звеньев в полном схемном коде PK строки MCK и формирование величины $KZ'e$ -ой строки записи результата.

Бл.35. Получение множества схемных кодов с помощью модуля

расшифровки обобщенного схемного кода СК.

Бл.37-41. Формирование l -ой строки записи результата.

В блоке 41 величина KZ'_e формируется как количество звеньев в схемном коде $СХХ$.

15-2-74

ТПТЗм2

3.3.6. Модуль перевода $0000II \rightarrow \alpha_2 0III$

Модуль предназначен для перевода конститuent из представления $0000II$ в представление $\alpha_2 0III$.

Исходными данными для работы модуля являются:

- MKI - множество конститuent, подлежащих переводу;
- массив индивидуальных представлений конститuent ИИД;
- значение второго знака в коде представлений конститuent;
- номер вершины, соответствующей роду структуры, содержащему конститuent.

В результате работы модуля получаем множество $MK2$ из MKI заменой стандартных имен конститuent индивидуальными именами ($\alpha_2 = 0$) или терминами ($\alpha_2 = I$) в случае, если они найдены в массиве ИИД.

Блок-схема модуля представлена на рис. 3.3.6.

В блок-схеме приняты следующие обозначения.

N_6 - номер вершины, соответствующей роду структуры, содержащему рассматриваемые конститuent.

MKI - множество конститuent, подлежащих переводу.

N - число элементов множества MKI .

NW_{ii} - номер вершины, соответствующей i -той строке массива ИИД.

$STMK_i$ - стандартное имя конститuent, соответствующей i -той строке ИИД.

$ini(i)$ - конститuenta, соответствующая i -той строке ИИД, в представлении $000III$.

$IN(i)$ - конститuenta, соответствующая i -той строке ИИД, в представлении $0I0III$.

α_2 - значение второго знака в коде представления конститuent (α_2 принимает значение 0 или I)

$MK2$ - множество, получаемое из MKI заменой стандартных имен конститuent индивидуальными именами ($\alpha_2 \neq 0$) или терминами ($\alpha_2 = I$) в случае, если они найдены в массиве ИИД.

В блок-схеме приведены следующие блоки.

Блок 6. Выбор очередной строки (i) массива индивидуальных представлений ИИД.

Бл. I4. Формирование j -того элемента множества $MK2$ заменой стандартного имени конститuent индивидуальным именем.

Бл. 15. Формирование j -того элемента множества МК2 заменой стандартного имени конstituэнты индивидуальным термином.

15.2.20

ТТК.2.202

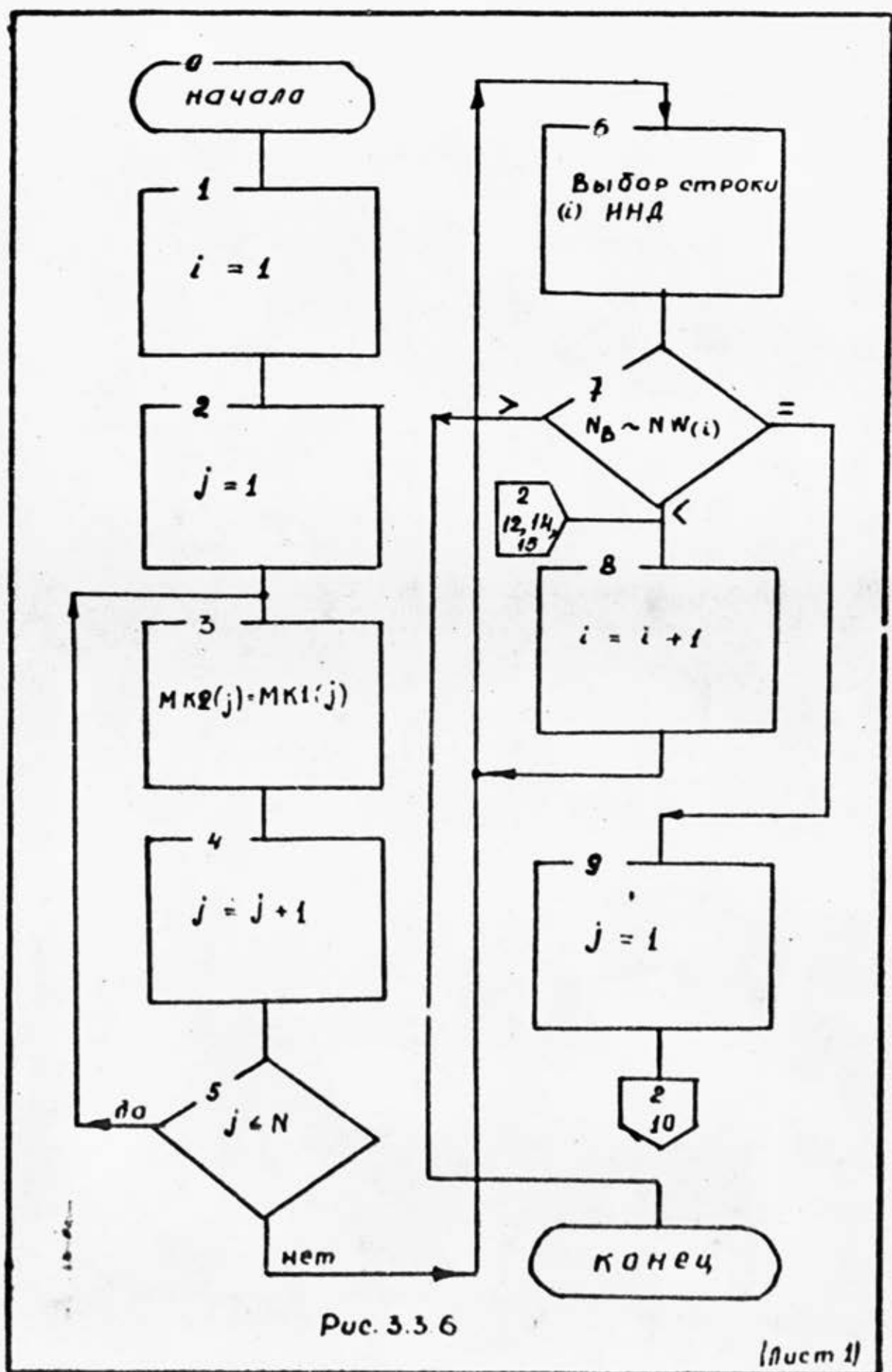


Рис. 3.3.6

(лист 1)

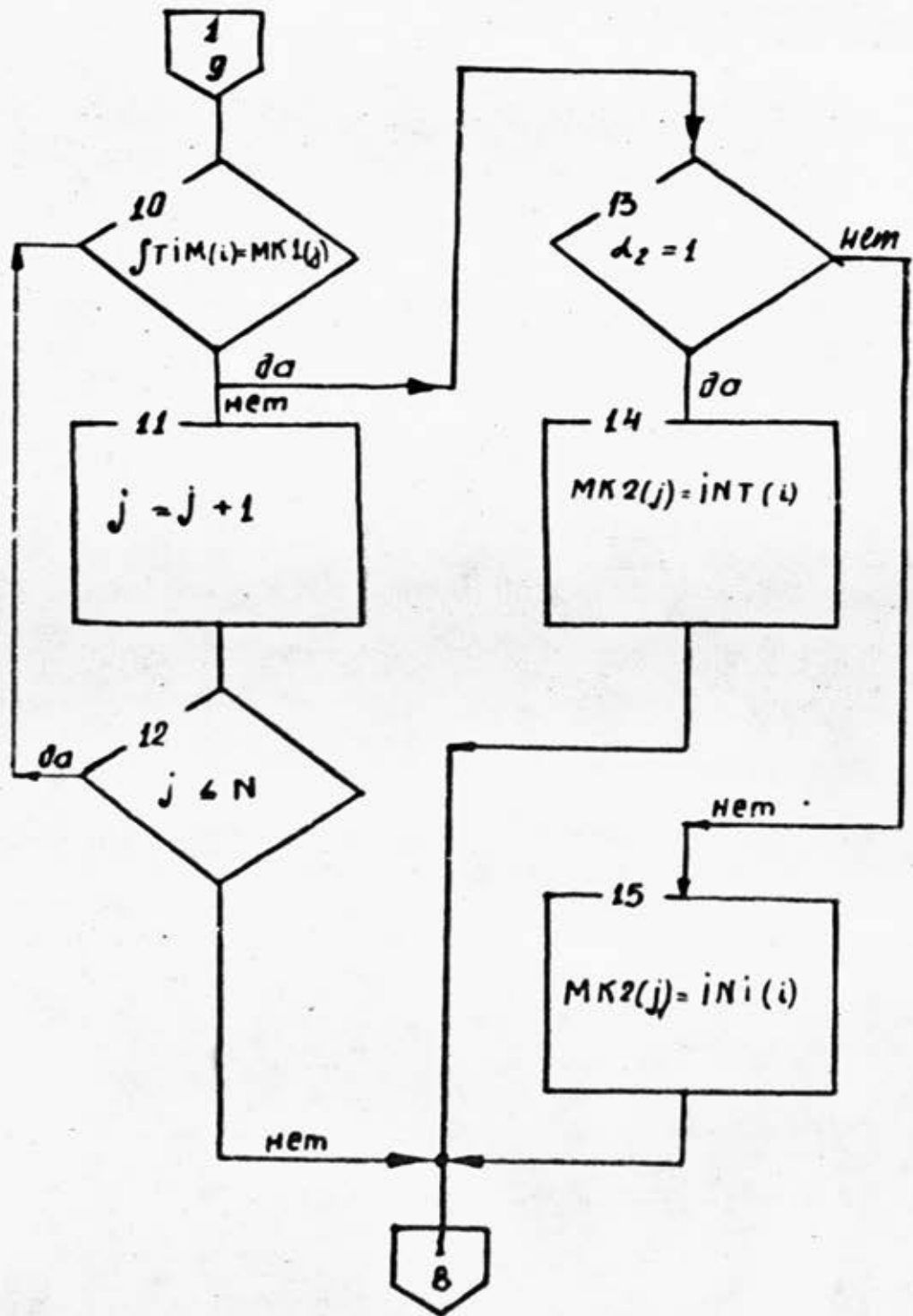


Рис. 3.3.6

3.3.7. Модуль перевода $0000II \rightarrow 100III$.

Модуль предназначен для замены в выражении конститuent стандартных имен конститuent их индивидуальными именами.

Исходными данными для работы модуля являются:

- $MN1$ - множество конститuent, в выражении которых необходимо произвести замены,
- номер вершины, соответствующей роду структуры, содержащему конститuent из $MN1$.

В результате работы модуля получаем множество выражений конститuent $MN1$, в которых произведена замена стандартных имен конститuent их индивидуальными именами.

Блок-схема модуля представлена на рис. 3.3.7.

В блок-схеме приняты следующие обозначения:

$MN1$ - множество конститuent, в выражении которых необходимо произвести замены стандартных имен конститuent их индивидуальными именами.

M - число элементов множества $MN1$

N_6 - номер вершины, соответствующей роду структуры, содержащему конститuent множества $MN1$.

α_2 - значение второго знака в коде представления конститuent множества $MN1$.

$\overline{MN1}$ - множество выражений конститuent $MN1$.

$MK1$ - множество конститuent, входящих в выражения множества $\overline{MN1}$.

$MK2$ - множество конститuent, полученных в результате перевода конститuent множества $MK1$ по модуль перевода $0000II \rightarrow 0\alpha_20III$.

$MOD1$ - модуль перевода $0000II \rightarrow 0\alpha_20III$.

$WJDW(k,B)$ - подпрограмма выделения выражения конститuent C из элемента B записи рода структуры.

$WJDK$ (выражение) - подпрограмма выделения множества конститuent, входящих в выражение.

ZAM (выражение) - подпрограмма, осуществляющая замену конститuent в выражении их индивидуальными именами в случае, если они указаны в $MK2$.

$MN2$ - множество выражений, полученных из $\overline{MN1}$ заменой стандартных имен конститuent их индивидуальными именами, если они указаны в $MK2$.

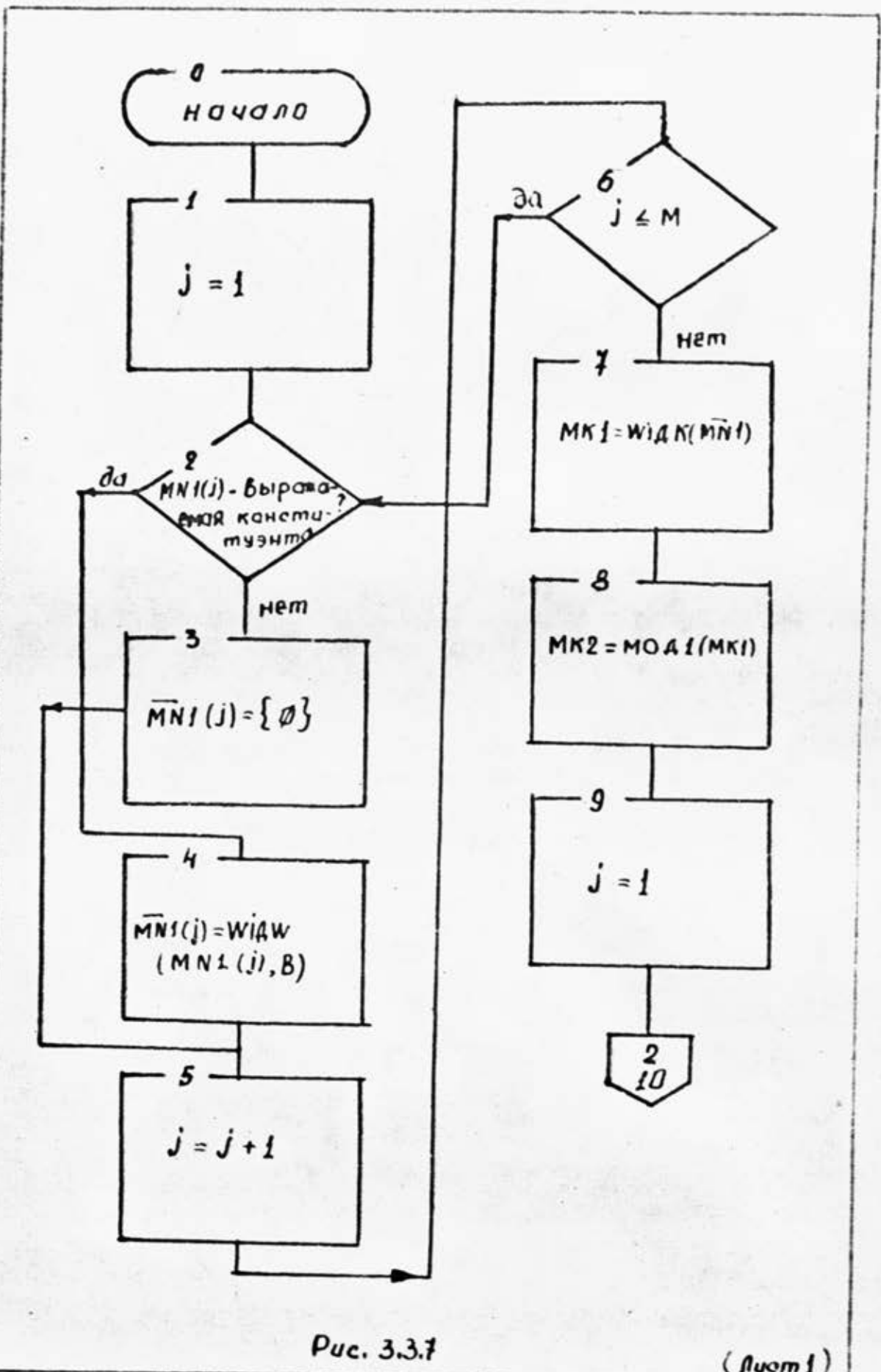
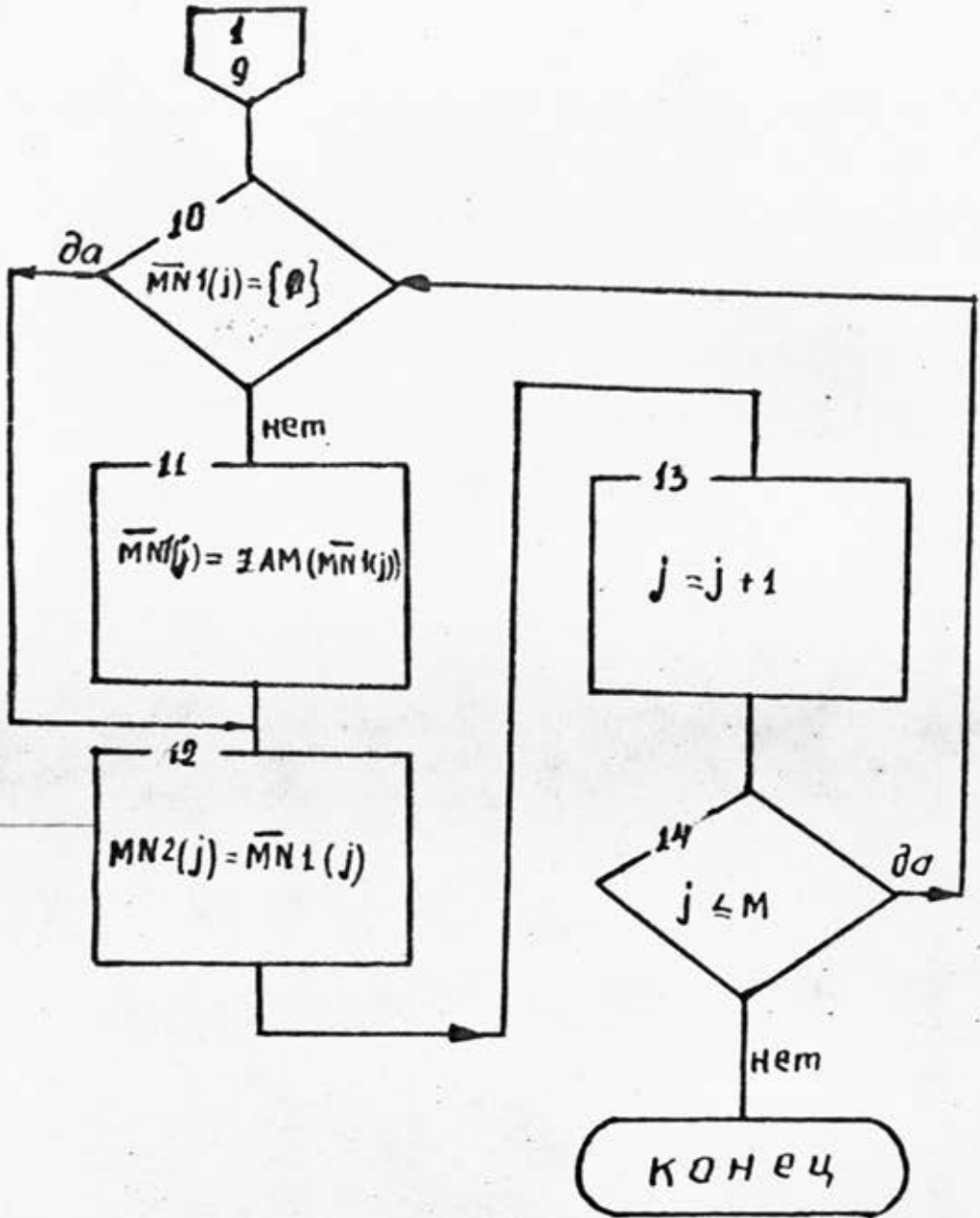


Рис. 3.3.7

(Алгоритм 1)

45-2-76
ТПЗен2



15-2-76
ГПТЗКМ2

Рис. 3.3.7. (Лист 2)

В блок-схеме приведены следующие блоки:

Бл. 03. Формирование выражений для невыражаемых конститут-
энт.

Бл. 04. Определение выражения конститутэнт множества
 $MN1$ с помощью специальной подпрограммы $WODW(C, B)$.

Исходные данные для работы подпрограммы:

- конститутэнт $MN1(j)$;
- элемент B записи рода структуры, соответствующей номе-
ру вершины N_6 .

Подпрограмма определяет класс конститутэнт $MN1(j)$, вы-
деляет из элемента B записи рода структуры запись выражений
конститутэнт этого класса (W) и из выделенной записи выражений
 W определяет выражение конститутэнт $MN1(j)$.

В результате работы подпрограммы получаем выражение кон-
ститутэнт $MN1(j)$.

Бл. 07. Получение множества конститутэнт $MK1$ с помощью
подпрограммы $WJDK$ (выражение).

Исходные данные для работы подпрограммы:

$MN1$ - множество выражений конститутэнт $MN1$.

Подпрограмма выделяет множество различных конститутэнт,
входящих в выражения множества $MN1$.

В результате работы подпрограммы получаем множество кон-
ститутэнт $MK1$.

Бл. 08. Получение множества конститутэнт $MK2$ с помощью мо-
дуля перевода $MOD1$.

Исходными данными для работы модуля являются:

- множество конститутэнт $MK1$.
- N_6 - номер вершины.
- d_2 - значение второго знака в коде представления конститут-
энт ($d_2=0$).

Бл. II. Формирование множества $\overline{MN1}$ с помощью специальной
подпрограммы ZAM (выражение).

Исходными данными для работы подпрограммы являются:

- $MK1$ - множество стандартных имен конститутэнт.
- $MK2$ - множество индивидуальных имен конститутэнт, соот-
ветствующих конститутэнтам множества $MK1$.

- $\overline{MN1(j)}$ - выражение конститутэнт $MN1(j)$.

Подпрограмма находит в выражении $\overline{MN1(j)}$ конститутэнт,
отыскивает их в множестве конститутэнт $MK1$, по формуле соот-

ветствия

$МК1(j) \rightarrow МК2(j)$

определяет индивидуальные имя конститuentы и производит замену стандартных имен конститuent в выражении $MN(j)$ на их индивидуальные имена.

В результате работы подпрограммы в выражении $\overline{MN(j)}$ осуществляется замена конститuent их индивидуальными именами.

Бл. 12. Формирование выходного массива $MN2$.

3.3.8. Модуль нахождения базового индекса.

Модуль предназначен для нахождения базового индекса.

Исходными данными для работы модуля являются:

- номер вершины N_6 ;
- операционная схема ОПСХ;
- массив ЭТА (см. описание 3.3.).

Подпрограмма определяет значение базового индекса в случае, если N_6 является начальной вершиной операционной схемы (ОПСХ).

В результате работы модуля получаем значение базового индекса.

В блок-схеме приняты следующие обозначения:

S_0 - нулевой слой ОПСХ.

N_6 - номер вершины ОПСХ.

STB - элемент записи ЭТА (стандартная буква).

N_0 - элемент записи ЭТА (значение базового индекса).

V_i - значение базового индекса.

В блок-схеме приведены следующие блоки.

Бл.01. Выбор нулевого слоя операционной схемы.

Бл.03. Выдача сообщения в случае, если в нулевом слое отсутствует вершина с номером N_6 .

Бл.05. Выдача сообщения в случае, если массив ЭТА не содержит строки, соответствующей номеру вершины N_6 .

Бл.08. Выдача сообщения в случае, если в найденной строке ЭТА элемент STB определяет отождествляющее отображение.

Бл.06. Выбор строки ЭТА, соответствующего номеру вершины N_6 .

Бл.09. Определение базового индекса.

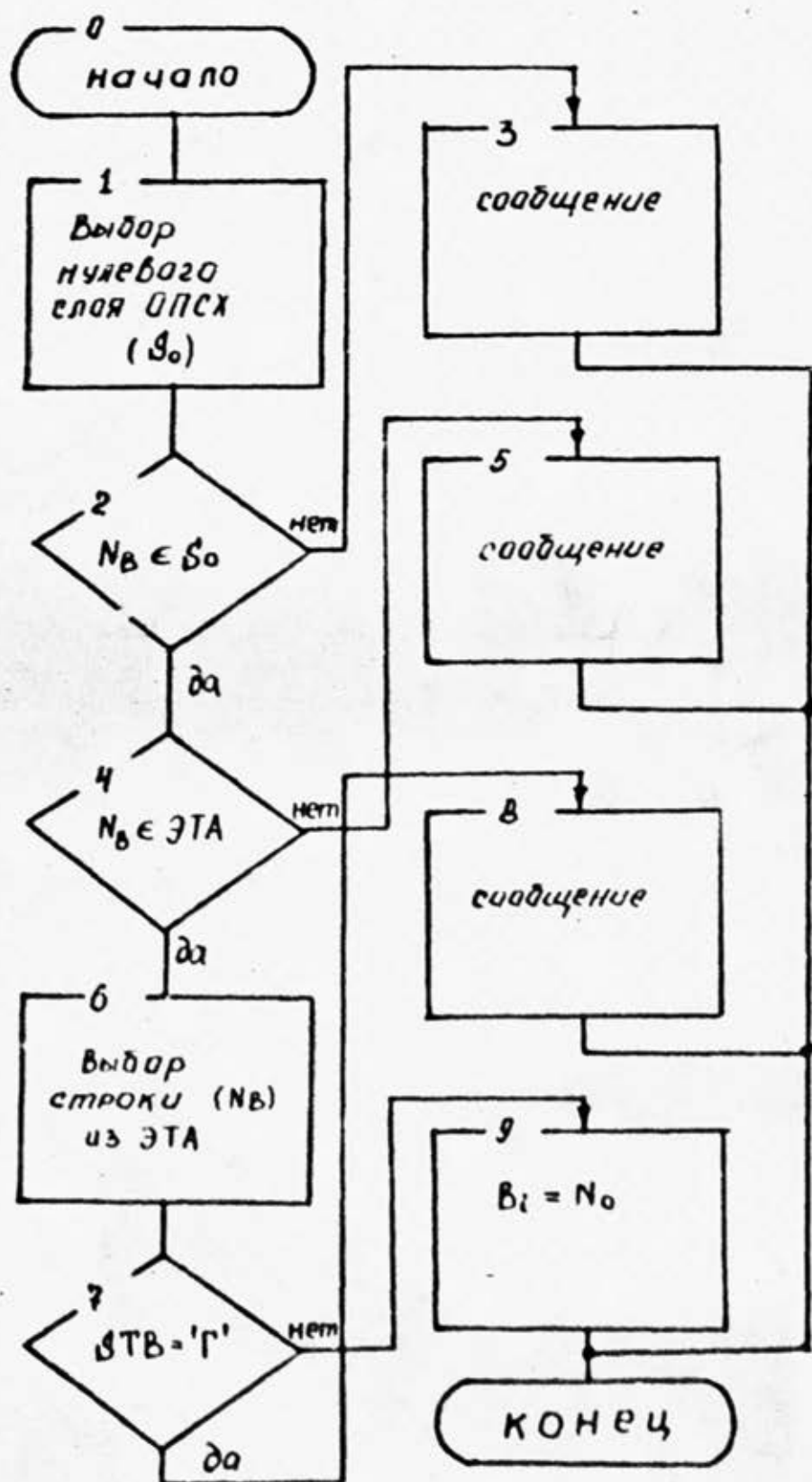


Рис. 3.3.8

3.4. Характеристики информационных массивов.

ППП перевода представлений и связь с другими ППП.

Пакет прикладных программ перевода представления использует следующие массивы:

- 1) массив ИНД- массив индивидуальных терминов и обозначений (БД),
- 2) массив МСК- массив сокращенных схемных кодов (БД),
- 3) массив МТПП- массив требования на перевод представления (промежуточный),
- 4) массив ППК- массив, полученный в результате перевода конститuent (выходной).

Ниже будет дано описание каждого из этих массивов.

3.4.1. Массив ИНД.

1) Имя.

Имя массива, употребляемое в программах, должно содержать идентификаторы проекта и варианта.

2) Информация, содержащаяся в записи массива.

Каждая запись массива соответствует одной конститuentе рода структуры или дополнения и задает индивидуальные имя и термин этой конститuentы.

3) Организация массива.

Массив ИНД имеет записи неопределенной длины, которые упорядочены по номерам вершин и по номерам конститuent.

4) Метод доступа- прямой или индексно-последовательный.

5) Максимальный размер - 5000 записей.

6) Структура записи массива ИНД показана на рис. 3.4.1.

Номер вершины	Стандартн. имя	Индивидуаль- ное имя	Индивиду- альный термин	Признак конца записи
------------------	-------------------	-------------------------	-------------------------------	----------------------------

Рис. 3.4.1.

7) Получение массива ИНД.

Массив ИНД вносится в БД (или пополняется) после следующих действий:

- ввода формы 7 в КОЗУ,
- проверки правильности перфорации,
- проверки того, что вершины, указанные в форме, принад-

лежат операционной схеме, а конститuentы принадлежат родам структур и дополнениям, соответствующим этим вершинам.

- проверки того, нет ли уже в ИНД записи, соответствующей вводимой конститuentе (в случае пополнения ИНД).

8) Использование массива ИНД в ППП перевода представлений.

Массив ИНД используется пакетом перевода представлений всякий раз, когда в задании на перевод представлений присутствует код представления, в котором $\alpha_4 = I$.

9) Использование массива ИНД в других ППП.

Массив ИНД используется ППП внесения изменений при внесении изменений в этот массив.

3.4.2. Массив МСК.

1) Имя.

Имя массива, употребляемое в программах, должно содержать идентификаторы проекта и варианта.

2) Информация, содержащаяся в записи массива.

Каждая запись массива соответствует одному сокращенному схемному коду и содержит его расшифровку.

3) Организация массива.

Массив МСК имеет записи переменной длины.

4) Метод доступа - прямой и индексно-последовательный.

5) Максимальный размер - 1000 записей.

6) Структура записи массива МСК показана на рис. 3.4.2.

Сокращенный схемный код	: Кол-во звеньев Z	:	Полный схемный код N ₀ : <u>и/и</u> : M ₁ : M ₁ : <u>и/и</u> : M ₂ : ... : <u>и/и</u> : M ₂	:	N ₂
-------------------------------	--------------------------	---	---	---	----------------

Рис. 3.4.2.

В каждой записи массива выделены следующие поля:
Сокращенный схемный код - условное обозначение полного схемного кода (поле фиксированной длины);

Количество звеньев - количество звеньев в полном схемном коде (количество стрелок).

N_0, N_1, N_2 - номера вершин схемного кода

M_1, \dots, M_2 - номера аргументов схемного кода.

7) Получение массива МСК.

Массив МСК вносится в БД (или пополняется) после следую-

ших действий:

- ввода формы в МСЗУ,
- проверки правильности перфорации,
- проверки того, что вводимые сокращенные схемные коды отсутствуют в массиве МСК (при пополнении),
- проверки правильности полного схемного кода.

8) Использование массива МСК в ППП перевода представлений.

Массив МСК используется пакетом перевода представлений всякий раз, когда в задании на перевод представлений присутствует сокращенный схемный код.

9) Использование массива МСК в других ППП.

Массив МСК используется ППП внесения изменений при внесении изменений в этот массив.

3.4.3. Массив МТПП.

1) Имя.

Массив МТПП является внутренним по отношению к ППП (он вводится в БД). Поэтому специальные требования к имени массива не выдвигаются.

2) Информация, содержащаяся в записи массива.

Каждая запись соответствует одному требованию на перевод представлений.

3) Организация массива.

Массив МТПП имеет записи переменной длины.

4) Метод доступа-последовательный.

5) Максимальный размер-500 записей.

6) Структура записи массива МТПП показана на рис. 3.4.3.

Номер требования	Номер схемн. кода	Код представления	Кол-во звеньев	Полный схем- ный код
---------------------	-------------------------	----------------------	-------------------	-------------------------

Рис. 3.4.3.

В каждой записи массива выделены следующие поля:

Номер требования - номер требования в форме 9,

Номер схемного кода - порядковый номер схемного кода в требовании;

Код представления - код представления конститутанты-ре-

зультата операции.

Кол-во звеньев- количество звеньев в полном схемном коде.

Полный схемный код- полный схемный код, по которому должна производиться Т или антиинтерпретация конституэнт.

7) Получение массива МТПП.

Массив получается ~~после~~ после ввода формы 9 и замены во вводимых требованиях сокращенных схемных кодов полными, а обобщенных схемных кодов- наборами полных схемных кодов.

8) Использование массива МТПП в ППП переводов представлений.

Массив МТПП используется пакетом как задание на работу.

3.4.4. Массив ППК.

1) Имя.

Массив ППК является внутренним по отношению к ППП (он не вводится в БД). Поэтому специальные требования к имени массива не выдвигаются.

2) Информация, содержащаяся в записи массива.

Каждая запись соответствует одной конституэнте, переведенной по одному схемному коду одного требования, и содержит требуемое представление этой конституэнты.

3) Организация массива.

Массив ППК имеет записи неопределенной длины.

4) Метод доступа- последовательный.

5) Максимальный размер-50000 записей.

6) Структура записи массива ППК показана на рис.3.4.4.



Рис.3.4.4.

В каждой записи массива выделены следующие поля:
Номер требования-и номер схемного кода- переносятся из МТПП.

Стандартное имя- имя конституэнты, подвергаемой перево-
ду.

Номер вершины- номер вершины, соответствующей результа-

15-2-26
ТПЗМ2

ту перевода.

Базисный индекс- базисный индекс рода структуры или дополнения, содержащего результат перевода.

Простое внутреннее представление- простое внутреннее представление конститутэнты- результата перевода.

7) Получение массива ППК.

Массив ППК получается в результате работы ППП перевода представлений по обработке массива ИТПП.

8) Использование массива ППК в ЦПП перевода представлений.

Массив ППК является выходным для пакета. По нему происходит распечатка формы IO.

15-2-76

Т.П.Зин2

3.5. Ограничения и возможности расширения ППП.

При программировании ППП перевода представлений необходимо учитывать следующие ограничения:

- 1) максимальное число индивидуальных имен или терминов не превосходит 5000 ;
- 2) максимальное число сокращений схемных кодов не превосходит 1000 ;
- 3) максимальное число расшифрованных требований на перевод представлений (записей в МПП) не превосходит 500.

К функциональным ограничениям ППП внесения изменений, которые предусмотрены на данном этапе, относятся ограничения, описанные в п.3.1. данного документа. В соответствии с ними результат перевода конституэнт не может быть.

- 1) выражением, содержащим термины конституэнт ;
- 2) схемным обозначением ;
- 3) стандартным термином.

Кроме того, целесообразно получать полные представления (с указанием номера вершины, соответствующей результату перевода) и поэтому требуется, чтобы в коде представления результата $\alpha_j = I$.

Расширение возможностей ППП перевода представлений может производиться в направлении постепенного избавления от указанных выше функциональных ограничений, а также ослабления количественных ограничений (длины массивов и др.).

Программирование ППП перевода представлений следует организовать так, чтобы указанные выше возможности расширения пакета были осуществимы без большого объема пере.^{РАБОТКИ}

3.6. Средства программного контроля и контроля результатов.

Контроль информации, которой оперирует ШП переводов представлений, осуществляется в два этапа:

- 1) контроль форм 7 и 8 при вводе информации в БД;
- 2) контроль формы 9 при получении задания на перевод представлений.

Первый этап контроля производится программными средствами СУБД^{*}). В него входит контроль перфорации, проверка отсутствия вводимых индивидуальных имен и терминов, а также сокращенных кодов в соответствующих массивах, проверка правильности задания полных схемных кодов и т.д.

Второй этап контроля производится в самом пакете. При этом контролируется правильность перфорации формы 9, правильность задания требований на перевод представлений и т.д.

^{*} См. п.5 документа "Описание применения системы ШП".

4. Перепроектирование и внесение изменений. (ППП-внесения изменений).

4. I. Постановка задачи (содержание и метод решения).

Процесс машинного перепроектирования заключается в формировании измененной части проекта согласно изменениям, внесенным в основные входы машинных операций.

По "исходным" изменениям ЭВМ выдает проектировщику объем необходимых "производных" изменений и справочную информацию об операциях их получения. В результате принимается решение о частичном или полном перепроектировании. В случае частичного перепроектирования ЭВМ формирует лишь измененную часть проекта.

Основная часть получения проекта является результатом работы ППП формирования ГРС, R-интерпретации, перевода представлений T-и антиинтерпретации и формирования текста проекта. Таким образом, частичное перепроектирование заключается в следующем:

- внесение изменений во входную информацию вышеуказанных пакетов;
- перенос и сохранение неизменяющейся части проекта;
- выдача сведений проектировщику об объемах "производных" изменений;
- получение изменяющейся части проекта.

Необходимо предусмотреть возможности внесения изменений не только перед началом работы соответствующего пакета, но и во время его работы. В связи с этим следует определить в каких точках работы пакета разрешено внесение изменений.

Согласование вносимых изменений производится проектировщиком по информации, получаемой с помощью сервисных средств системы, результатов работы ППП перевода представлений, T и антиинтерпретации. Дополнительная информация о взаимосвязях констатуют должна вырабатываться при внесении изменений и сообщается проектировщику по запросу.

Внесение изменений должно производиться таким образом, чтобы пакеты могли работать с измененной информацией, как с первоначальной. В частности, должна быть реализована возможность вносить изменения и в измененную информацию в той же

или последующих точках работы соответствующего пакета, разрешенных для внесения изменений.

Внесение изменений должно охватывать:

1. В пакете формирования ГРС:

- операционную схему,
- базовые роли структур, дополнения, отображения

2. В пакете R -интерпретации:

- граф конституэнт,
- граф R -интерпретации,
- R -интерпретацию конституэнт,

3. В пакете перевода представлений:

- индивидуальные обозначения и термины,
- сокращенные коды T-интерпретации.

Метод решения. Решение поставленной задачи производится пакетом внесения изменения (ПВИ). Пакет работает под управлением ГУП и частично использует его средства. При получении задания на внесение изменения, ГУП прерывает работу соответствующего пакета в ближайшей следующей точке, в которой разрешено внесение изменений и вызывает ПВИ. Чтобы реализовать возможность отказа от внесенных изменений, используются средства ГУП по созданию контрольных точек. Перед внесением изменений копируется информация, которая может подвергнуться изменениям. Это позволяет возвращаться к предыдущему варианту или подварианту. Проектировщик может отменить создание контрольной точки в данном месте работы, тогда он теряет возможность отказаться от сделанного изменения.

За один сеанс работы ПВИ может последовательно обработать несколько заданий, причем разрешается вносить все изменения, имеющие смысл для данного пакета и места его работы.

При внесении изменений производится анализ на правильность вносимых изменений в объеме, соответствующем контролю входной информации пакетов. ПВИ реализует изменения в объеме, указанном в постановке задачи. Изменение каждого объекта допускается одним из трех способов:

- расширение,
- сокращение,
- замена.

Таким образом, основные функции, выполняемые при реше-

нии задачи перепроектирования следующие:

- изменение операционной схемы;
- изменение базовых родов структур, дополнений и отображений;
- изменение графа R-интерпретации;
- изменение значений R-интерпретационных конститuent;
- изменение индивидуальных обозначений, терминов, сокращенных кодов T-интерпретации;
- изменение уровней, имен разделов проекта и распределения конститuent по разделам проекта;
- фиксация состояния системы до внесения изменений,
- определение объемов перепроектирования.

15-2-76
ТПТЗел2

4.2. Алгоритм управления функционированием ППП (УПП).

Модуль УПП предназначен для управления процессом внесения изменений.

Модуль вызывается главной управляющей программой системы ППП (ГУП) при поступлении запроса ^{на} внесение изменений. Модуль (значит и пакет БИ) может вызываться только при определенном состоянии пакетов, в информацию которых вносятся изменения. Эти состояния называются точками возможного *внесения* изменений.

Для пакета формирования ГРС точками возможного внесения изменений являются моменты времени, когда УП пакета формирования ГРС начинает анализ для выбора операции.

Для пакета R - интерпретации точками возможного внесения изменения являются:

- момент времени, когда закончил свою работу модуль формирования нижнего замыкания;
- момент времени, когда закончил свою работу модуль удаления подграфа;
- момент времени, когда закончил свою работу модуль удаления дублирующих подграфов;
- момент времени когда закончена R - интерпретация конститuenty.

Для пакета перевода представления точками возможного внесения изменений являются моменты времени, когда закончена обработка строки задания формы 9.

При обращении к пакету БИ указывается:

обязательная информация:

- пакет, в информации которого вносятся изменения;
- тип вносимых изменений (в одно задание может входить несколько типов);

и необязательная информация - значения ключей.

Либо вместо обязательной и необязательной информации - служебная (для изменения значения ключей,

подразумеваемых по умолчанию), см. описание блока 28.

Блок-схема алгоритма модуля приведена на рис. 4.2.1.

В блок-схеме использованы следующие обозначения:

ТИЗМ 1 - таблица см. рис. 4.2.2.

№	наименование	адрес 1	адрес 2
1.	ГРС	блок 13	блок 3
2.	<i>R</i> - интерпретация	блок 24	блок 14
3.	Перевод представления	блок 28	Блок 25
4.	Изменение карчей	блок 29	блок 30

Рис. 4.2.2.

Адрес 1 указывает, куда нужно передать управление если заданное изменение совпало с наименованием в этой строке.

Адрес 2 указывает, куда нужно передать управление если заданное изменение не совпало с наименованием в этой строке.

ТИЗМ 2

ТИЗМ 3

ТИЗМ 4

Таблицы, имеющие структуру аналогичную структуре ТИЗМ 1. В графе наименования этих таблиц перечислены имена модулей, производящих изменение, соответственно:

ТИЗМ 2 - РОПСХ,

- СОПСХ,

- ЗОП,

- ИБРС,

- ЗИК,

- ИОП,

ТИЗМ 3 - *Pr el*,

- *Cr el*,

- РАК,

- РСК,

- РРТ,

- СРТ.

- ИОЗ.
- ТИЗМ 4 - ИССК,
- ИЗИНИМ.

Полное наименование модулей см. 4.3

ТВЗ, ТРВЗ - см. 4.3.9.

КОПС, ТИИТ, - имена массивов (подробнее о содержаниях массивов см. 4.4.).

ЗНАЧР, КРВГ.

- УЗ - список, удаляемых записей, его состав см. блок 18.
- УЗ1 - список, включающий УЗ и часть вершин из ТРВЗ, получивших R - интерпретацию.
- ПЗ - Правильная часть задания на работу модуля. Она определяется в каждом модуле см. 4.3.

Описание отдельных блоков алгоритма.

Блоки 2, 14, 24, 28.

В блоках анализируется совпадение запроса на изменение ^{с одним} U_3 полей "наименование" ТИЗМ I.

Блок 3 Проверяется, что до пакета ВИ работала пакет формирования ГРС.

Блок 5 Имя модуля, реализующего требуемое изменение, выбирается из списка ТИЗМ 2. Это делается также, как и для ТИЗМ I.

Блок 7 В ТВЗ включается:

- ПЗ модуля ЗОП, при работе модулей РОПСХ, ЗОП;
- имя базового рода структуры, дополнения, при работе модулей ИБРС, ЗИК.
- формируемая при работе модуля ИОТН табл. ТВЗ.

15-2-16

ТПТЗ м 2

- Блок 8 , 10, 27. Проверяется, что есть необработанные запросы на изменения.
- Блок 11 Проверяется, что получен в ППП ГРС хотя бы один промежуточный род структуры, имя которого входит в ТРВЗ.
- Блок 12 Создаются новые поколения указанных массивов, в которые не включаются записи, содержащие имена из ТРВЗ.
- Блок 15 Проверяется, что до пакета ВМ работает пакет R - интерпретации.
- Блок 16 Имя модуля, реализующего требуемое изменение, выбирается из ТИЗМ 3. Кроме указанных выше модулей в ТИЗМ 3 включается (в графу "наименование") имя блока, удаляющего значения R - интерпретации из массивов ЗНАЧР и соответственно корректирующего массив КРВГ. Процедура выбора для ТИЗМ 3 аналогична процедуре выбора для ТИЗМ 1.
- Блок 18. В ТВЗ включается:
- ПЗ модуля РАК, в случае работы этого модуля;
 - ПЗ модуля РСК, в случае работы этого модуля;
 - ПЗ модуля РРТ, в случае работы этого модуля;
 - ПЗ модуля СРТ, в случае работы этого модуля.
- В УЗ включается:
- Т (см. 4.3.13), в случае работы модуля $SRel$;

15-2-76

ТНТЗен2

- СУПГ (см. 4.3.10) в случае работы модулей РСК или СРТ.

При каждом входе в блок происходит добавление новых значений ТВЗ и УЗ.

- Блок 23 Создается новое поколение массивов, в которые не включены R - интерпретации конституэнт или подвыражений, имена которых входят в УЗ1.
- Блок 25 Имя модуля, выполняющего требуемое изменение, выбирается из ТИЗМ 4.
- Блок 28 Описание, использование ключей и их определение см. 4.3.
Значения ключей могут задаваться тремя способами:
- по умолчанию;
 - вместе с заданием ^{на} внесении изменения;
 - по запросу модуля, дошедшего до соответствующего блока.
- Блок 28 изменяет значения ключей, которые будут подразумеваться по умолчанию.
- Блок 30 Сообщается, что в пакете ВП нет средств реализующих заданные изменения.

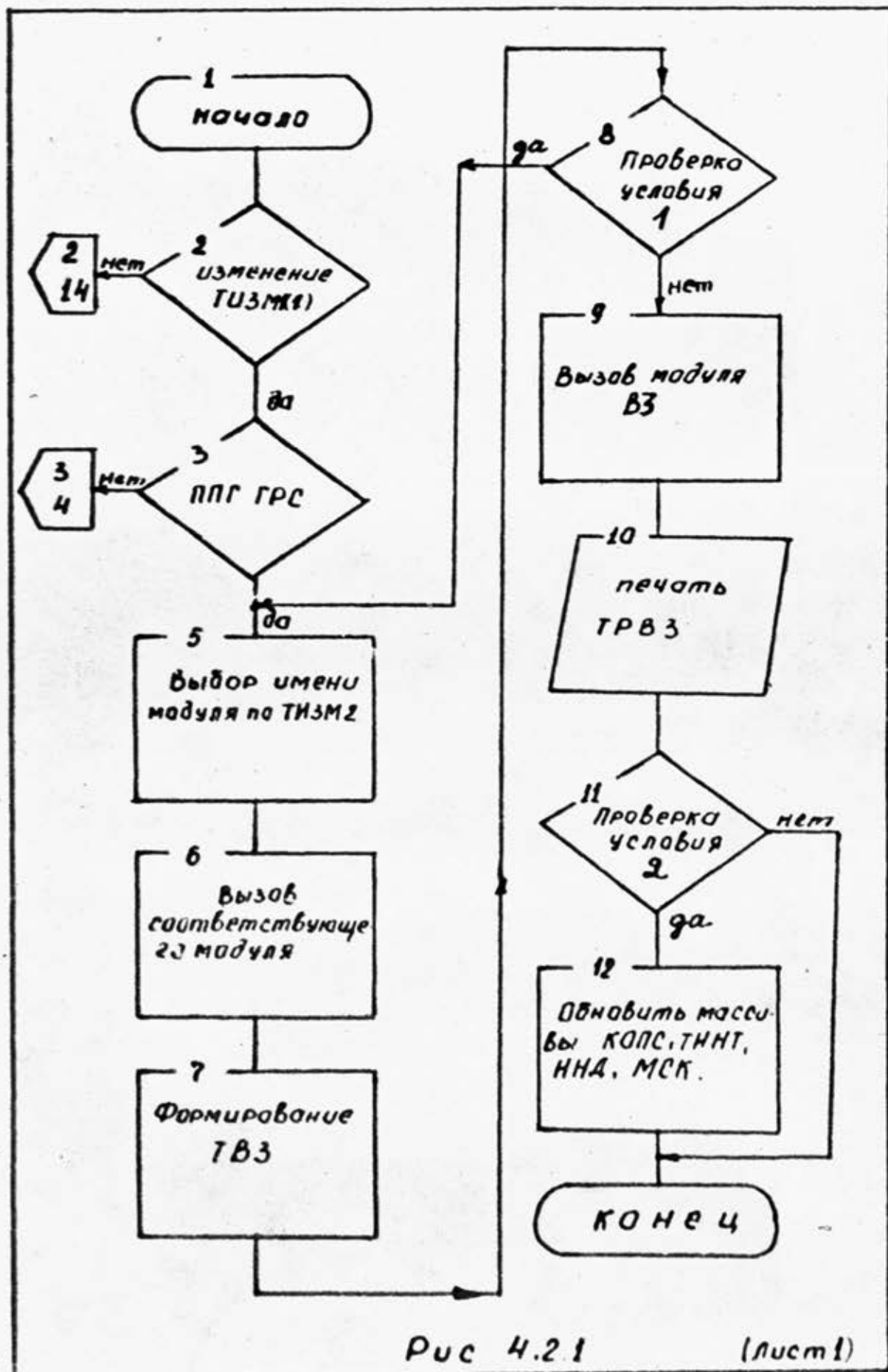


Рис 4.2.1

(лист 1)

16-2-9

ТПГЗК2

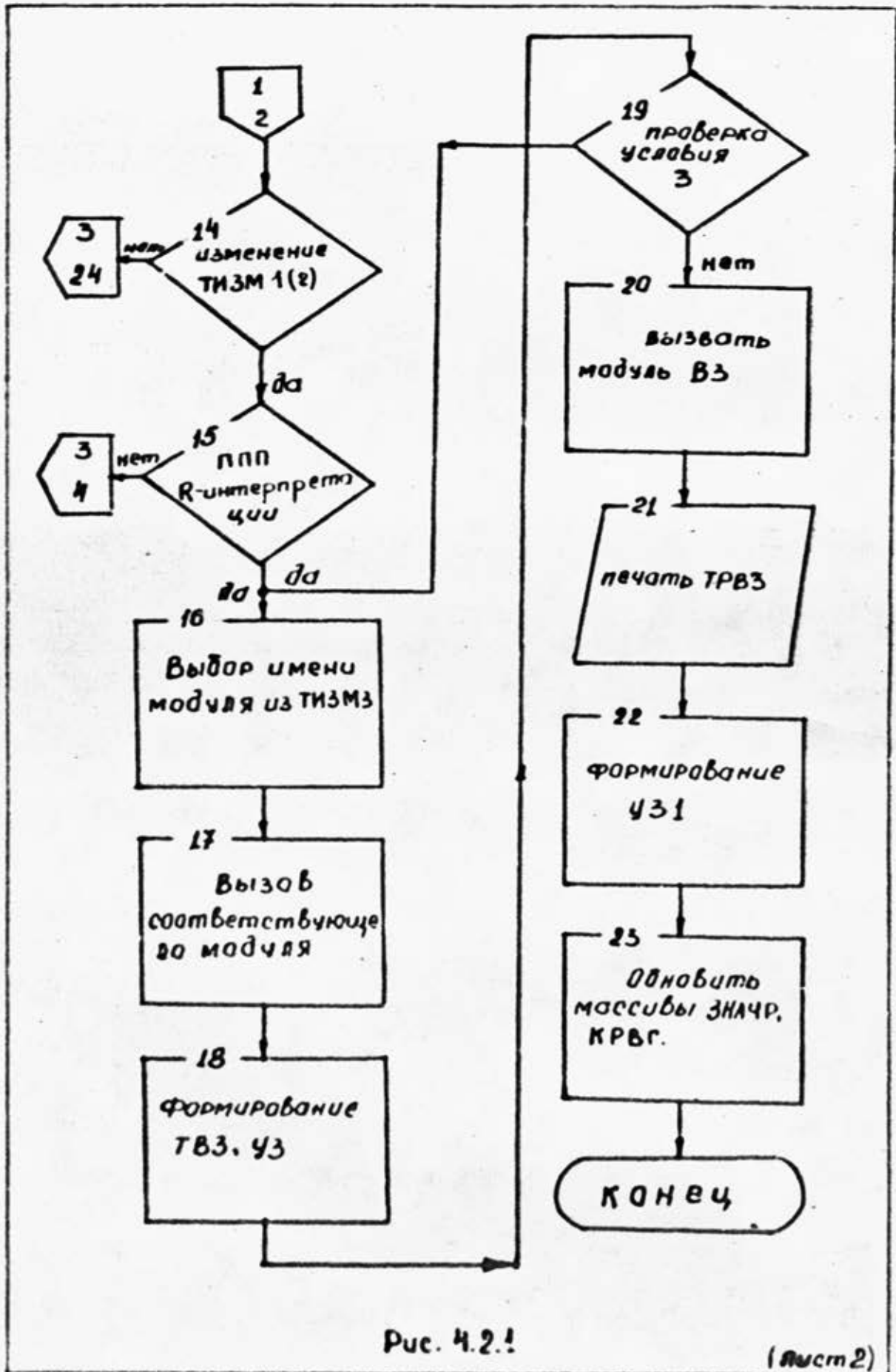
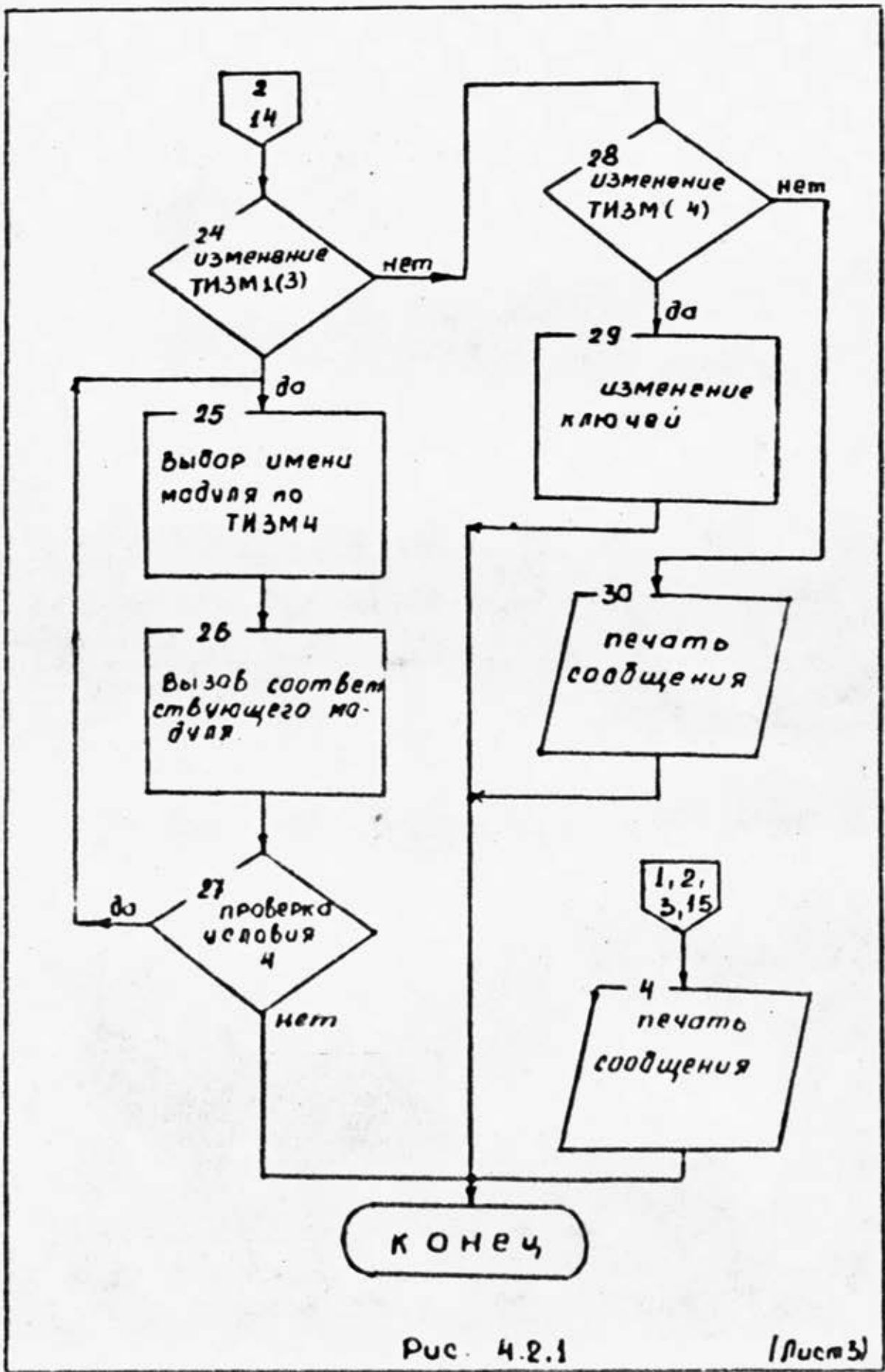


Рис. 4.2.1

(Лист 2)

15.4.28
ТИЗМ2



15-2-26
ТНЗМ2

Рис. 4.2.1

(лист 3)

4.3. Описание алгоритмов модулей тела пакета.

Списание алгоритмов модулей тела пакета предполагает знакомство с описаниями алгоритмов пакетов формирования ГРС, R -интерпретации и перевода представлений. В дальнейшем используем ряд понятий из перечисленных пакетов, в частности из ППП ГРС следующие понятия:

- конституента;
- операционная схема (ОПСХ);
- род структуры (Φ);
- главный род структуры (ГРС);
- дополнение (E);
- отождествляющее отображение (Γ);
- отождествляющее отображение V ;
- оператор $F_{\vee \Phi}$

Из пакета R -интерпретации использованы следующие понятия:

- отождествляющее отображение Γ ;
- граф конституент (ГК);
- граф R -интерпретации (ГР);
- множество REL ;
- множество специнтерпретируемых конституент;
- множество автоматически интерпретируемых конституент;
- разрешающий терм;
- нижнее замыкание;
- операционное заполнение графа;
- подграф.

Из пакета перевода представлений использованы следующие понятия:

- индивидуальное имя, термин;
- сокращенный схемный код;
- полный схемный код.

Введем дополнительно следующие понятия: под текущим ГК понимается подграф ГК ГРС (длины > 1), полученный при работе пакета R -интерпретации или в результате работы модулей данного пакета.

Пусть REL - множество вершин текущего ГК, REL_1 - множество вершин ГК и ГРС. Если A - подмножество REL , то $N(A, GK)$

множество вершин нижнего замкания, определенного по A и $ГК$.

Определение 1.

Вершина B - элемент сильного нижнего замкания ($СНЗ(A)$), если $B \in N(A, ГК)$ и $B \notin N(REL \setminus \{A, ГД, ГК\})$.

Определение 2.

Пусть A - подмножество REL , тогда вершина B - элемент дополнительного нижнего замкания ($ДНЗ(A)$), если $B \in N(A, ГК ГРС)$ и $B \notin REL$.

Определение 3.

Пусть A - подмножество REL , определим верхнее замкание ($ВЗ(A)$) индуктивным образом:

1. $A \subset ВЗ(A)$.

2. Если найдется нижняя вершина подграфа $C(B)$, $B \in REL$, что $C(B) \in ВЗ(A)$, то $B \in ВЗ(A)$.

3. Других элементов, кроме определенных в п.п.1,2, множество $ВЗ(A)$ не содержит.

В дальнейшем подмножество, по которому производится верхнее замкание, обозначается $ТВЗ$, и $ВЗ(ТВЗ)$ обозначается $ТРВЗ$. Множество $ДНЗ(A)$ в дальнейшем обозначается $ТДВ$, и множество $СНЗ(A)$ - $СУВ$.

Под макро модулем в данном пакете понимается модуль, состоящий из нескольких модулей, число которых определяется при генерации.

ТРЕК - таблица, описывающая макро модуль.

В i -той строке таблицы указано:

- режим (имя модуля, входящего в макро модуль);
- метка, определяющая, куда нужно перейти, если заданный режим совпал с режимом в i -той строке;
- метка, определяющая, куда нужно перейти, если заданный режим не совпал с режимом в i -той строке (для всех i , кроме конечного, она указывает на $i+1$ строку **ТРЕК**; для конечного i метка указывает на блок печати).

В **ТРЕК И** - код, идентифицирующий данный макро модуль.

КЛКН - n -ый ключ в модуле с кодом K .

В данном подграфе описаны следующие модули:

- расширения операционной схемы, (РОПСХ);
- сокращения операционной схемы, (СОПСХ);
- замены операции, (ЭП);

- изменения базового рода структуры (дополнения), (ИБРС);
 - замена имени конstituэнты в базовом роде структуры (дополнении), (ЗМК);
 - изменения отождествляющего отображения R , (ИСТН);
 - сокращения сильного нижнего замыкания, (ССНЗ);
 - дополнительного нижнего замыкания, (ДНЗ);
 - верхнего замыкания, (ВЗ);
 - исключения вершин из графа R -интерпретации, (ИЗГР);
 - расширения графа R -интерпретации, (РГР);
 - расширения множества интерпретируемых конstituэнт, (P REL);
 - сокращения множества REL, (С REL);
 - расширения множества автоматически интерпретируемых конstituэнт, (РАК);
 - расширения множества специнтерпретируемых конstituэнт, (PCK);
 - сокращения множества разрешающих термов, (СРТ);
 - расширения множества разрешающих термов, (РРТ);
 - изменения операционного заполнения, (ИОЗ);
 - изменения индивидуальных имен конstituэнт, (ИЗИНИМ);
 - изменения сокращенных схемных кодов, (ИССК).
- Если A - множество имен конstituэнт, то A' - множество конstituэнт, имена которых указаны в A ; \bar{A} - множество имен конstituэнт, входящих в выражения конstituэнт из A' .

4.3.1. Описание алгоритма модуля расширения операционной схемы.

модуль предназначен для введения новых вершин (операций) в операционную схему и замены операций или их аргументов в старых вершинах.

Модуль вызывается управляющей программой пакета. Вся необходимая информация вводится самим модулем.

Блок-схема приводится на рис. 4.3.1. В блок-схеме приняты следующие обозначения:

ТКСВ - таблица кодов старых вершин, т.е. вершины старого графа операционной схемы

ТКНВ - таблица кодов новых вершин ОПСХ, т.е. добавляемых вершин.

ТКНВ1 - часть вершин ТКНВ, не совпадающая с вершинами из ТКСВ.

в - вершина

$N(v)$ - номер слоя вершины "в"

Зоп - замена операции.

Опишем отдельные блоки алгоритма.

Блок 6. Проверяется, что в ТКНВ не включены вершины из ТКСВ.

Блок 7. Сообщается, какие вершины из ТКНВ не попали в ТКНВ1.

Блок 10. Проверяется, что можно выбрать из ТКНВ "вершину", для которой аргументы операции, ей соответствующей, не указаны в ТКНВ, но указаны в ТКСВ. Это не позволяет при вводе расширения ОПСХ не требовать формативности в расширении ОПСХ.

Блок 11. Сообщается, для каких элементов ТКНВ условие блока 10 не выполняется.

Блок 12. Ввод, вершины удовлетворяющей указанному условию.

Блок 18. Замена операций в заданных вершинах. Модуль сам вводит необходимую ему информацию.

4.3.2. Описание алгоритма модуля сокращения операционной схемы.

Модуль предназначен для сокращения операционной схемы, т.е. удаления из ОПСХ указанных операций и всех операций,

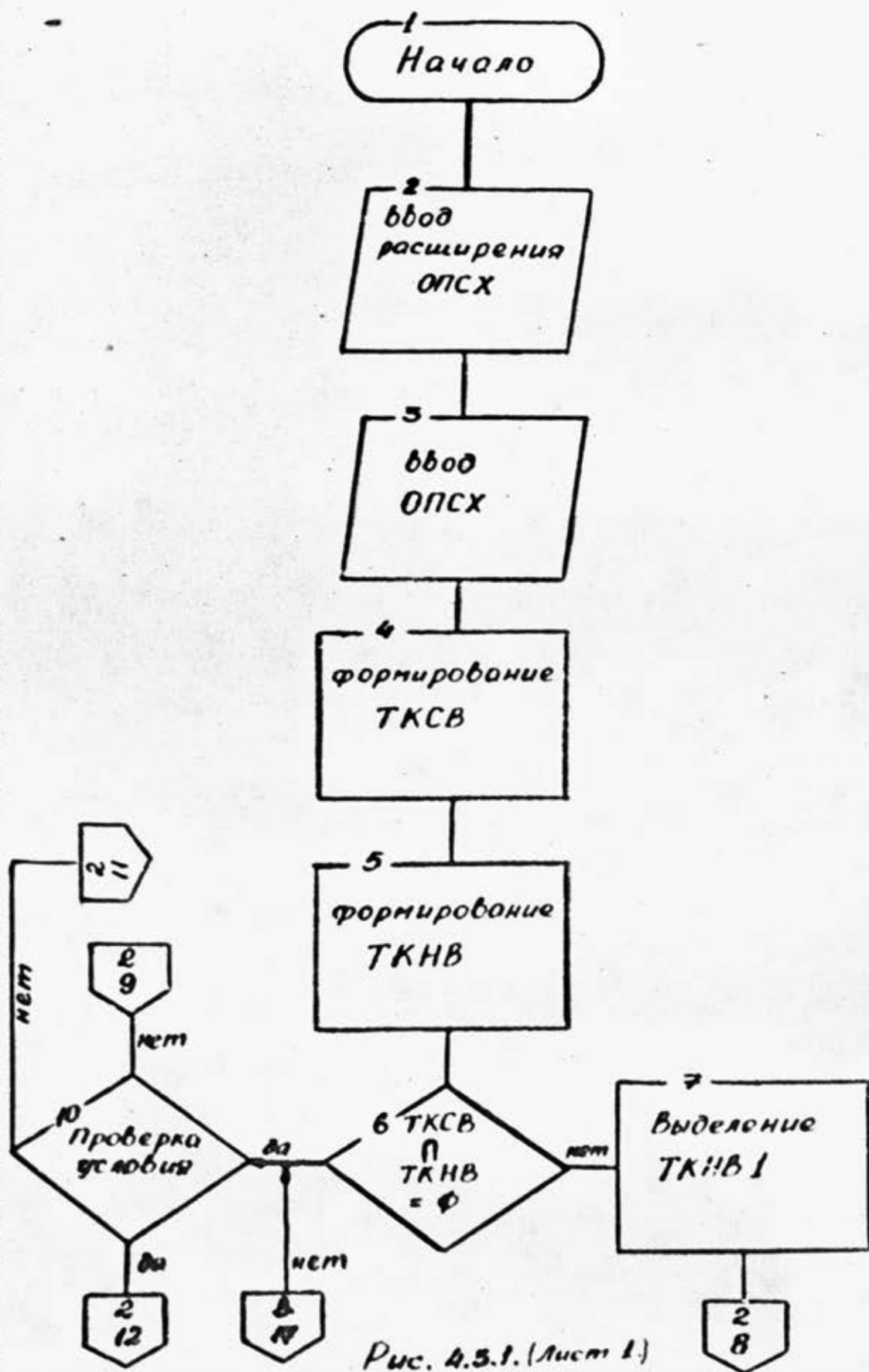


Рис. 4.5.1. (Лист 1.)

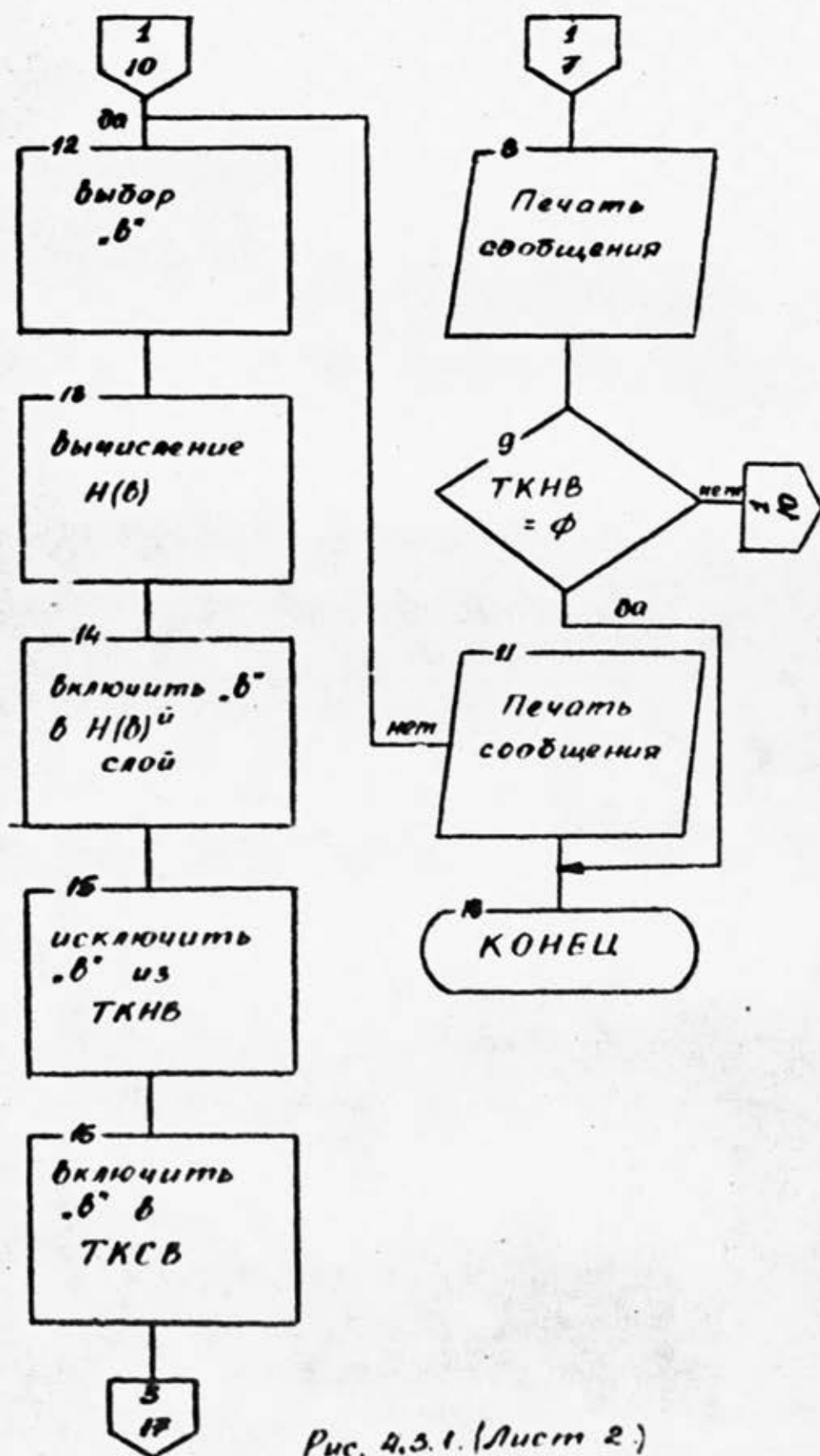


Рис. 4.3.1. (Лист 2.)

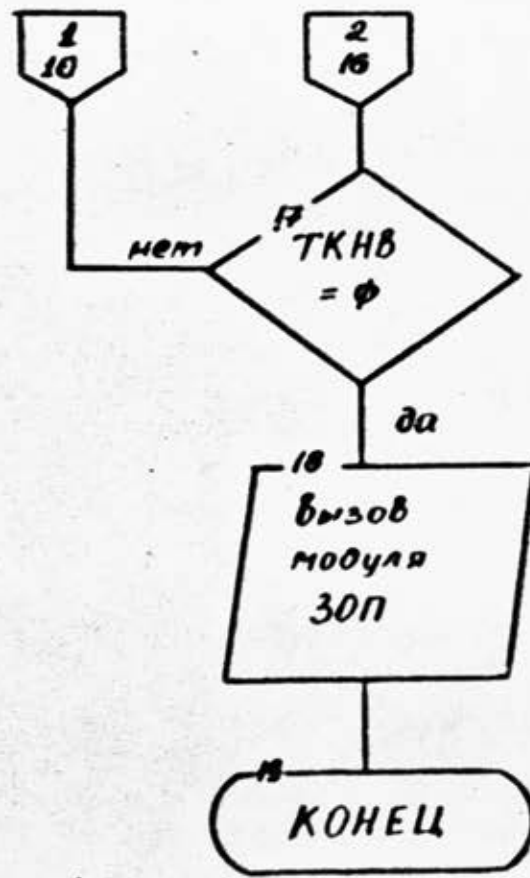


Рис. 4.3.1. (Лист 3.)

которые необходимы только для вычисления аргументов, указанных операция (ССНЗ).

Модуль вызывается определяющей программой пакета. Необходимая для работы информация вводится самим модулем.

В модуле используются модули ССНЗ и ВЗ, описанные в данном пакете.

Блок-схема приводится на рис. 4.3.2.

В блок-схеме приняты следующие обозначения:

ТКИБ- таблица кодов исключаемых вершин.

Н(в)- номер слоя вершины "в" из ТКИБ.

Опишем отдельные блоки алгоритма.

Блок 3. Ввод сокращения ОПСХ по форме 1

Блок 5. Определяется Н(в) для всех "в" из ТКИБ по следующему правилу:

$$N(v) = \max_{b \in TKIB} N(b) + 1$$

Блок 6. Сортировка ТКИБ по номерам слоев.

Блок 9. Определяет удалять ли вершины из ТРВЗ.

4.3.3. Описание модуля замены операции.

Модуль предназначен для замены одной или нескольких операций, сопоставленных вершинам графа операционной схемы.

Модуль может вызываться управляющей программой пакета при получении задания на изменения операционного заполнения графа и модулем расширения операционной схемы. В обоих случаях модуль автономно вводит необходимую для работы информацию по форме 1.

Алгоритм модуля.

Алгоритм начинает работу с ввода операционной схемы, в том случае, когда работает автономно, в противном случае получает доступ к ОПСХ, если вызван модулем расширения операционной схемы. Затем вводятся по форме I операции, заменяющие операции в ОПСХ. Перед каждой заменой операции должно проверяться, что все аргументы новой операции определены в ОПСХ и находятся на ниже расположенных по отношению к данному слоях. В случае нарушения этого условия замена не производится и об этом нарушении сообщается проектировщику.

15.2.76
ТПтЗен2

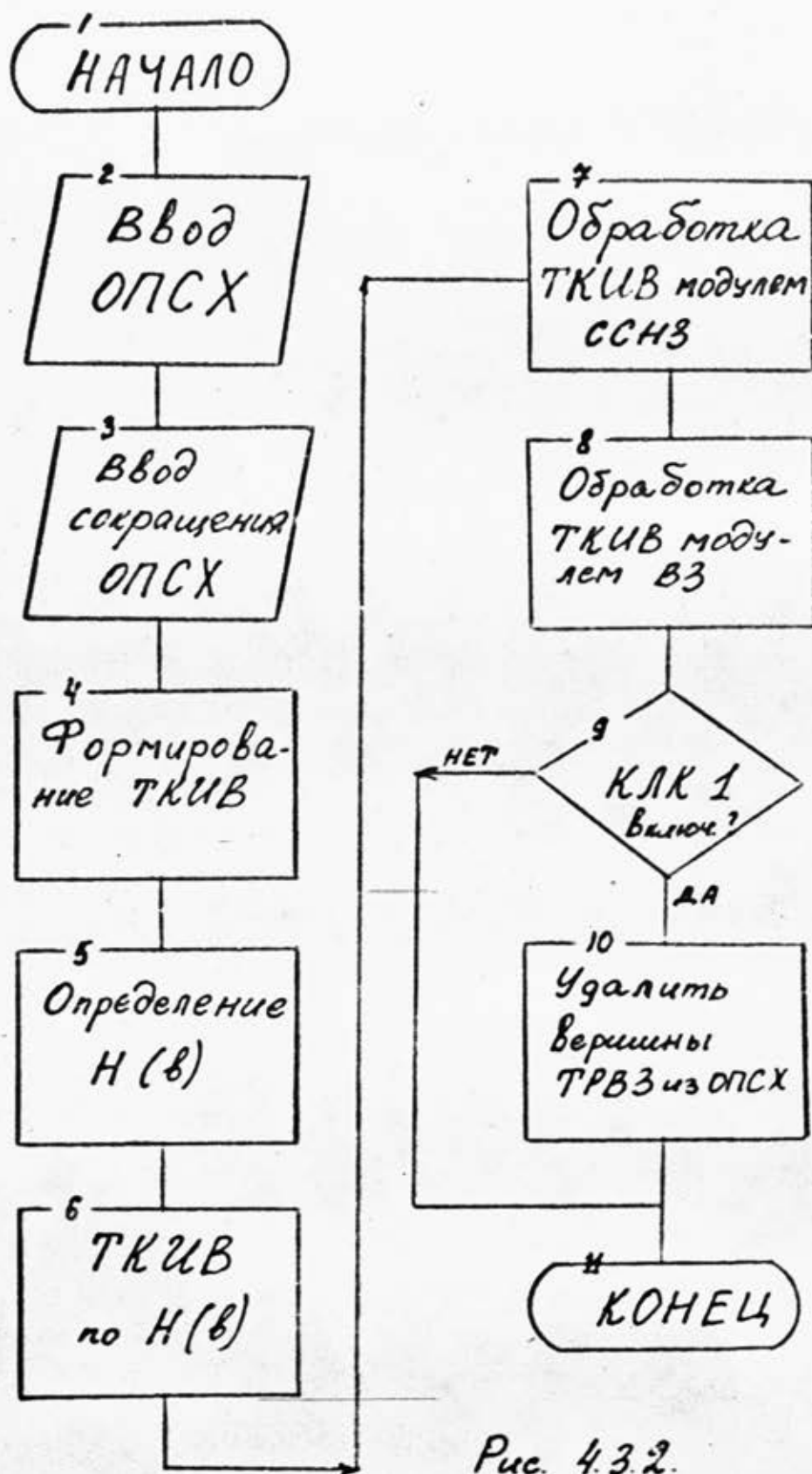


Рис. 4.3.2.

при каждой замене операции, код вершины которой сопоставляется, эта операция, должен заноситься в ТВЗ.

Ввиду простоты алгоритма, блок-схема его не приводится.

4.3.4. Описание макро модуля изменение базового рода структуры (дополнения).

Макромодуль предназначен для изменения базового рода структуры или дополнения. Макромодуль вызывается управляющей программой пакета, которая указывает, что должно производиться или расширение или сокращение или замена. Остальная необходимая информация вводится макро модулем.

Для режима расширения (замены) новые (заменяемые) конститутенты вводятся по форме 2, причем требование формативности среди новых (заменяемых) конститутент, может нарушаться. В случае необходимости, модуль приведет их к формативному виду и произведет необходимые изменения имен. Это позволяет для режима расширения не указывать индексы вводимых конститутент. Для режима сокращения замена конститутент, подлежащих сокращению, вводятся по форме 4.

Блок-схема алгоритма приведена на рис. 4.3.3.

В блок-схеме приняты следующие обозначения.

ТИК- таблица имен конститутент.

A- множество имен конститутент, входящих в ТИК и определенных в Ф.

A_1 - имена конститутент, входящих в A и удовлетворяющих условию 3.

C- имя конститутенты, входящей в A_1 и удовлетворяющей условию I.

D- имя конститутенты, входящее в ТИК и удовлетворяющее условию I.

B- выражения конститутент, в которых должны быть изменены имена конститутент.

B_1 результат работы оператора F_v .

Опишем отдельные блоки алгоритма.

Блок 4. Формируются имена конститутент, подлежащих сокращению, замене или вновь добавляемые. Конститутенты классов П, К и Д должны входить одновременно. Если один элемент из пары не задан, то он определяется либо из Ф (E)

либо требуется от проектировщика.

Блок 6. Условие 4. Все имена из ТИК присутствуют в $\Phi(E)$.

Блоки 8, 42. Сообщается, какие имена из ТИК не вошли в А.

Блок 10. Проектировщик определяет, нужна ли информация о связях конституэнт.

Блок 11. Реализуется модулем формирования графа конституэнт в пакете R-интерпретации.

Блок 13 - определяет, нужно ли производить изменение.

Блок 15 - удаляет конституэнты, имена которых находятся в ТИК из $\Phi(E)$.

Блок 16 - определяет, удалять ли конституэнты, имена которых указаны в ТРВЗ.

Блоки 13-21. Производят изменения имен в выражениях конституэнт, вызванных вносимыми исходными изменениями, когда либо надо переместить конституэнту на заданное ее индексное место и оно уже занято, либо уплотнить конституэнты после сокращения. Изменения делаются одновременно для конституэнт ^{классов} Π, K, Φ, M

Блок 22 - производит контроль $\Phi(E)$. Этот контроль реализован модулем СУБД.

Блок 23 - печатается сообщение, если $\Phi(E)$ не удовлетворяет требованиям контроля.

Блок 24 - определяет, надо ли печатать $\Phi(E)$.

Блок 28 - сообщается проектировщику, что заданный режим не реализован в модуле.

Блоки 29, 48 - Условие I.

Можно выбрать из ТИК имя конституэнты, в выражение которой не входят имена других конституэнт из ТИК.

Блоки 30, 49 - сообщение о невыполнимости условия I.

Блок 31 - Выбор А.

Блоки 32, 53 - Определяется по индексам конституэнт, входящих в выражение.

Блок 44 - Условие 3. Проверяется, что имена конституэнт, входящих в А, определены в $\Phi(E)$.

Блок 46 - сообщается, какие конституэнты не удовлетворяют условию 3.

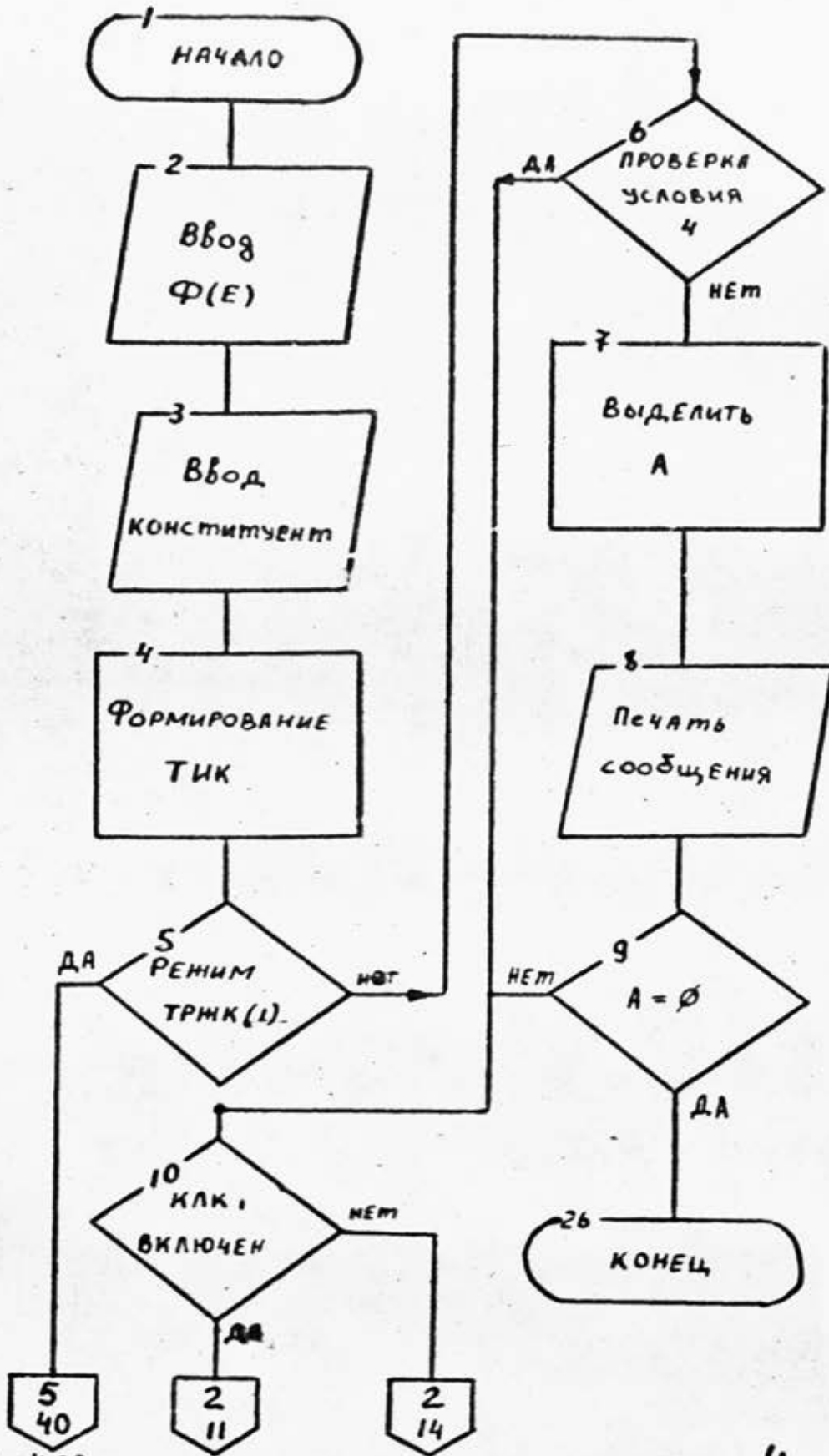


Рис 4 33

15-2-86
ТПТЗк2

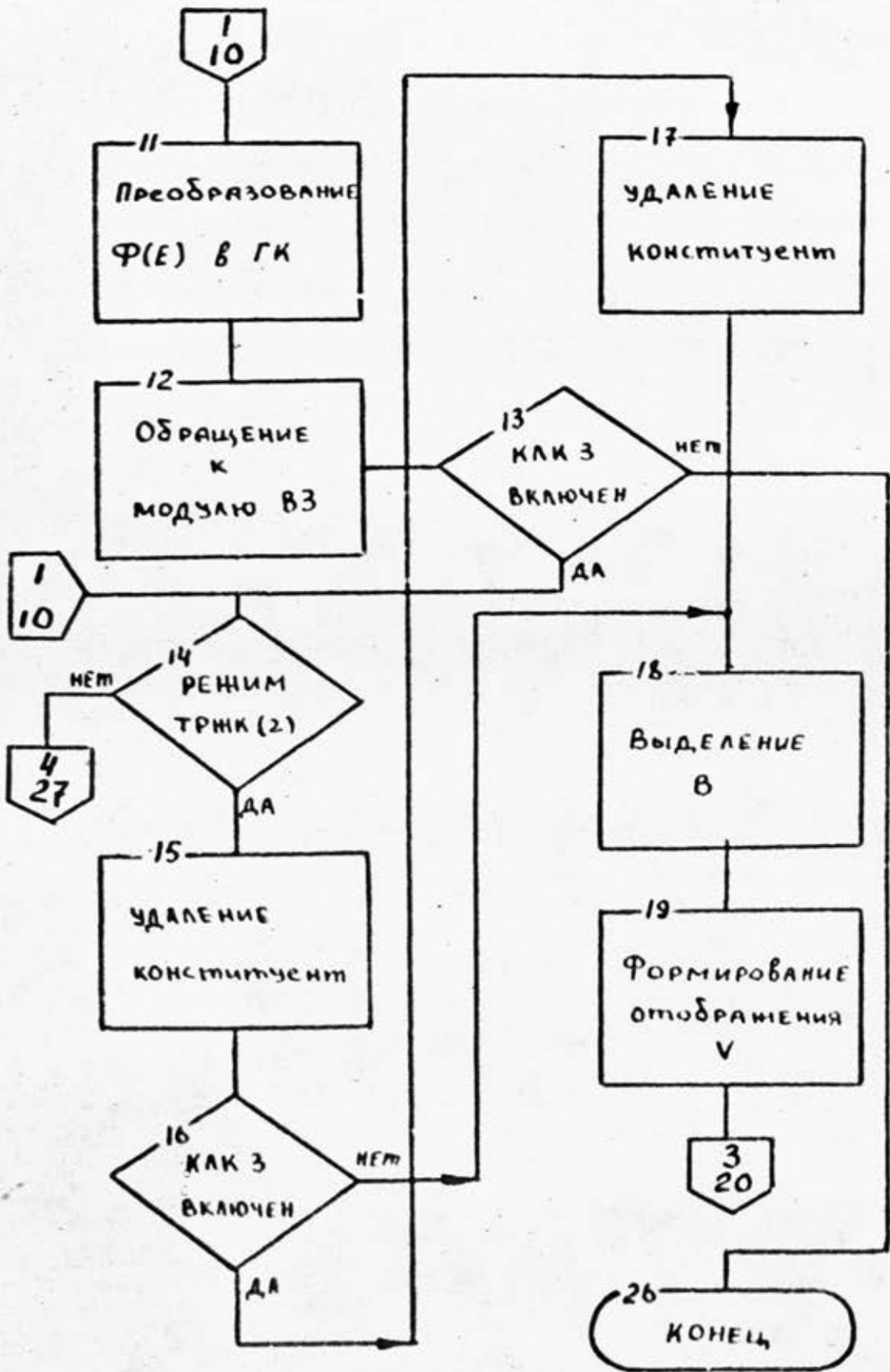


Рис 433

(Лист 2)

15-2-16
ТПЗКН2

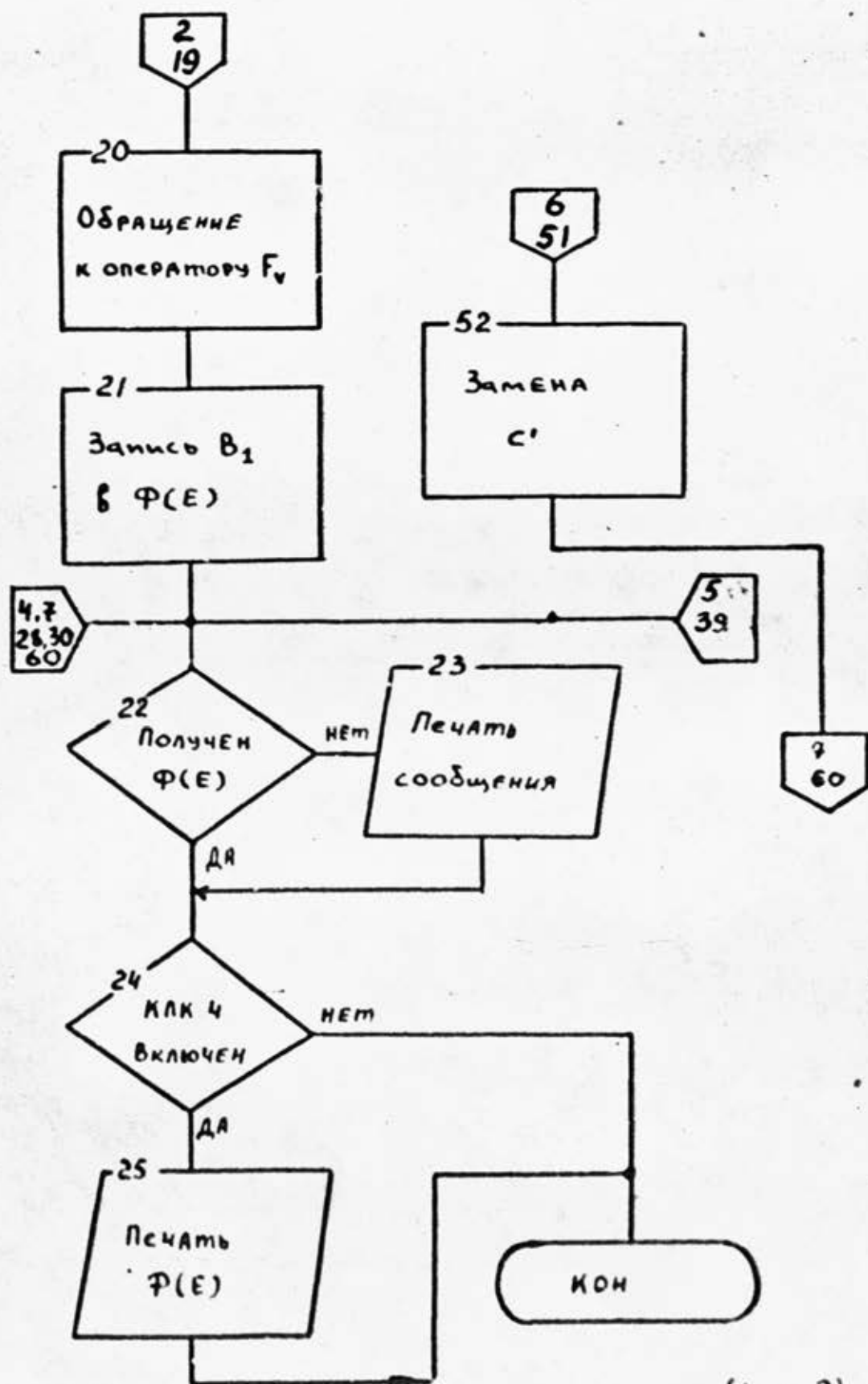


Рис 433

(Лист 3)

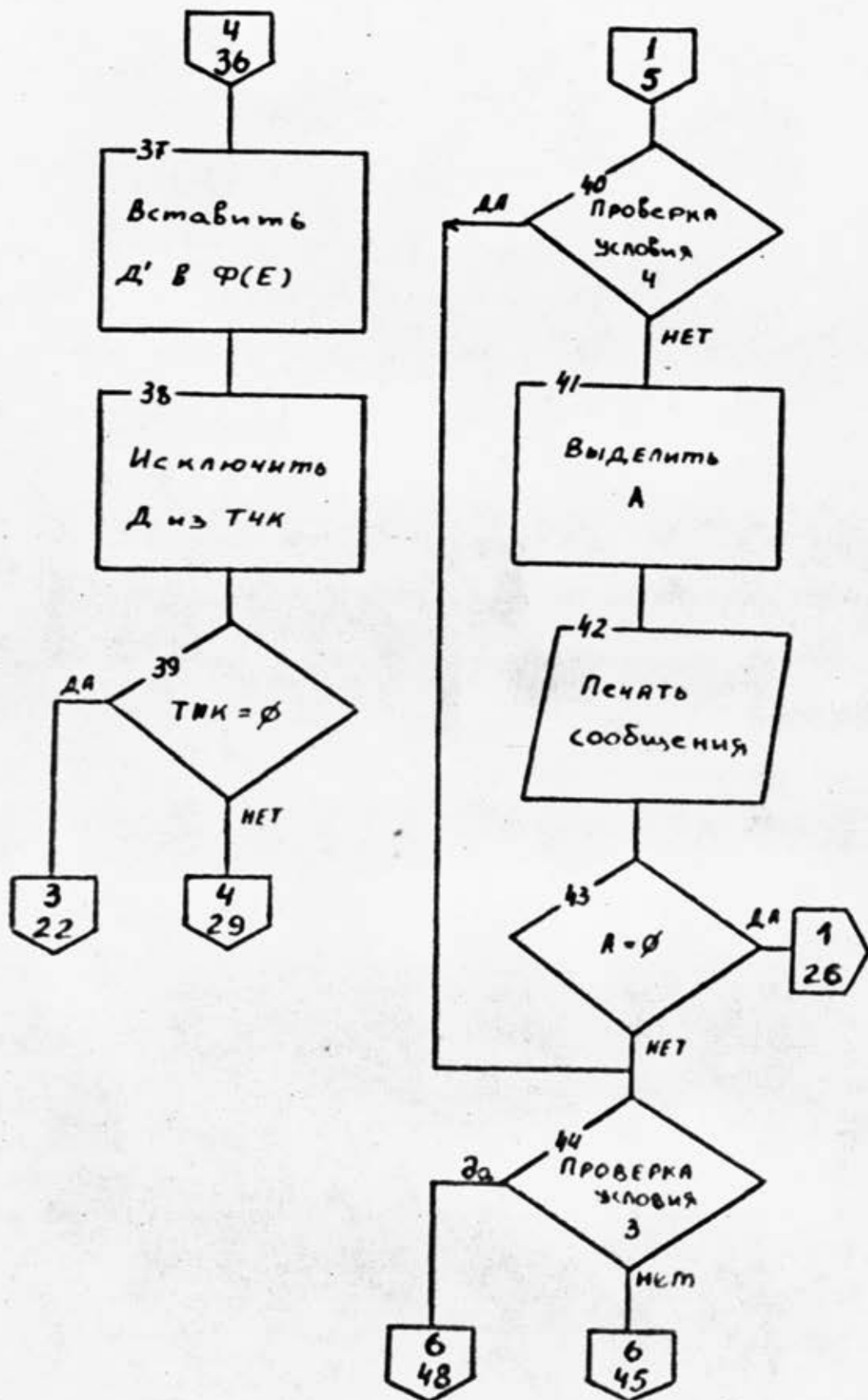


Рис. 433

(Лист 5)

15-2-86
ТНТЗКР

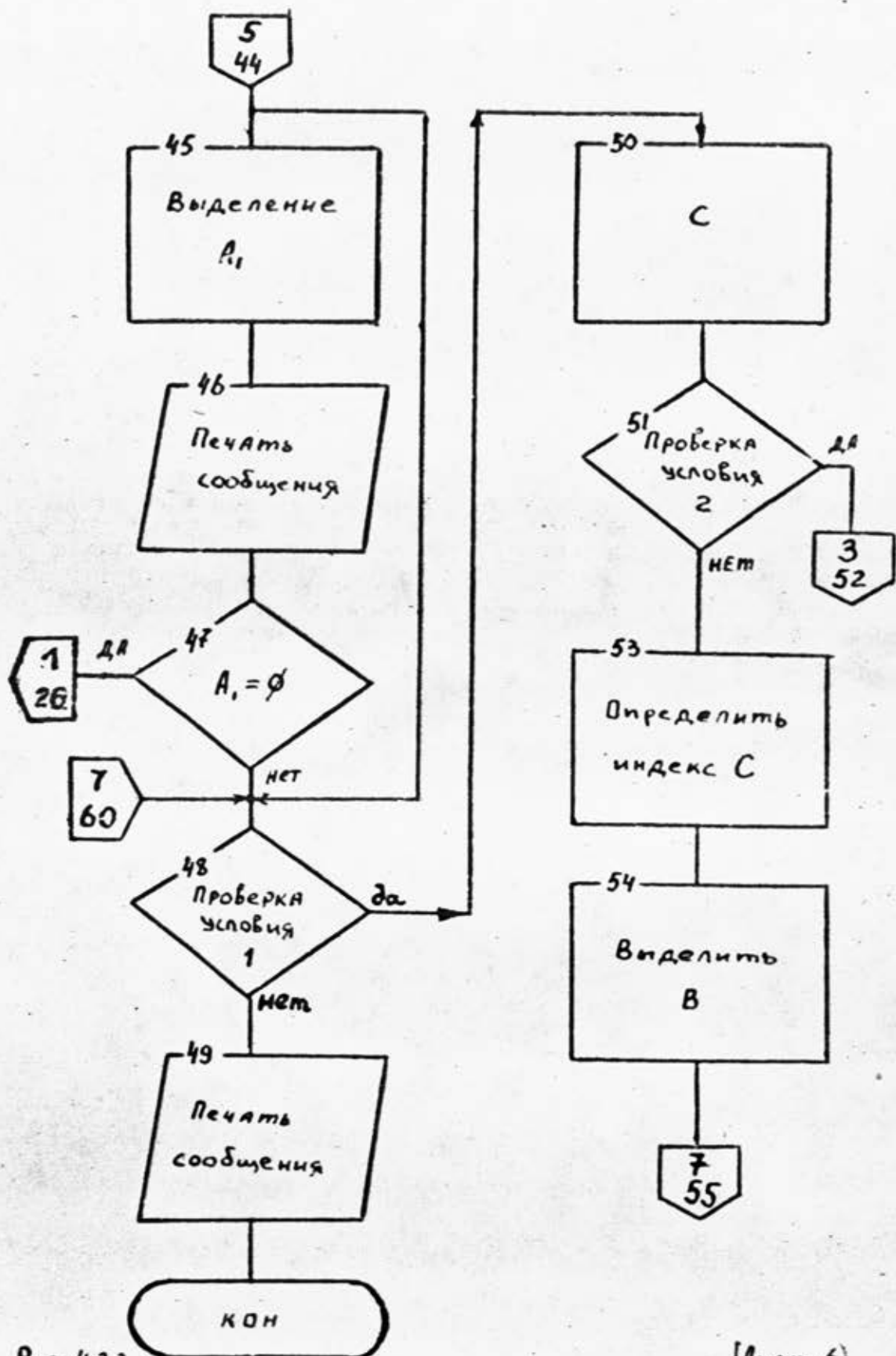


Рис 433.

[Лист 6]

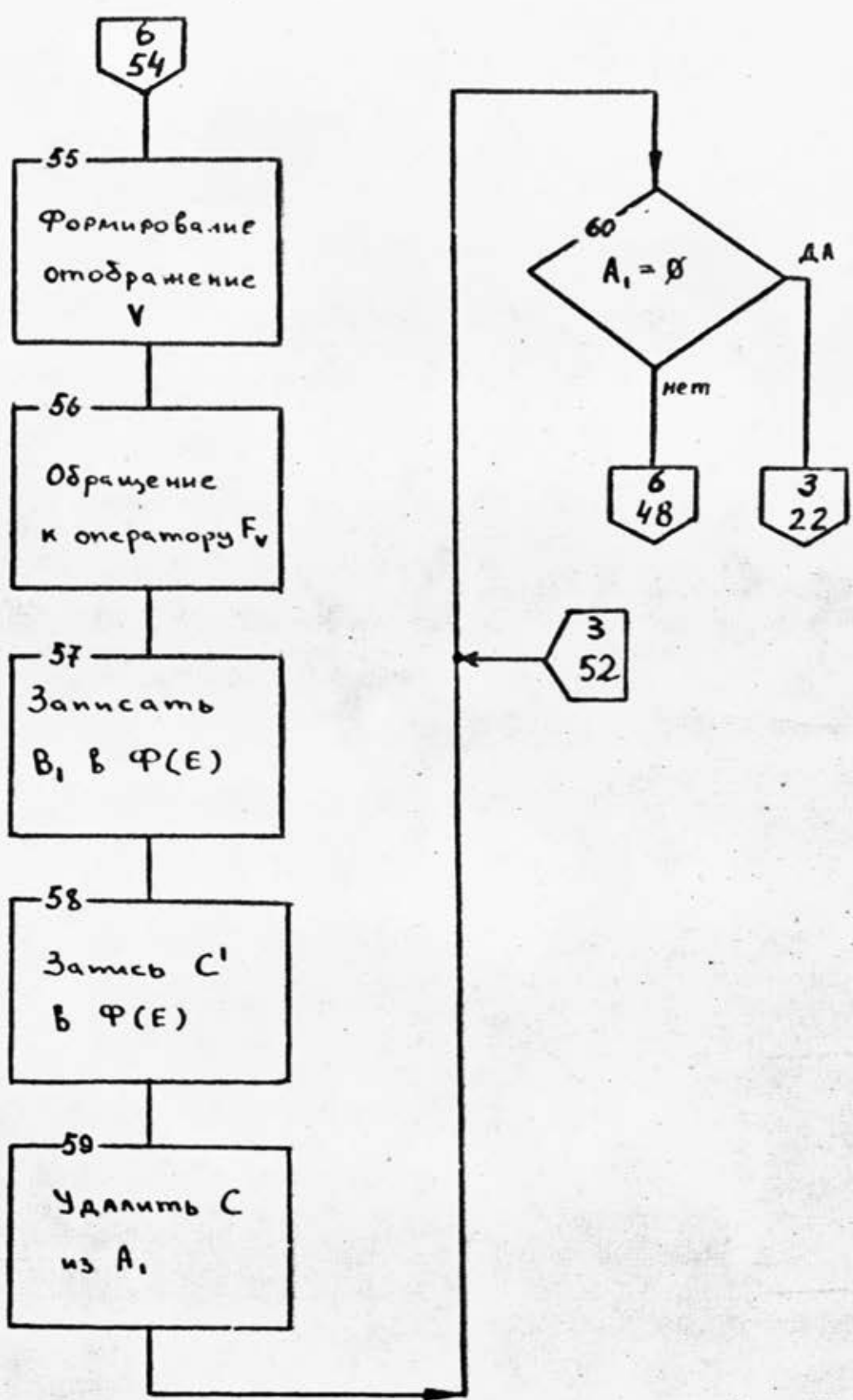


Рис. 433.

(Лист 7.)

15-2-76
ТПГЗен2

Блок 50 - выбор С.

Блок 51 - Условие 2: проверяется, что в новое выражение входят имена конститuent с меньшими индексами.

4.3.5. Описание модуля замены имени конститuent в роде структуры, дополнении.

Модуль предназначен для замены одного или нескольких имен конститuent в одном (или нескольких последовательно расположенных) выражении (ниях) конститuent в указанном роде структуры, дополнении.

Модуль вызывается управляющей программой пакета (УПГ). Вся необходимая информация вводится самим модулем: отображение γ по форме Π , род структуры (дополнение) в виде ^{виде} хранимом в банке данных.

Блок-схема алгоритма приведена на рис. 4.3.4.

В блок-схеме приняты следующие обозначения:

γ - задание для работы модуля, в виде, задаваемом формой Π . Каждая строка формы отдельное задание для того же $\Phi(E)$.

Опишем отдельные блоки алгоритма.

Блок 5 - проверяется, что в $\Phi(E)$ не найдется конститuent, в выражении которой имя одной конститuent должно быть заменено за несколько имен других конститuent.

Блок 7 - сообщение о неоднозначно заданном γ .

Блок 8 - проверяется, что должны быть изменены выражения конститuent в указанных границах, а не одно выражение.

Блоки 11-13 - производятся замены имен.

Блок 16 - проверяется, что получен род структуры (дополнение). Контроль производится модулями из БД.

4.3.6. Описание макромодуля изменения отождествляющего отображения. Π

Макромодуль предназначен для изменения соответствия между базовыми родами структур, дополнениями, отображениями и начальными вершинами графа операционной схемы.

Модуль вызывается управляющей программой пакета, которая указывает, что должно производиться расширение или сокращение,

или замена. Остальная необходимая для работы информация вводится самим модулем. Изменения для h вводятся по той же форме, что и отображение h .

Блок-схема приведена на рис. 4.3.5.

В блок-схеме приняты следующие обозначения:

h_1 - расширяемые, сокращаемые или заменяемые элементы;

h_2 - сужение отображения h_1 , у элементов которого вершины определены в ОПСХ;

h_3 - элементы из h_1 не попавшие в h_2 ;

h_4 - сужение h_2 , определяемое требованием однозначности;

T - множество имен базовых родов структур, дополнений, отображений, определенных в h_2 ;

T_1 - множество начальных вершин ОПСХ, для элементов которого изменено отображение h_2 ;

$A(h_2)$ - множество имен базовых родов структур, дополнений, отображений, которые есть в h , но не встречаются в h_2 ;

$B(h_4)$ - множество имен базовых родов структур, дополнений, отображений, индексы которых встречаются в h_4 .

Опишем отдельные блоки алгоритма.

Блок 9 - сообщается, какие элементы h_1 , не попали в h_2 .

Блок 12 - удаление $\Phi, E \in T'$ из массива Φ, E и удаление $\Gamma \in T'$ из массива Γ .

Блок 15 - сообщается проектировщику, что указанный режим в модуле не задействован.

Блок 18 - сообщаются элементы h_1 , уже находящиеся в h .

Блок 19 - замена базовых индексов на коды вершин, которым сопоставляются эти элементы.

Блок 20 - объединение двух таблиц.

Блок 22 - сообщается, какие элементы h_1 не попали в h_2 .

Блок 24 - в h производятся замены согласно h_2 .

Блок 26 - сообщается, какие элементы h_2 нарушают однозначность h . Соответствующие элементы h остаются без изменений.

Блок 31 - заменить в $B'(h_4)$ старые коды вершин на новые согласно h_4 .

Блоки 32, 33 - подготовка информации для верхнего зам-
кания.

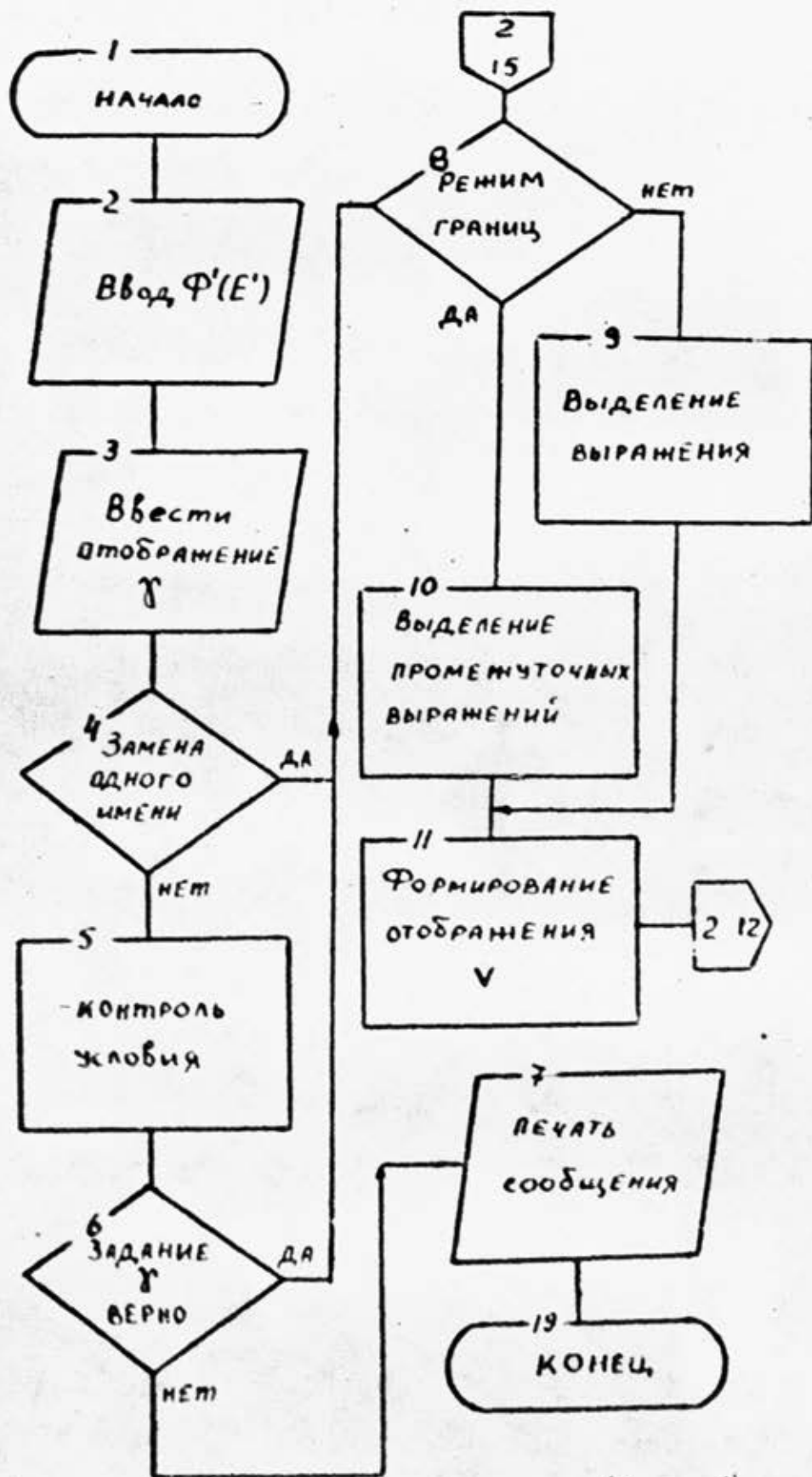


Рис. 4.3.9

(Лист 1)

15.2.76
ТТЗМ2

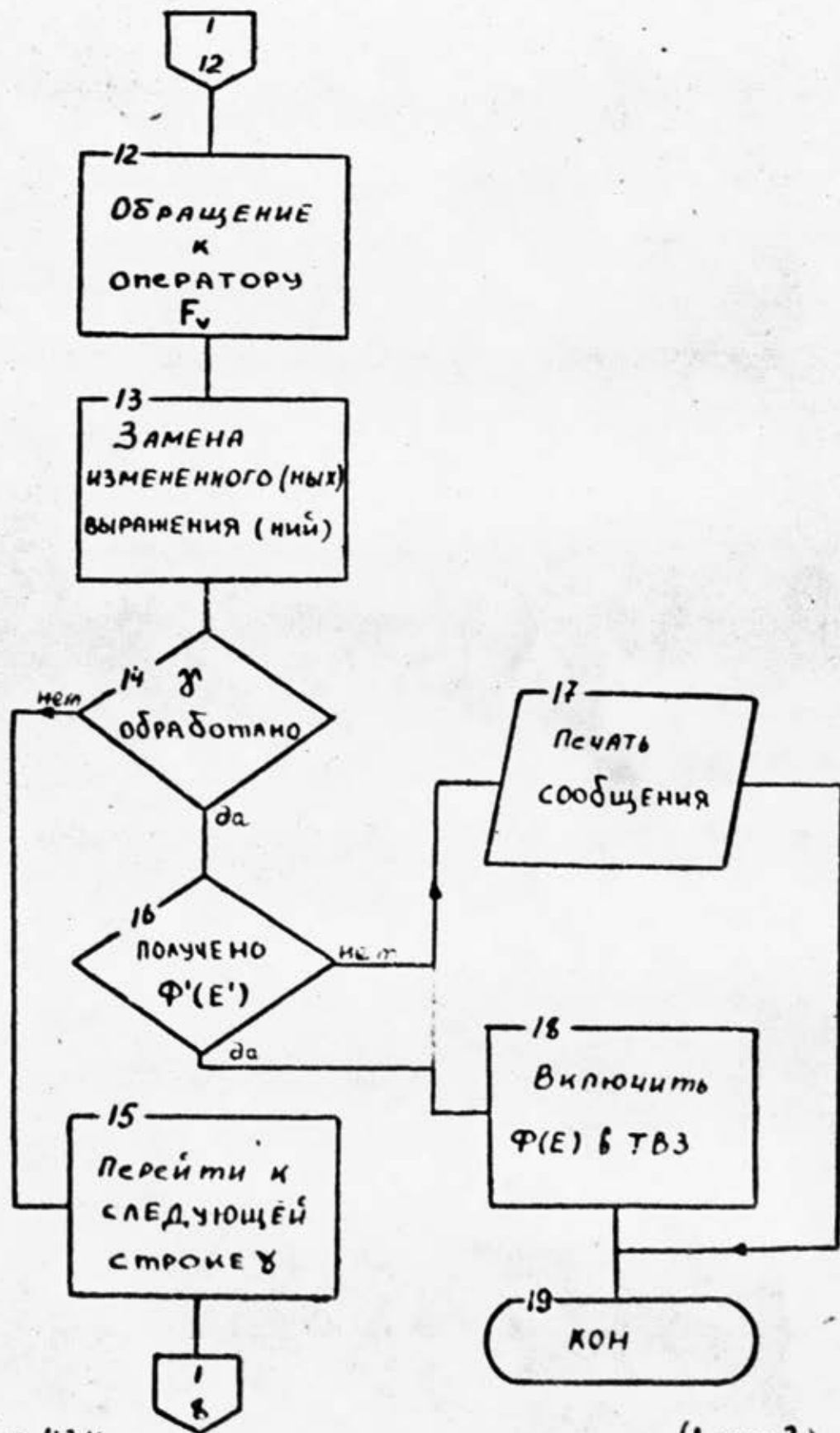


Рис. 434

(Лист 2)

18-2-86
ТПТЗкп2

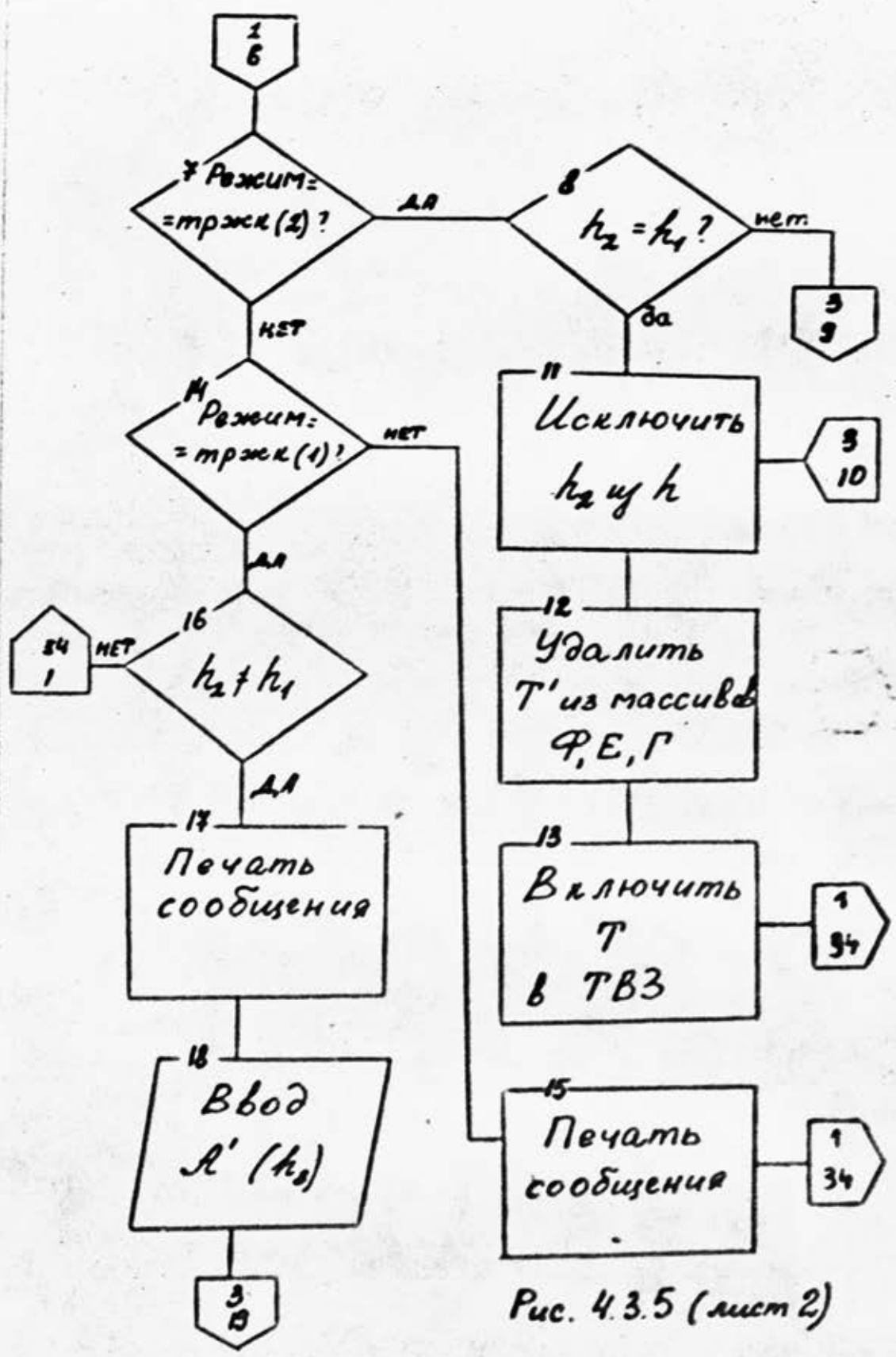


Рис. 4.3.5 (лист 2)

15.2.76
ТП.3.м2

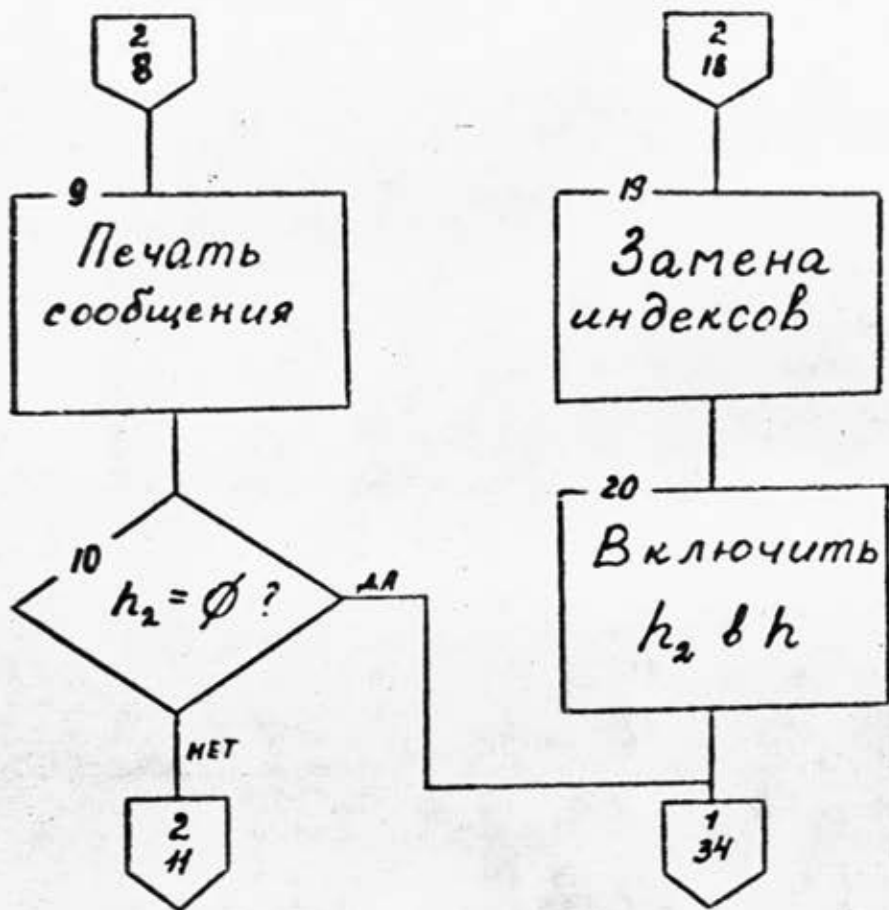


Рис 4.3.5 (лист 3)

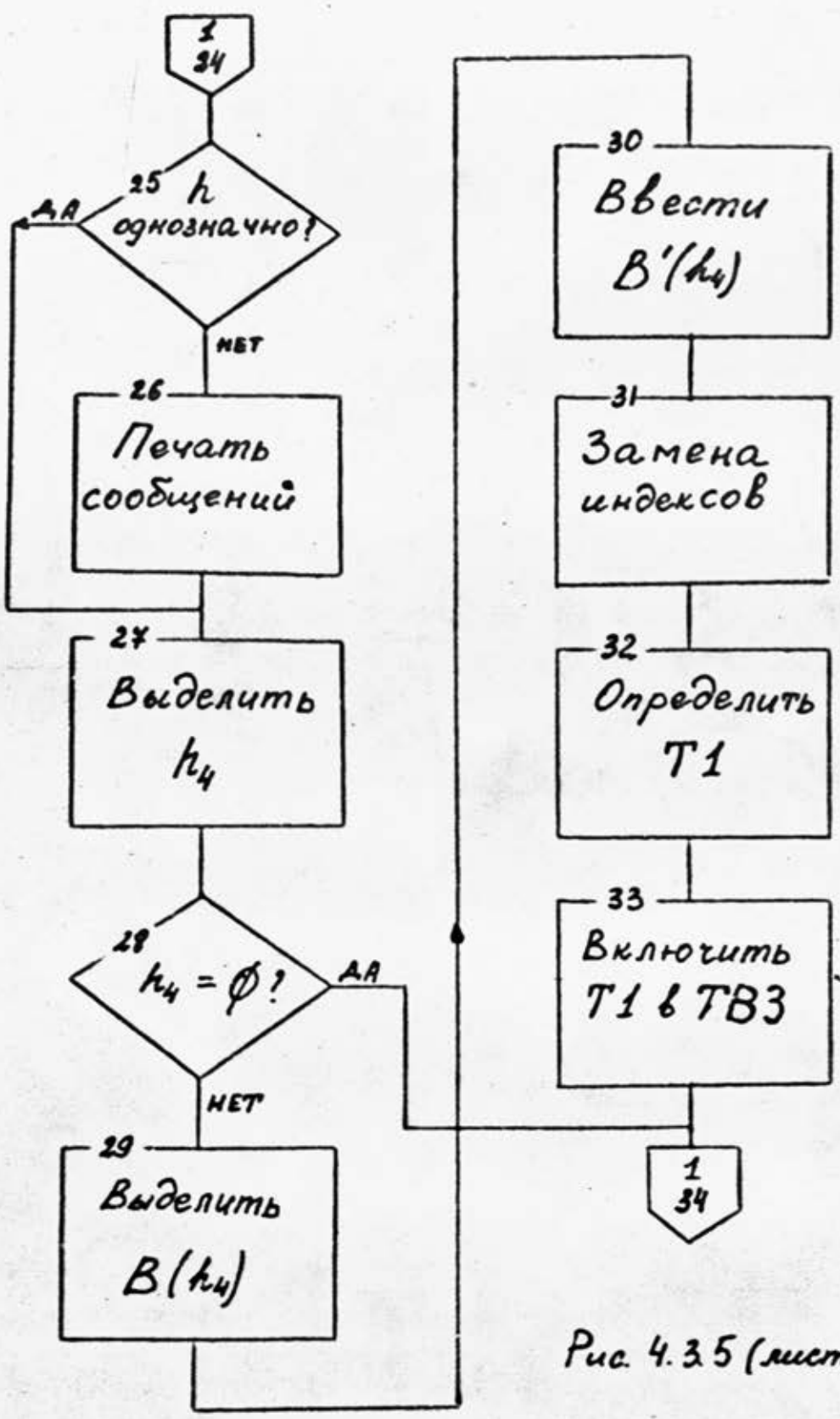


Рис. 4.35 (лист 4)

15-2-96
ТПТЗК12

4.3.7. Описание алгоритма модуля сокращения сильного нижнего замыкания.

Модуль предназначен для сокращения сильного нижнего замыкания данных конститuent.

Модуль вызывается следующими модулями пакета:

- сокращения множества REL ;
- сокращения разрешающих термов;
- расширения множества специнтерпретируемых конститuent;
- сокращения операционной схемы.

При обращении к модулю задается список конститuent и ГХ.

Блок-схема алгоритма приведена на Рис.4.3.6. В блок-схеме использованы следующие обозначения:

СК - задание на работу алгоритма;

A - часть СК, элементы которого определены в ГХ;

A_i - элементы A, $i = 1, 2, \dots, 2$;

B_j - элементы ГХ $j = 1, 2, \dots, N$;

N - число элементов в ГХ;

d, d_1 - рабочие переключатели (начальные значения $d=0$, $d_1=0$);

$S_{уб}$ - список вершин, которые будут удалены, (определяется в ходе работы алгоритма);

$S(B_j)$ - конститuenty, входящие в выражение B_j ;

M - каталог с числом элементов, равным N; $M(B_j)$ - соответствует вершине B_j ; $M(B_j)$ может принимать значения 0, 1, 2;

$M_1(B_j)$ - часть каталога M, соответствующая тем вершинам $S(B_j)$, для которых значение M не равно 2.

Опишем отдельные блоки алгоритма.

Бл. 4. Порядок элементов A соответствует порядку этих конститuent в ГХ.

Бл. 6. Сообщается проектировщику, какие конститuenty из СК не попали в A.

Бл.10. Проверяется, является ли вершина B_j возможно удаляемой.

Бл.12. Все вершины из $S(B_j)$ помечаются как остающиеся.

Бл.13. Проверяется, что не все вершины графа просмотрены.

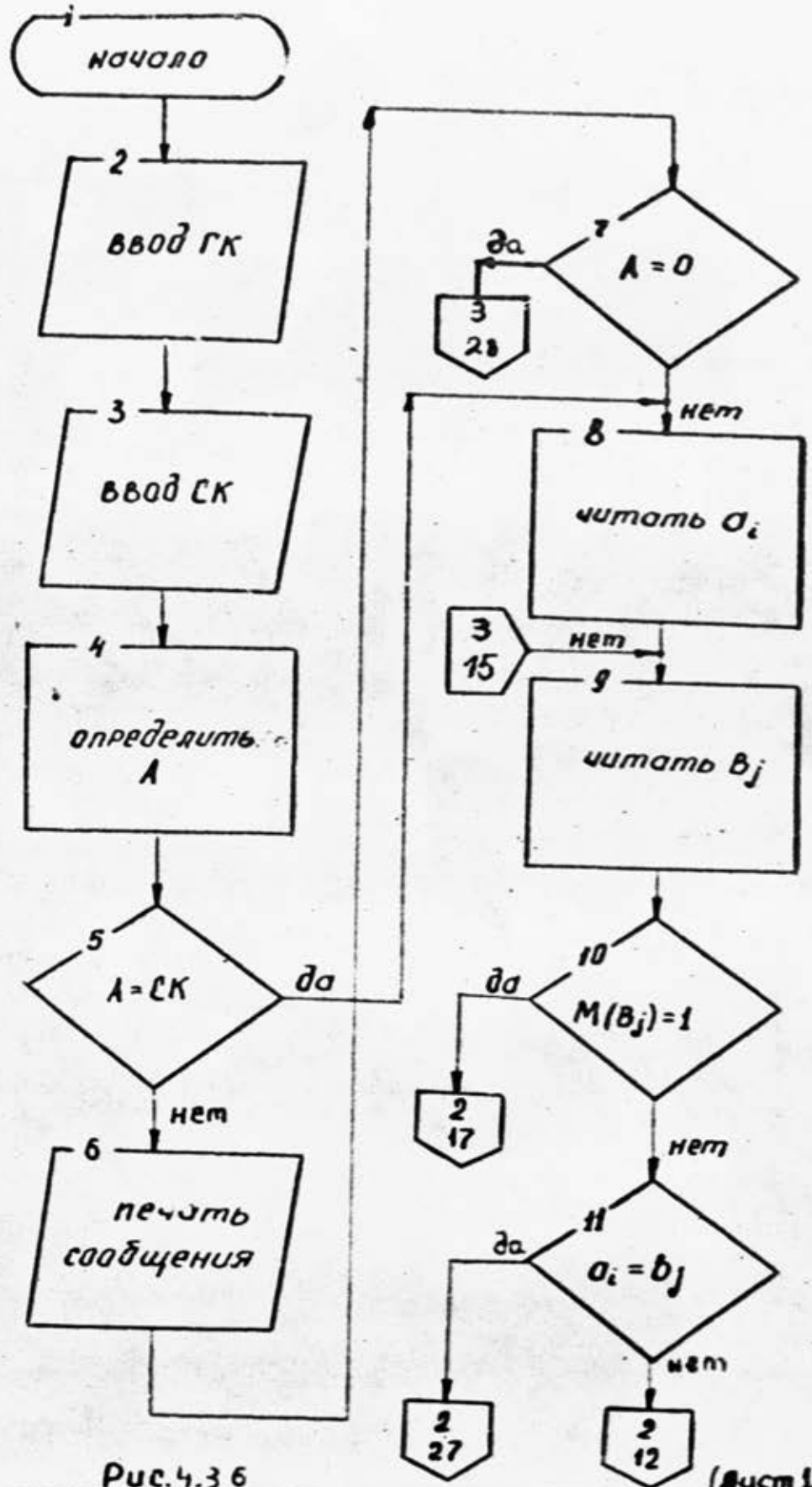


Рис. 4.36

(лист 1)

15-2-78
ТПЗЧ2

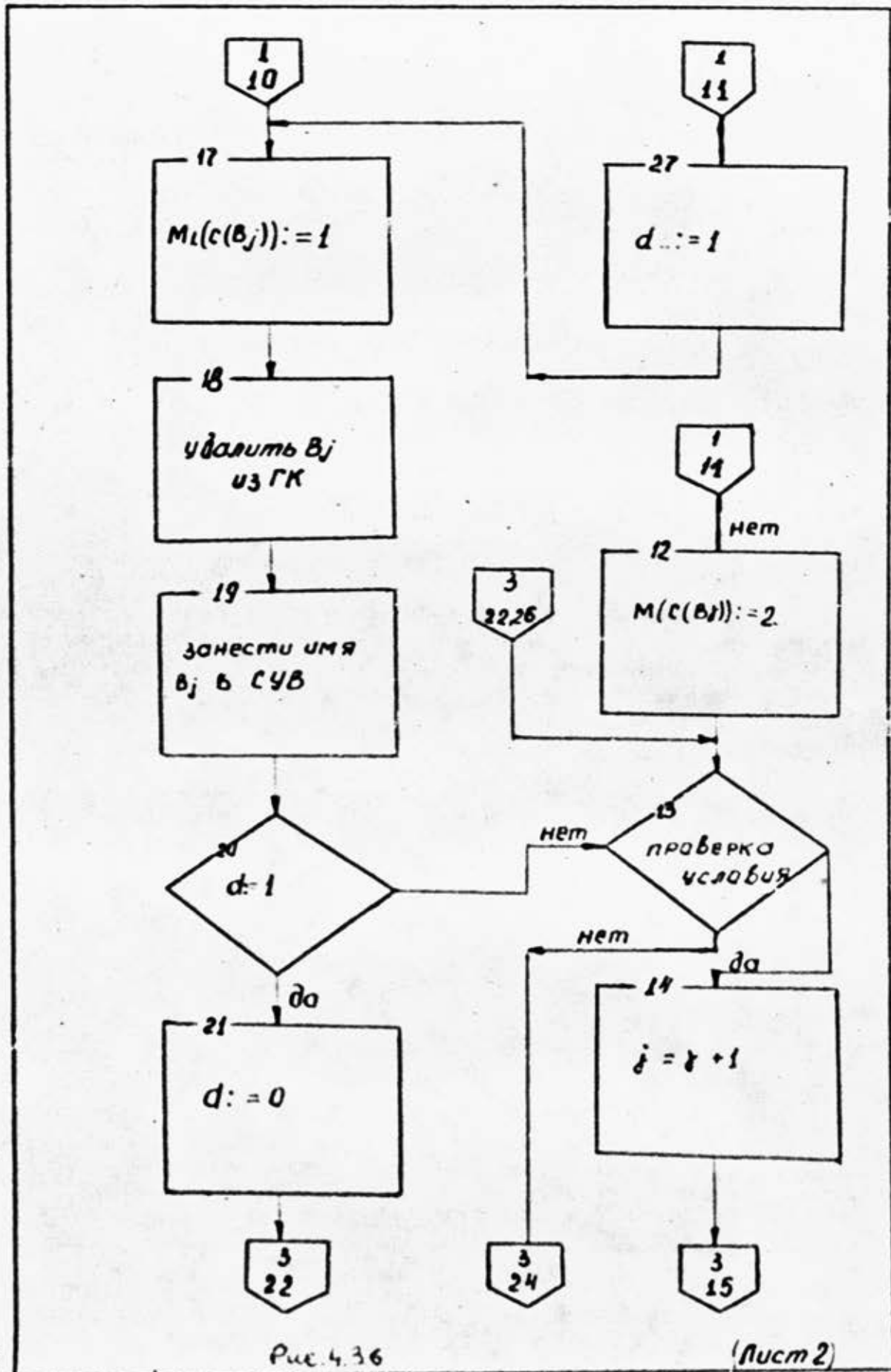
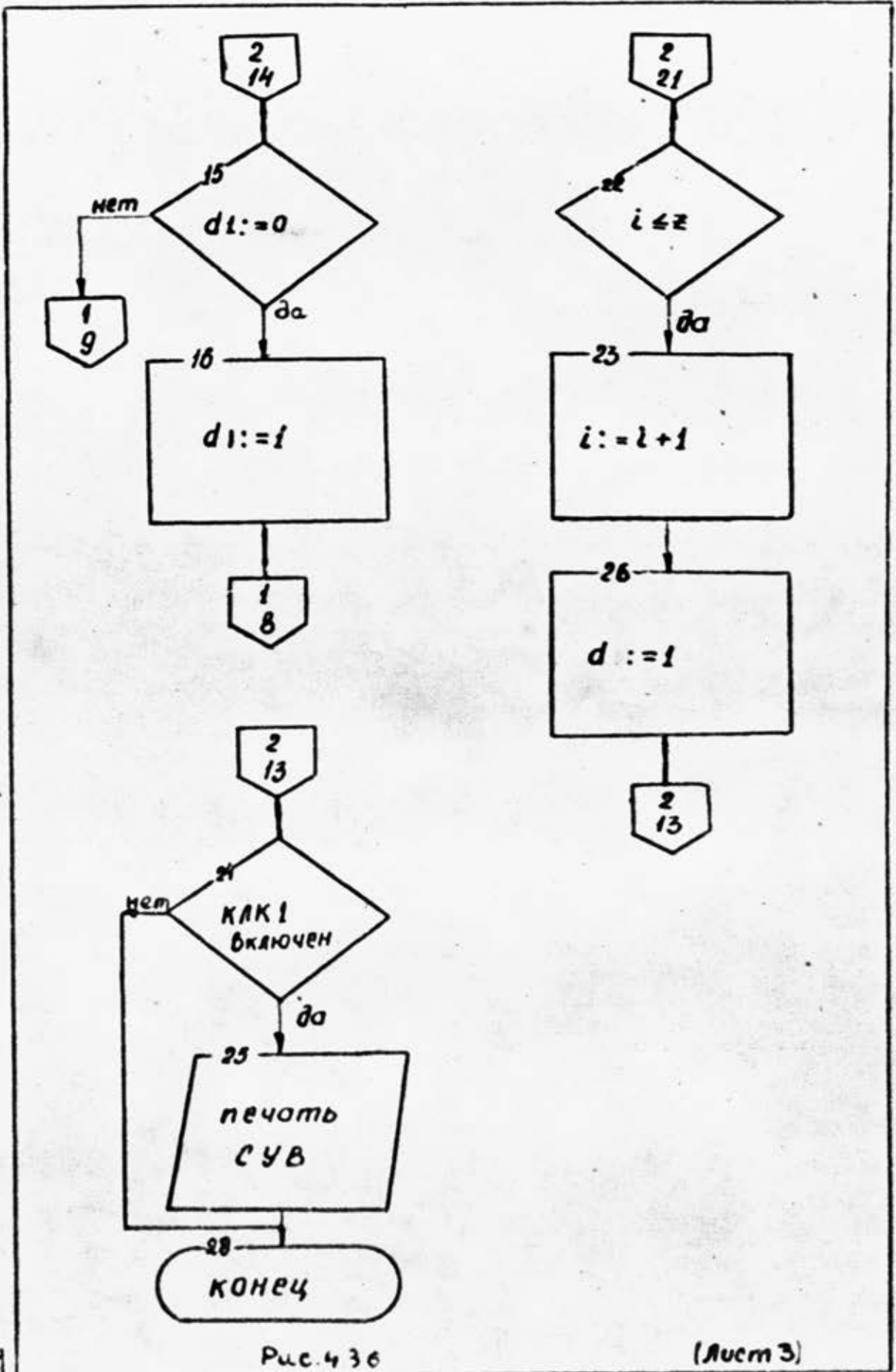


Рис. 4.36

(Лист 2)

15-2-3
ТПТЗен2



15.2.76
ТПТЗКМ2

Рис. 436

(Лист 3)

Бл.17. Все вершины из $S(V_u)$, которые не помечены, как оставляемые, помечаются меткой "возможно удаляемые".

Бл.24. Печатать ли СВВ?

4.3.9: Описание алгоритма модуля дополнительного нижнего замыкания.

Модуль предназначен для определения дополнительного нижнего замыкания.

Модуль вызывается модулями:

- расширения множества интерпретируемых конститuent;
- расширения множества разрешенных термов;
- расширения множества автоматически интерпретируемых конститuent.

Алгоритм реализуется однократным просмотром текущего ГК и ГК ГРС. Блок-схема алгоритма приведена на Рис. 4.3.7. В блок-схеме использованы следующие обозначения:

СК - список конститuent, по которым определяется дополнительное нижнее замыкание;

A - часть конститuent СК, определенных в ГК ГРС;

Z - число конститuent в ГК ГРС;

Z₁ - число конститuent в текущем ГК;

a_i - элемент ГК ГРС $i = 1, 2, \dots, Z$;

b_j - элемент текущего ГК $j = 1, 2, \dots, Z_1$;

S(d_i) - нижние вершины подграфа a_i;

M - каталог; число элементов = Z; каждый элемент двоичный, т.е. принимает значение 0 или 1 (1 означает, что элемент входит в нижнее замыкание);

ТДВ - таблицы вершин, входящих в дополнительное нижнее замыкание.

Описание отдельных блоков алгоритма.

Бл. 4. Элементы A приводятся к порядку, соответствующему их порядку в ГК ГРС.

Бл. 5. Сообщается, какие элементы СК не являются вершинами ГК ГРС.

Бл. 8. $K(a_i)$: = 1, если $a_i \in A$;

$K(a_i)$: = 0; если $a_i \notin A$.

Бл.20. $K(S(a_i))$: = 1.

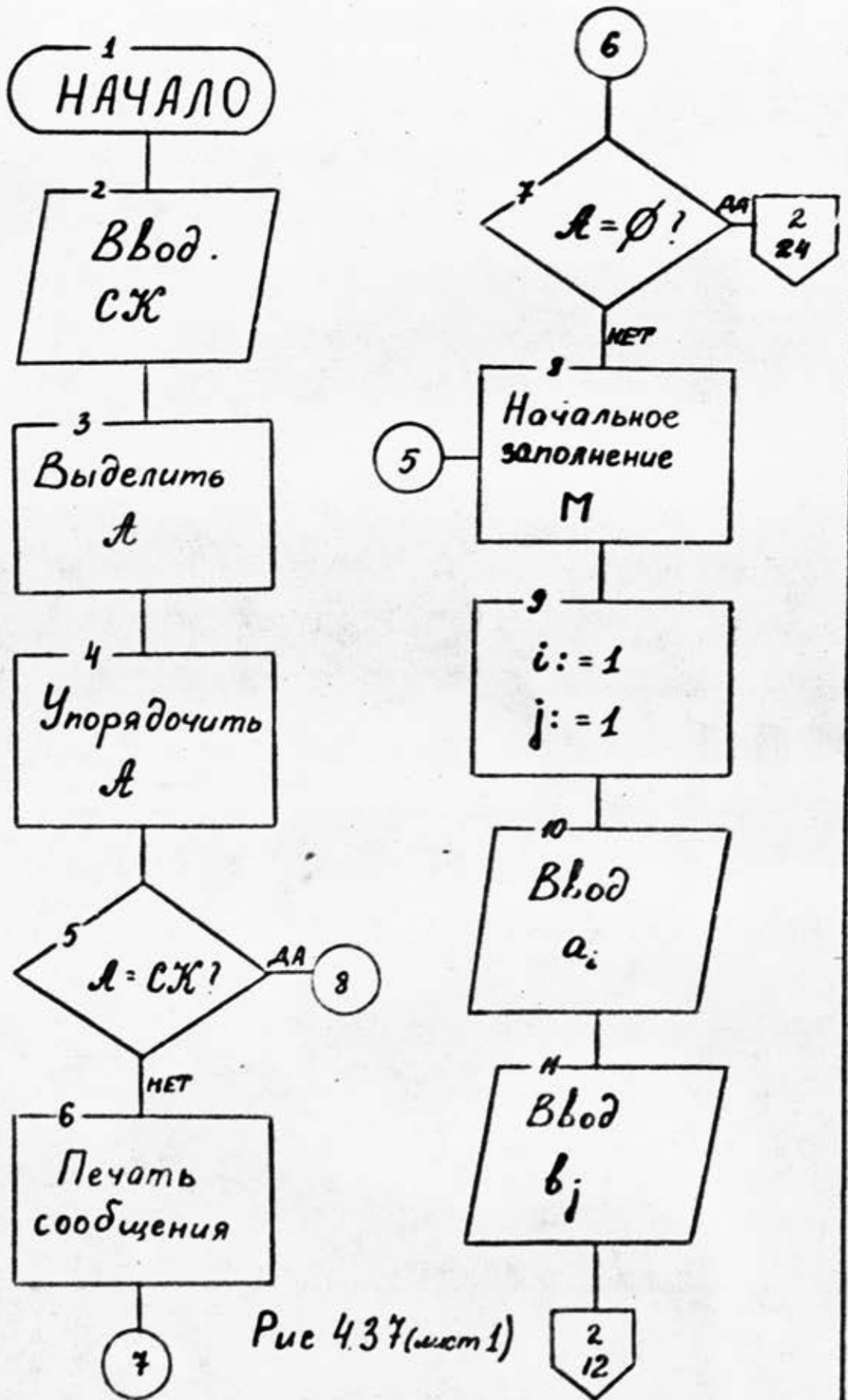


Рис 4.37 (мет 1)

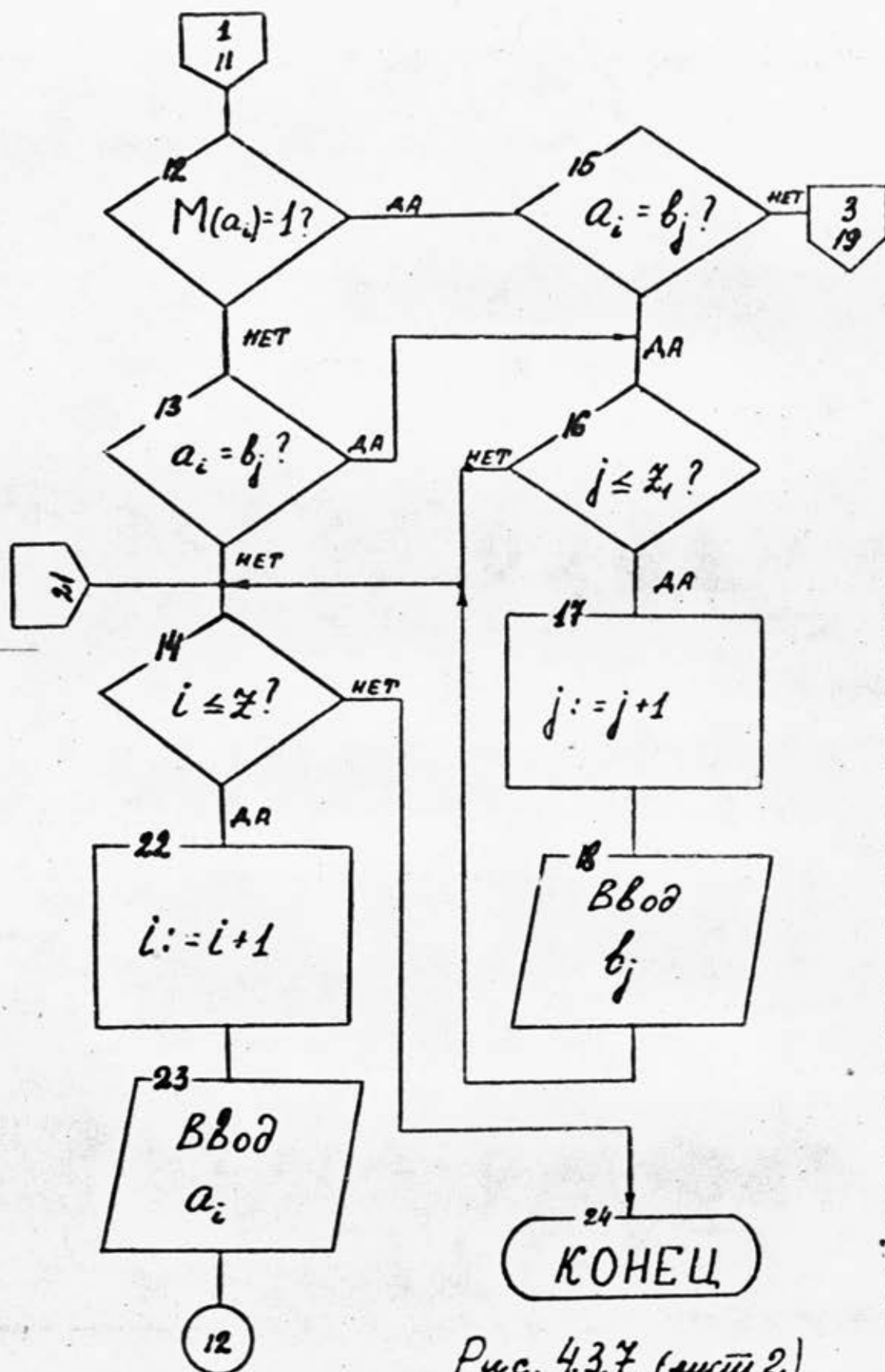


Рис. 4.3.7 (закінч. 2)



Рис. 4.3.7 (лист 3)

Бл.21. Пополнение текущего ГК.

4.3.9. Описание алгоритма модуля верхнего замыкания.

Модуль предназначен для выбора вершин графа, входящих в верхнее замыкание. При этом могут формироваться таблицы, где для вершин из верхнего замыкания указаны нижние вершины их подграфов, входящие и не входящие в верхнее замыкание.

Модуль вызывается управляющей программой пакета, модулем сокращения множества REL, модулем сокращения операционной схемы.

Входной информацией модуля является список вершин графа, по которым производится верхнее замыкание и ГК.

Блок-схема алгоритма приведена на Рис. 4.3.8. В блок-схеме приняты следующие обозначения:

ТВЗ - таблица задания верхнего замыкания;

ТВЗ1 - часть ТВЗ, определенная в ГК текущем;

M - каталог, каждый элемент которого принимает значение 0,1. Число элементов каталога равно 2^z ;

z - число элементов ГК;

a_i - i -тый подграф ГК;

$c(a_i)$ - нижняя вершина подграфа a_i ;

$d_1(a_i)$ - часть нижних вершин подграфа a_i , не входящая в ВЗ;

$d_2(a_i)$ - часть нижних вершин подграфа a_i , входящая в ВЗ;

ТРА - таблица разрешенных аргументов, где для $q_i \in \text{ТРВЗ}$ указаны вершины $d_1(q_i)$;

ТНРА - таблица неразрешенных аргументов, где для $q_i \in \text{ТРВЗ}$ указаны вершины $d_2(q_i)$;

ТРВЗ - таблица результата верхнего замыкания. Таблица состоит из вершин, входящих в верхнее замыкание.

Опишем отдельные блоки алгоритма.

Бл. 4. $M(a_i) = 1$, если $a_i \in \text{ТВЗ1}$;

$M(a_i) = 0$, если $a_i \notin \text{ТВЗ1}$.

Бл. 7. Переход на Бл.15 производится тогда, когда $M(c(a_i)) = 0$ для всех нижних вершин подграфа a_i .

Бл.10: Ключ определяет, производить ли заполнение таблиц ТРА и ТНРА.

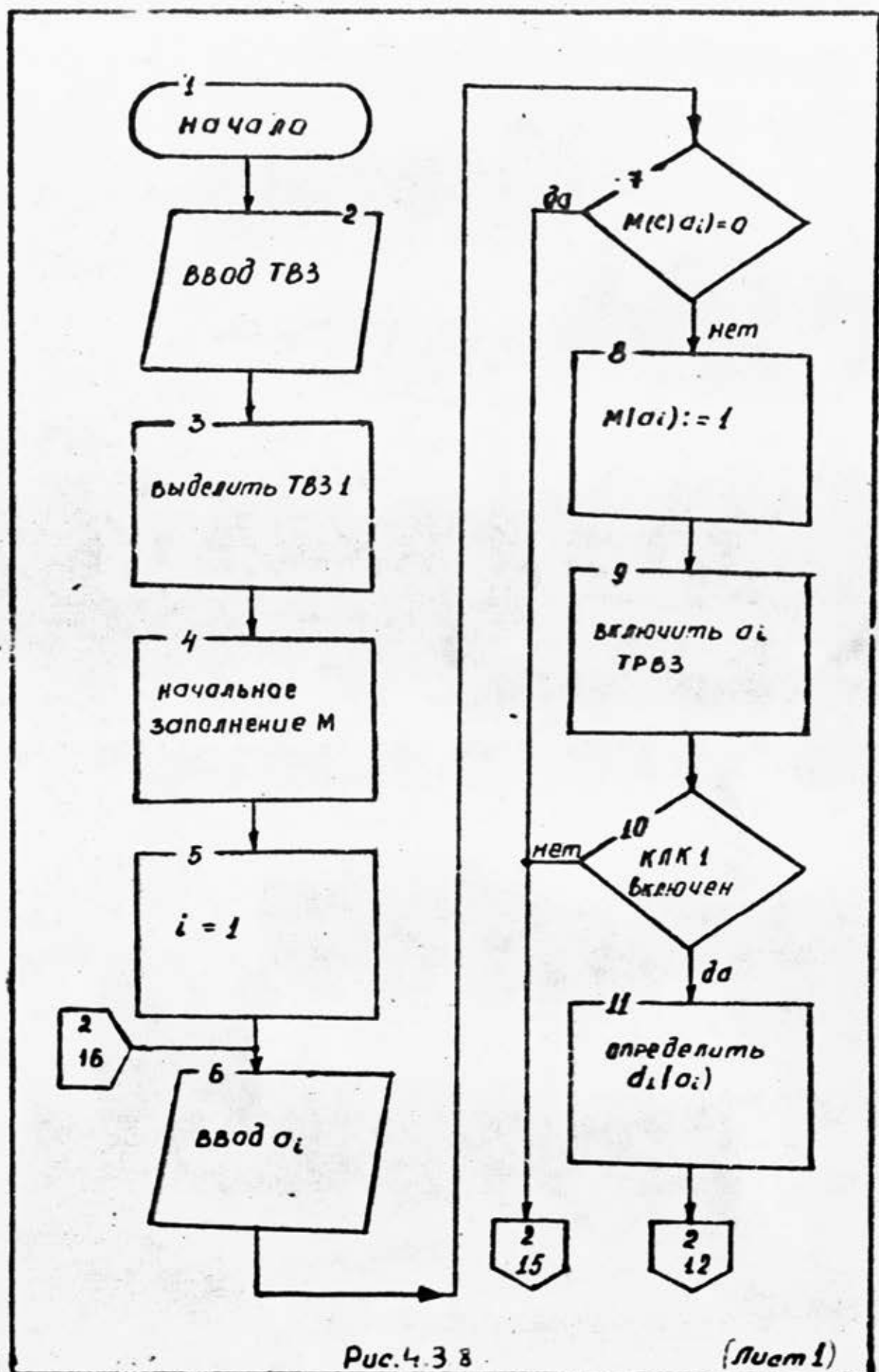
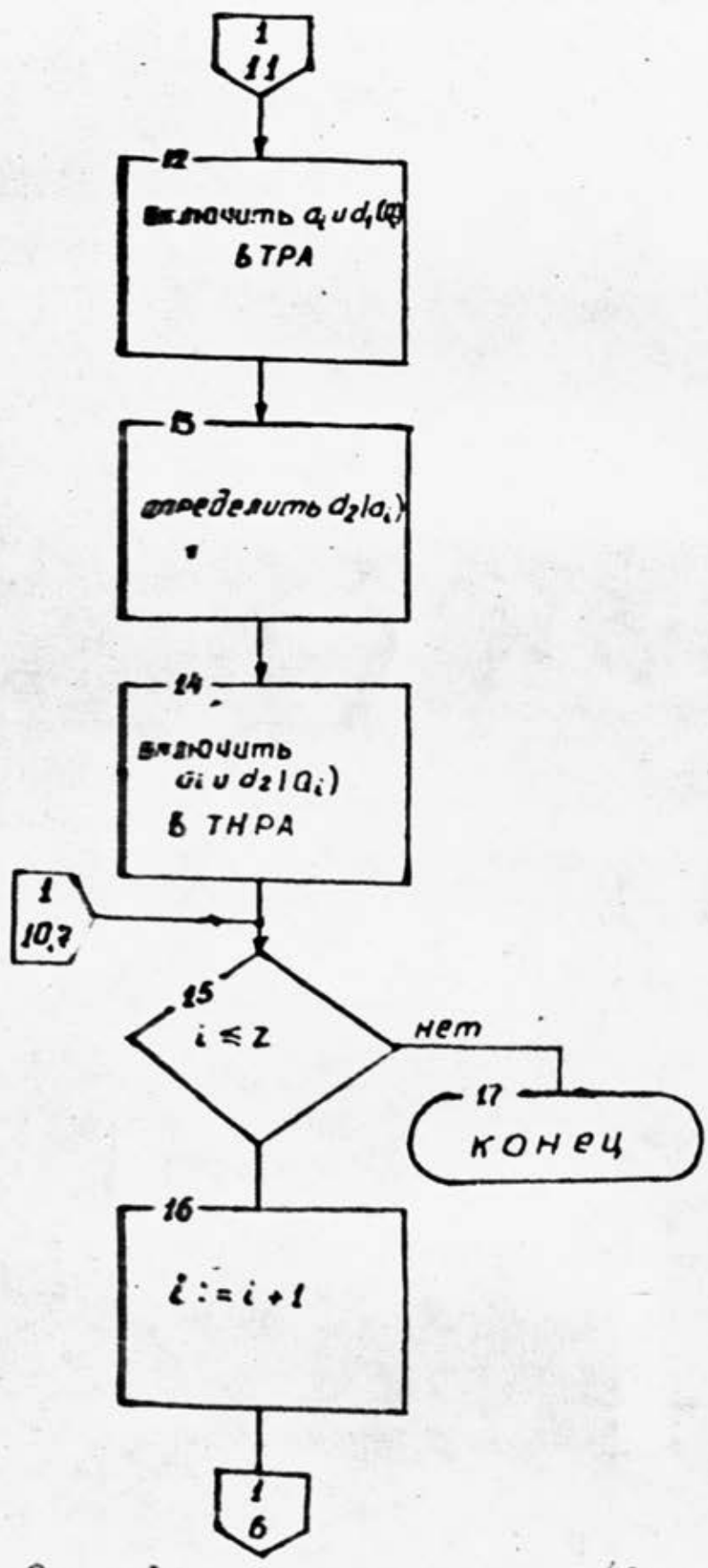


Рис. 4.38

(лист 1)

ТМЗ кн 2



4.3.10. Описание алгоритма модуля исключения вершин из графа R -интерпретации.

Модуль предназначен для исключения одной (или нескольких) конституэнт(т) (m), в развернутом или неразвернутом в подграф виде, из графа R -интерпретации.

Модуль вызывается модулем сокращения разрешающих термов и модулем раскрытия множества специнтерпретируемых конституэнт.

Исходные данные:

- список исключаемых конституэнт;

- граф R -интерпретации;

- таблица соответствия (ТС), выработываемая модулем удаления дублирующих подграфов в пакете R -интерпретации.

Результатом работы модуля является измененный ГК и список удаленных подграфов (СУПГ).

Блок-схема алгоритма приведена на Рис.4.3.9. В блок-схеме приняты следующие обозначения:

СИК - список исключаемых конституэнт;

ТУ - таблица удаления;

ТИУ - таблица имен удаляемых подграфов;

ТС - таблица соответствия; состоит из пар вида (Y, M) (Y, M), где Y - заменяемое имя, M - заменяющее имя (подробнее см. в пакете R -интерпретации);

a_{ik} - k -ое имя из i -ого слоя ТУ;

$B(a_{ik})$ - все пары из i -ого слоя ТС, в которых встречается имя a_{ik} ;

$B(a_{ik})$ - подмножество $B(a_{ik})$, состоящее из пар (\dots, a_{ik}) , в котором все пары типа (j, a_{ik}) заменены на пары (j, d) , $(d, a_{ik}) \in B(a_{ik})$;

z_i - число подграфов на i -том слое GR ;

d - имя подграфа, см. определение $B(a_{ik})$;

ТЗИ - таблица заменяемых имен, состоящая из пар (d, d_{ik}) ;

T - максимальный номер необработанного слоя ТУ;

v_{ik} - имена нижних вершин k -ого подграфа i -ого слоя ГК;

$H(v)$ - номер слоя вершины "v";

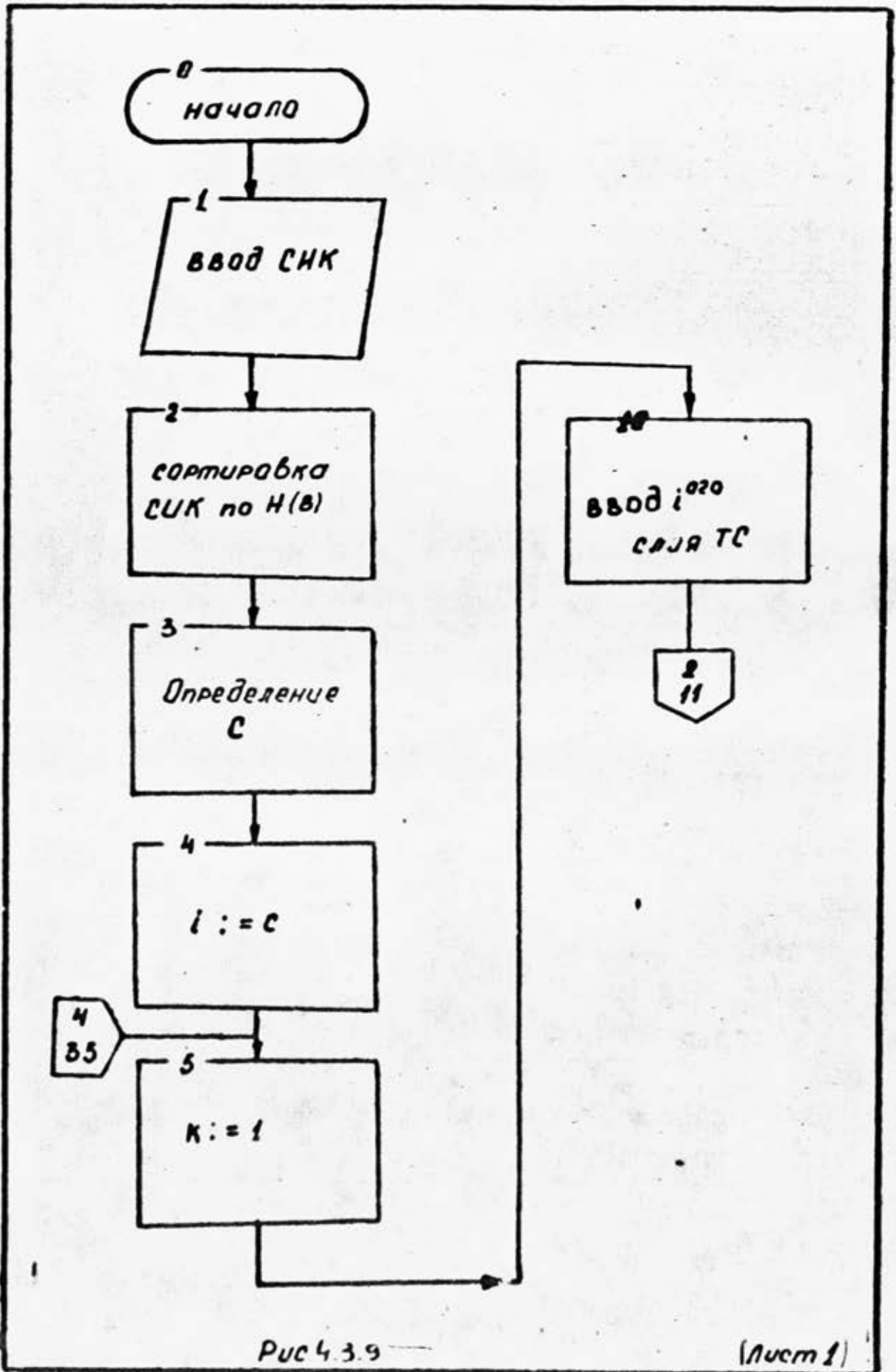


Рис 4.3.9

(Лист 1)

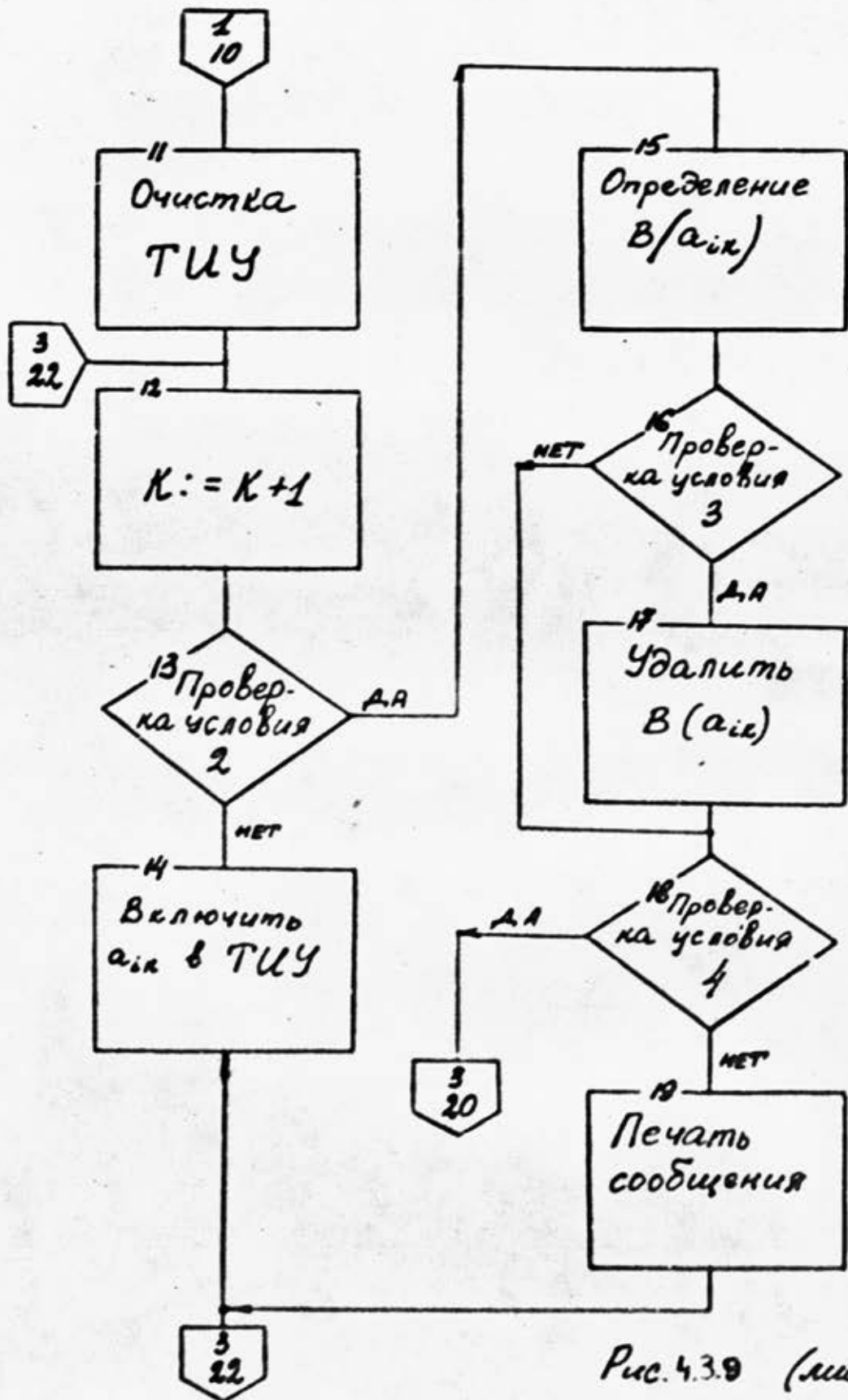


Рис. 4.3.9 (лист 2)

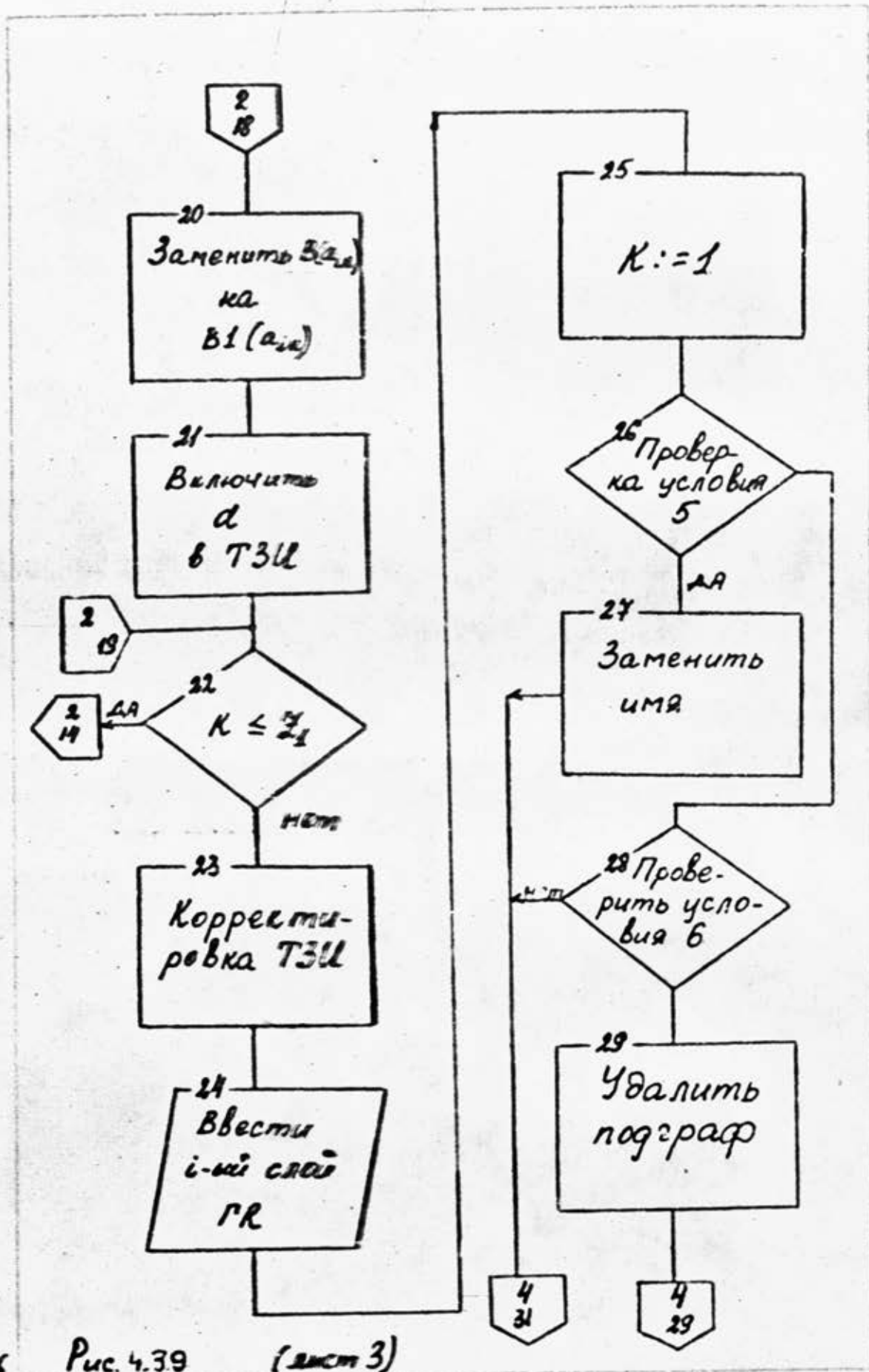


Рис. 4.39 (лист 3)

15-2-76
ТПТЗМ2

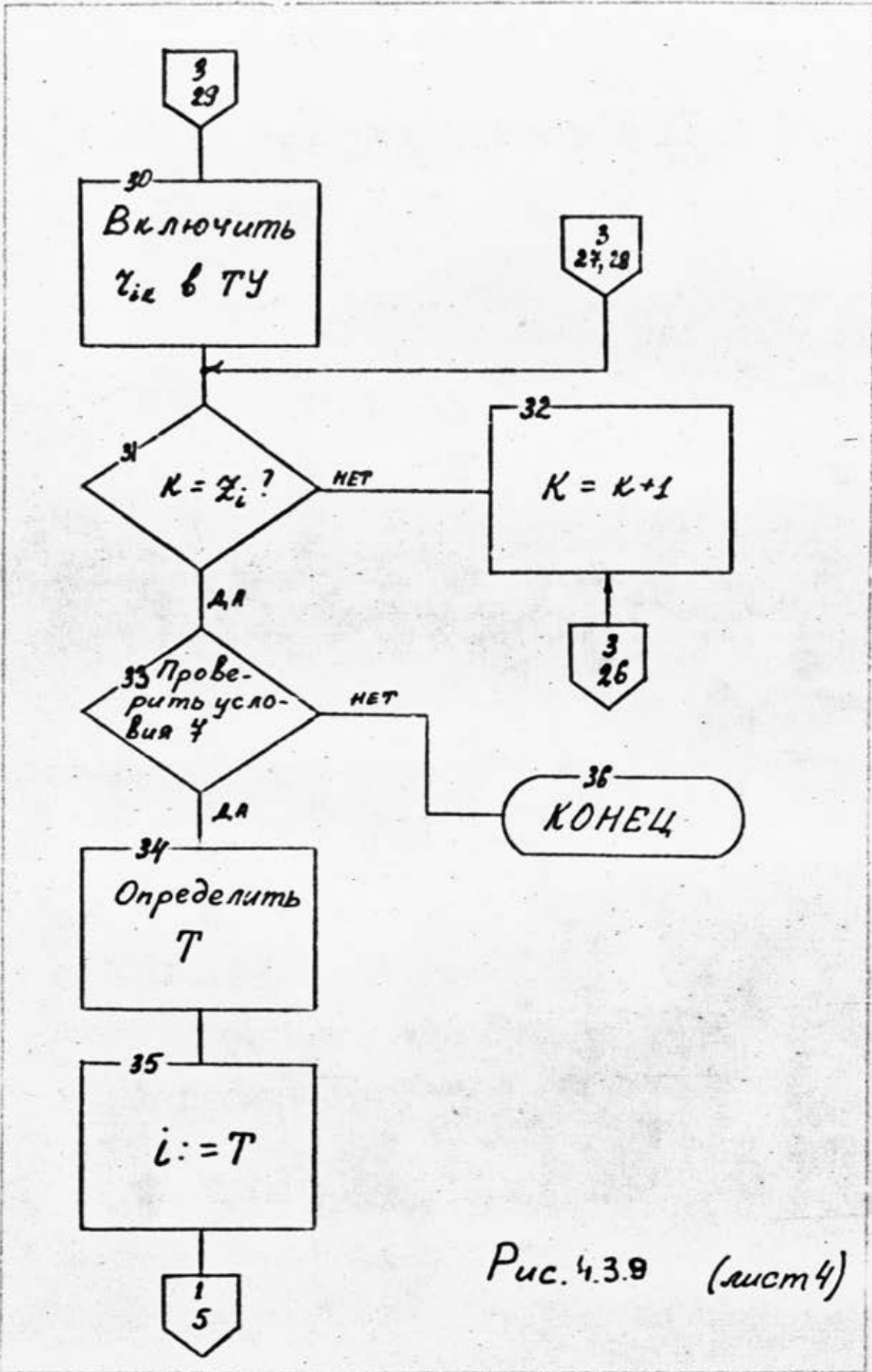


Рис. 4.3.9 (лист 4)

15.2.76
ТПТЗм2

c - максимальный номер слоя в СИК.

Описание отдельных блоков:

Бл. 2. Получение начального заполнения ТУ после сортировки.

Бл. 13. Проверке подлежит условие, что множество $B(a_{ik})$ не пусто.

Бл. 16. Проверке подлежит условие, что множество $B(a_{ik})$ одноэлементное и имеет вид (a_{ik}, \dots) .

Бл. 18. Проверке подлежит условие, что множество $B(a_{ik})$ состоит из пар вида (a_{ik}) .

Бл. 19. Сообщение, что в ТС - ошибочные записи.

Бл. 23. Устраняются цепочки замен, т.е. пары (α, β) , (β, γ) ; (γ, δ) заменяются парой (α, δ) .

Бл. 26. По ТЗИ проверяется, что имя a_{ik} - заменяемое.

Бл. 27. Имя подграфа заменяется на указанное в ТЗИ.

Бл. 29. При удалении подграфа его имя включается в список удаленных подграфов (СУПГ).

Бл. 30. Имена нижних вершин должны быть помещены в слои ТУ, номера которых совпадают с номерами слоев этих вершин.

Бл. 33. Проверке подлежит условие, что не все указанные в ТУ слои обработаны.

4.3. II. Описание модуля расширения графа R -интерпретации.

Данный модуль совпадает с модулем организации вычислительного процесса по вычислительной схеме в пакете R -интерпретации. Поэтому в данном пункте приведена только операционная схема. Она задает схему получения измененных ГК и ГR для расширения множества интерпретируемых конститuent. Все модули в операционной схеме реализованы в пакете R -интерпретации.

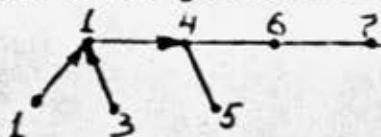


Рис. 4.3.10.

1 - модуль удаления подграфа;

2 - задание на работу модуля удаления подграфа (список

специализируемых конститuent);

3 - ГХ;

4 - модуль управления детализацией подграфа;

5 - S_{An} ;

6 - модуль приведения к расслоенному виду;

7 - модуль удаления дублирующих подграфов.

Подробнее о задаче графа см. 2.2.1.

4.3.12. Описание алгоритма модуля расширения множества интерпретируемых конститuent.

Модуль предназначен для включения в текущий ГХ конститuent из ГХ ГРС. При этом определяются дополнительные конститuent, которые должны быть включены в текущий ГХ. Для обеспечения вычисления заданных конститuent начальный список конститuent вводится по форме

Модуль вызывается управляющей программой пакета. Блок-схема алгоритма приведена на Рис.4.3.12. В блок-схеме использованы следующие обозначения:

СК - список конститuent, которые должны быть включены в ГХ;

A - часть элементов СК, которые определены в ГХ ГРС и не определены в текущем ГХ.

Опишем отдельные блоки алгоритма.

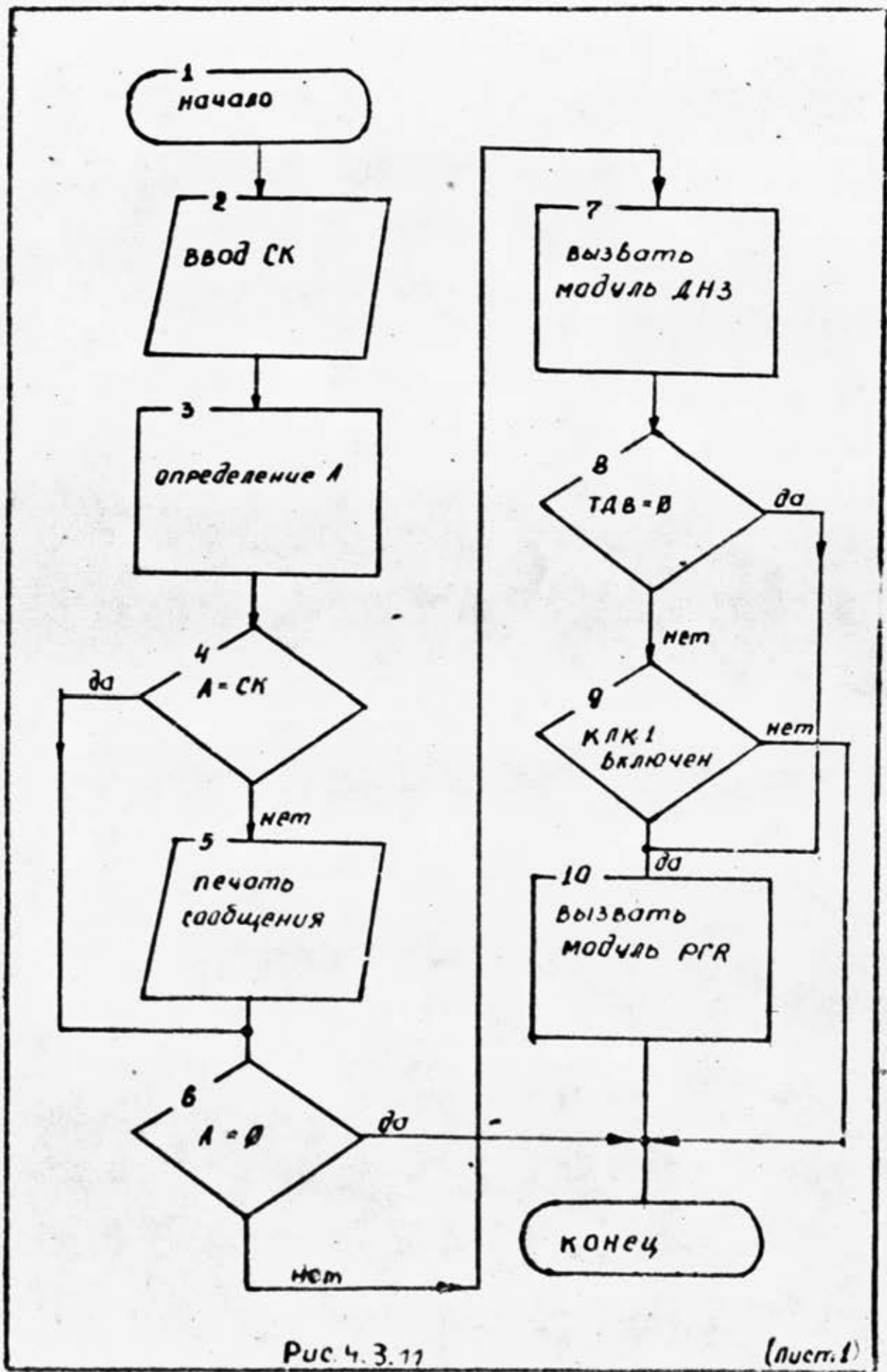
Бл. 5. Сообщается, какие элементы СК не попали в A.

Бл. 9. Проектировщик определяет, производить ли изменения.

4.3.13. Описание алгоритма модуля сокращения множества REL.

Алгоритм предназначен для сокращения множества REL. По заданному списку конститuent производится верхнее замыкание. Список вершин из верхнего замыкания разделяется на 2 подсписка проектировщиком: 1-ый определяет вершины, которые должны быть удалены; 2-ой - вершины, предназначенные для специализации.

Алгоритм применим как к графу конститuent, так и к графу



ТП-3м2

Рис. 4.3.11

(лист. 1)

R-интерпретация.

Модуль вызывается управляющей программой пакета.

Первоначальный список конститuent вводится по форме 4.

Список конститuent, предназначенный для специнтерпретации, вместе с разрешающими терминами вводится по форме 5.

Блок-схема алгоритма приведена на Рис. 4.3.12. В блок-схеме приняты следующие обозначения:

СК - список конститuent (задание на работу модуля);

А - часть СК, определенная в ГК;

В - список конститuent из ТРВЗ, предназначенных для специнтерпретации;

С1 - часть конститuent из В, не являющихся специнтерпретируемыми в старом ГК;

С2 - часть конститuent В, являющихся специнтерпретируемыми в старом ГК;

Д - часть конститuent из А, не входящая в В;

Т - список удаленных подграфов в результате работы модуля.

Опишем отдельные блоки алгоритма.

Бл. 5. Сообщается об элементах СК, не входящих в А.

Бл. 8. Проектировщик определяет, надо ли производить изменения.

Бл.11. Определяется, что подвергается изменениям: ГР или ГК.

Бл.21. Входом в модуль ИВГР является СУВ и Д.

Бл.24. Модуль УП (удаления подграфа) заимствуется из пакета R-интерпретации.

4.3.14. Описание алгоритма модуля расширения множества автоматически интерпретируемых конститuent.

Модуль предназначен для расширения множества автоматически интерпретируемых конститuent в графе R-интерпретации.

Модуль вызывается управляющей программой пакета.

Входной информацией, вводимой самим модулем, является список расширения, вводимый по форме 4.

Блок-схема алгоритма приведена на Рис. 4.3.13. В блок-схеме приняты следующие обозначения:

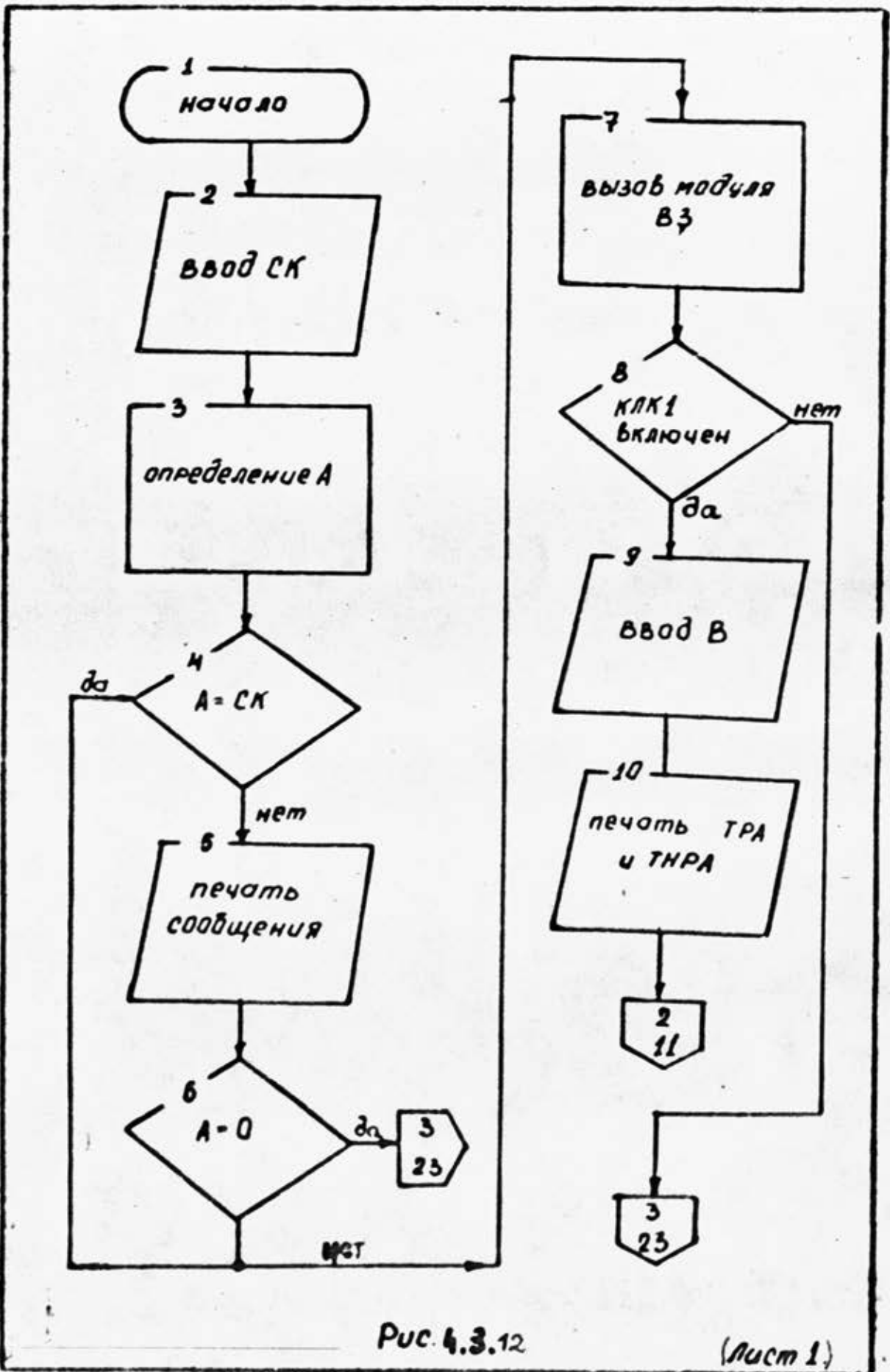


Рис. 4.3.12

ТПТЗКМ 2

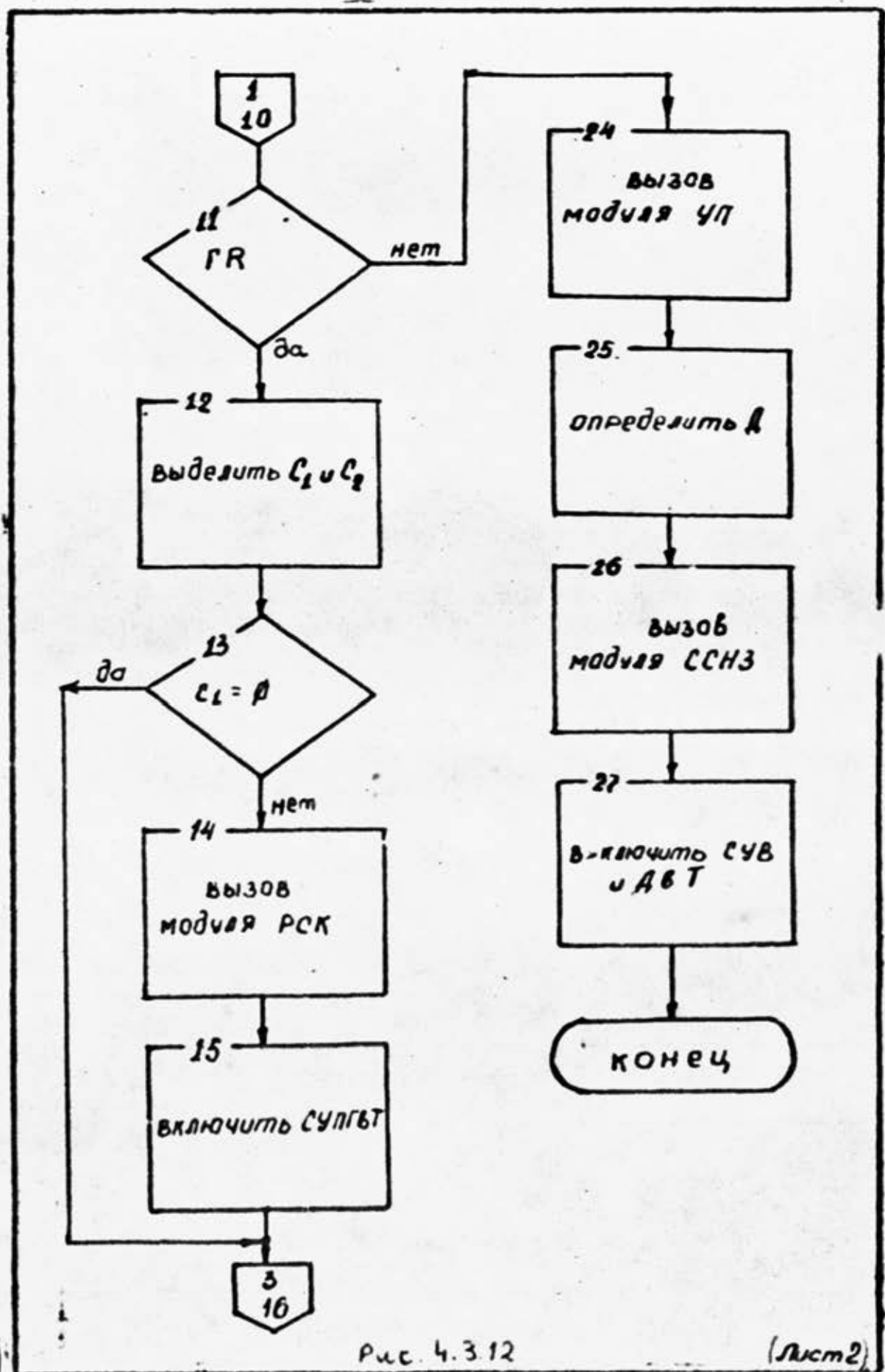


Рис. 4.3.12

(Лист 2)

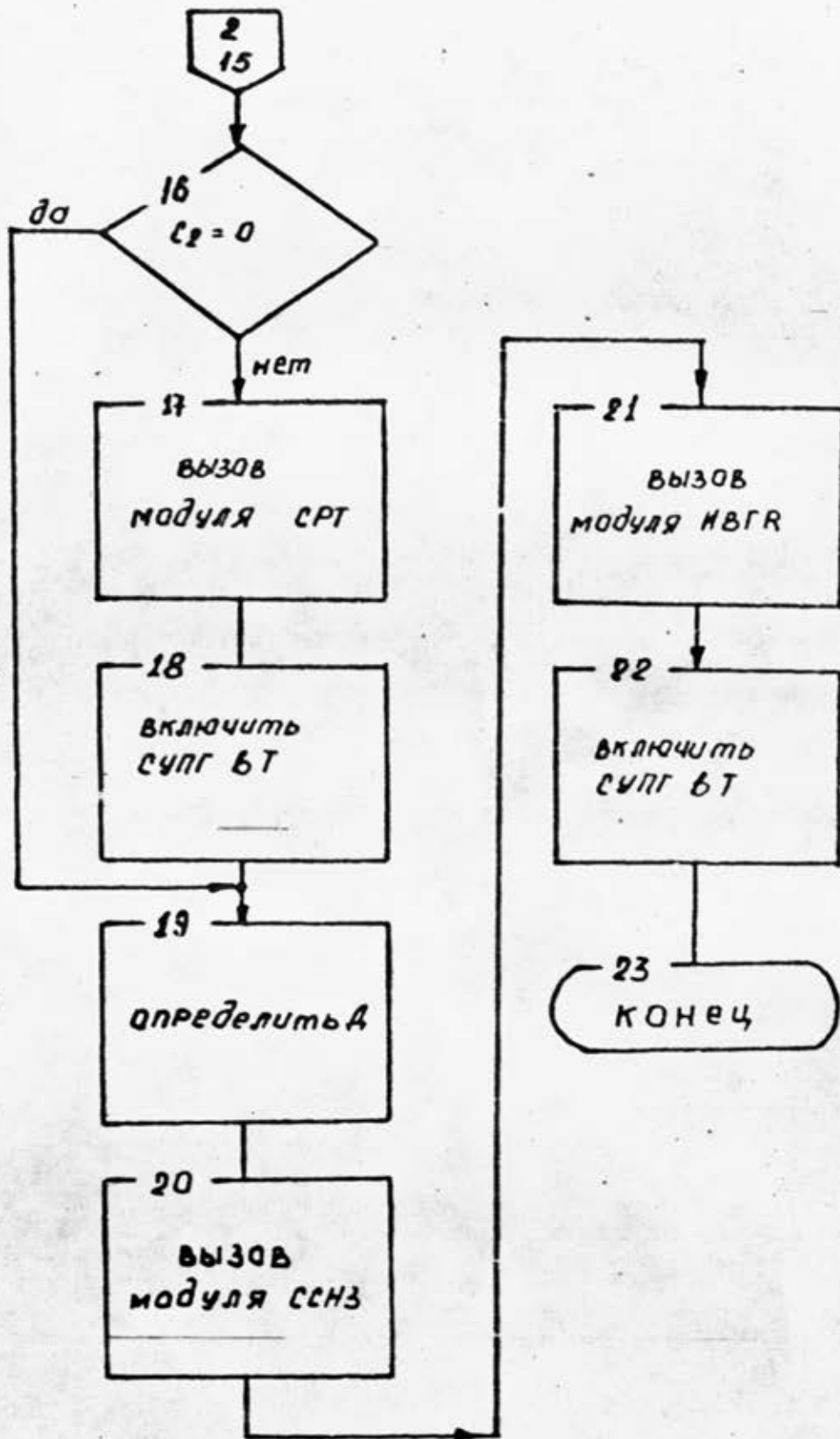


Рис 4.3.12

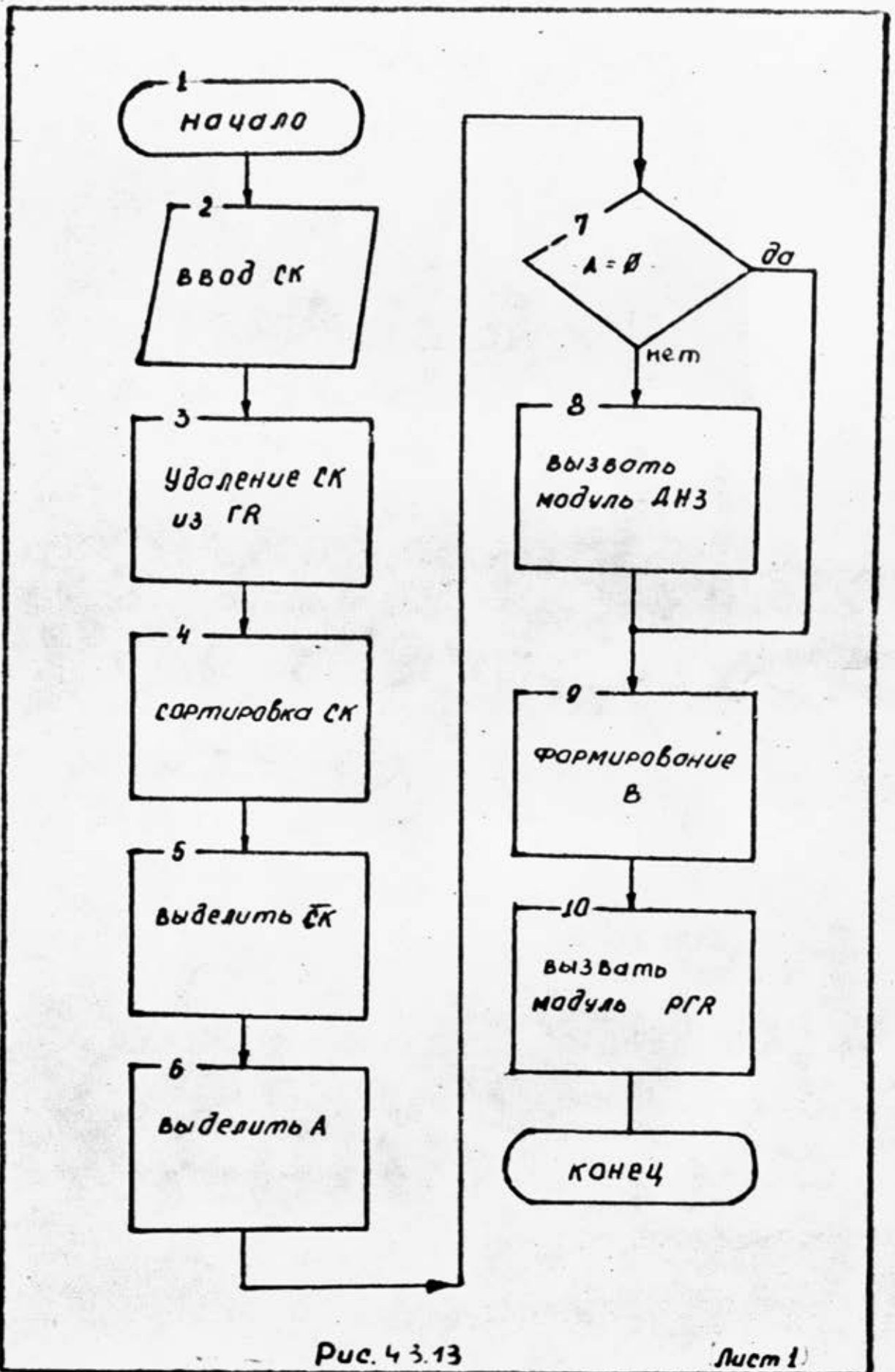


Рис. 4 3.13

$СК$ - список расширения множества автоматически интерпретируемых конституэнт;

$\overline{СК}$ - выражения конституэнт из $СК$;

A - множество термов, входящих в $СК$ и не являющихся разрешающими;

B - список имен конституэнт, включающий $СК$ и ТДВ (ТДВ-результат работы модуля ДНЗ);

PR - расширение графа R -интерпретации.

Опишем отдельные блоки алгоритма.

Бл. 3. Удаляются из слоев PR записи, соответствующие конституэнтам, указанным в $СК$.

Бл. 4. После определения номеров слоев для элементов $СК$, производится сортировка $СК$ по номерам слоев.

Бл. 5. Сортировка $СК$ позволяет определить $СК$ путем однократного просмотра PR .

Бл. 6. При выделении A используются записи, выделенные в блоке 3, и выделяются термы в $\overline{СК}$.

Бл. 8, 10. Модули ДНЗ и PR описаны в данном пакете.

4.3.15. Описание алгоритма модуля расширения множества специнтерпретируемых конституэнт.

Модуль предназначен для расширения множества специнтерпретируемых конституэнт после получения графа R -интерпретации.

Модуль вызывается управляющей программой пакета и модулем сокращения множества Rel . На вход модуля подается список расширения множества специнтерпретируемых конституэнт, заданных по форме 5 или во внутреннем виде, который определен вызвавшим его модулем.

Результатом работы модуля является изменение PR и СДНЗ. СДНЗ вырабатывается модулем ИВГК.

Блок-схема алгоритма приведена на Рис. 4.3.14. В блок-схеме использованы следующие обозначения:

$СК$ - список расширения множества специнтерпретируемых конституэнт;

A - подмножество $СК$, элементы которого определены в PR ;

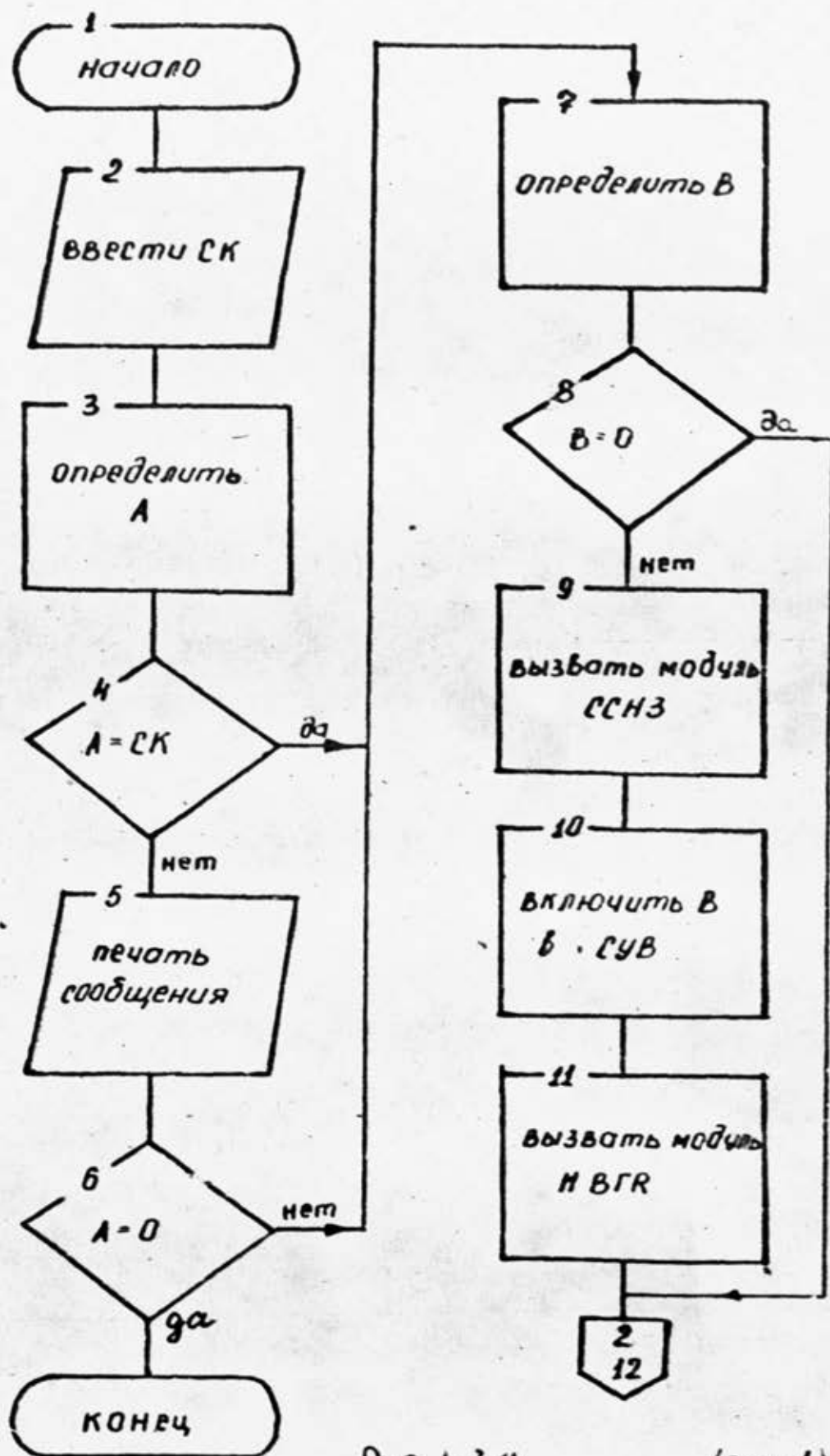


Рис. 4.3.14

(Лист 1)

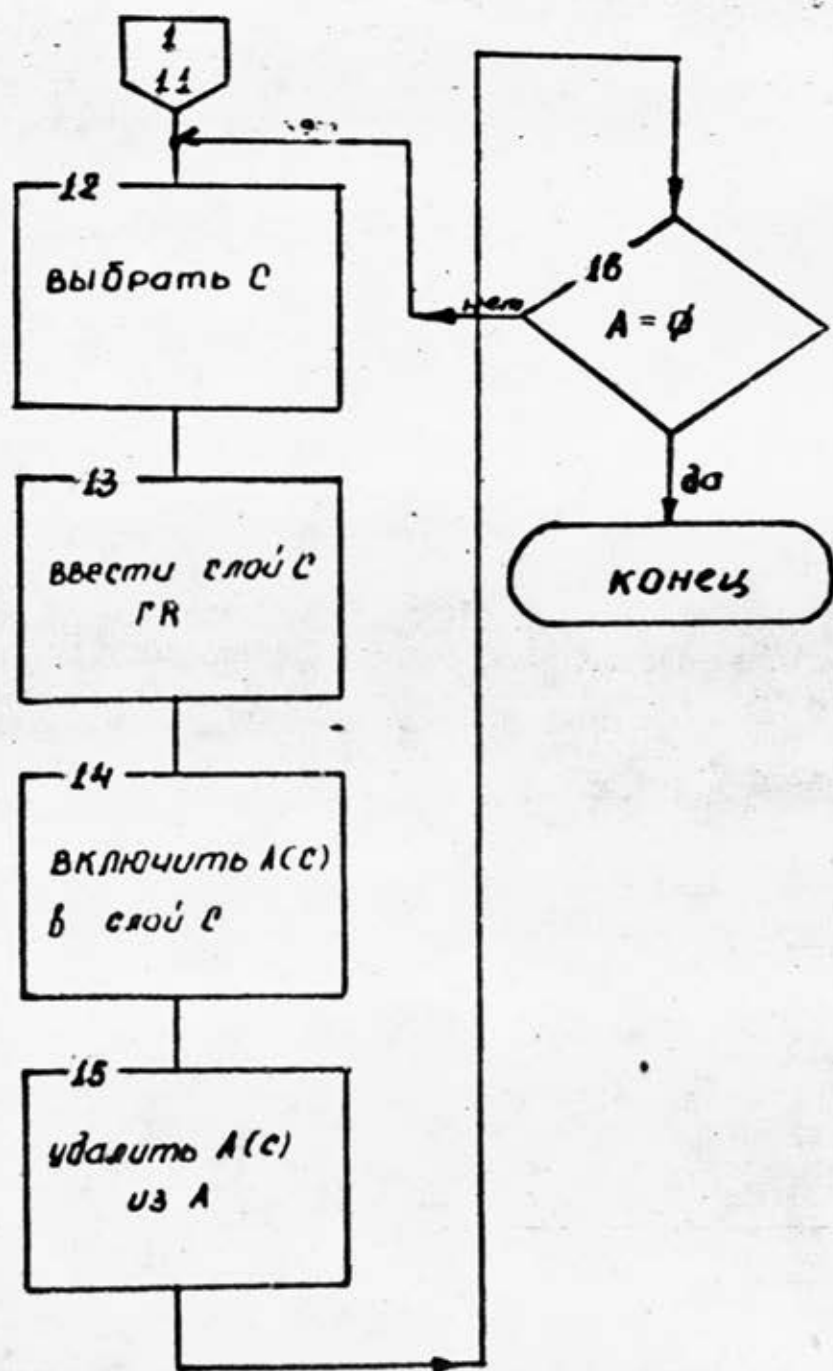


Рис 4.3.14

(Лист 2.)

В - часть термов, входящих в выражения конститuent из **A** и не являющихся разрешаемыми; организованная в виде таблицы

СУВ - список, получаемый после работы модуля ССПЗ;

С - максимальный номер слоя в "В";

A(c) - часть **A**, расположенная на слое **С**.

Описание отдельных блоков алгоритма.

Бл. 4. Сообщение, какие элементы СК не попали в **A**.

Бл. 9. Определение номеров слоев для всех инициальных элементов **В**.

Бл. 10. Включение на соответствующие слои **СУВ** элементов **В**;

Бл. 11. На вход модуля исключения конститuent из графа **R**-интерпретации подается **СУВ**.

4.3.16. Описание алгоритма модуля сокращения разрешаемых термов.

Алгоритм предназначен для сокращения разрешаемых термов у специнтерпретируемых конститuent в графе **R**-интерпретации.

Модуль вызывается управляющей программой пакета и модулем сокращения множества .

Модуль вводит список конститuent с их разрешаемыми термами по форме 5, либо получает их от вызвавшего его модуля. Результат работы модуля: модифицированный **ГР** и (если модуль **ИВГР** вызывался) **СУПГ**.

Блок-схема приведена на Рис. 4.3.15. В блок-схеме использованы следующие обозначения:

СК - список конститuent (задание на работу модуля).

Конститuent могут быть записаны в произвольном порядке:

A - конститuent, входящие в **СК** и в **ГР**;

в - элемент **СК**;

Н(в) - номер слоя "в";

с - максимальный номер "в" в **СК**;

ТВБ - таблица сокращения; содержит имена конститuent, которые должны быть удалены из **ГР**;

τ(в) - множество старых разрешаемых термов конститuent "в";

К(в) - множество новых разрешаемых термов конститuent

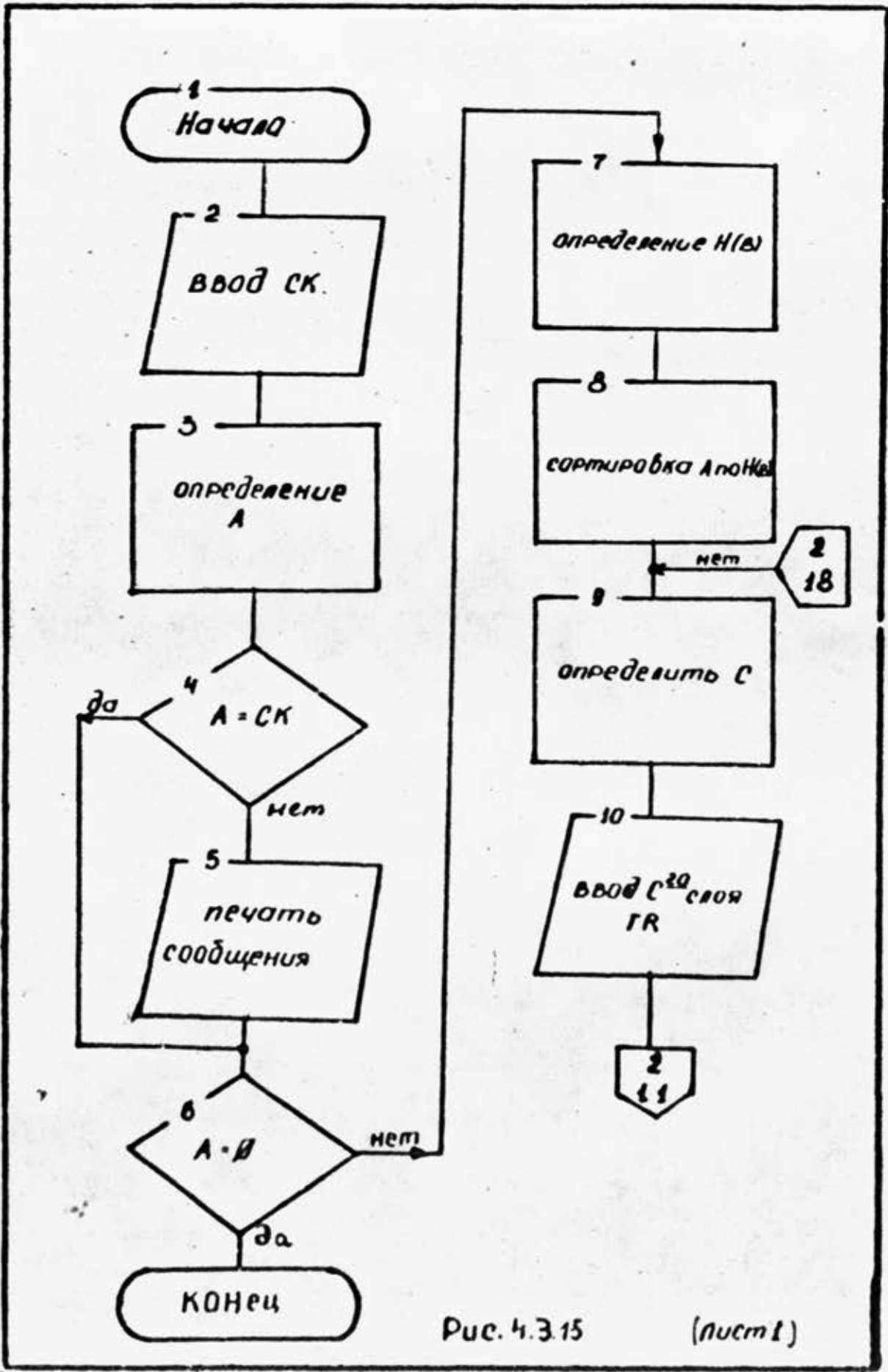


Рис. 4.3.15

(лист 1)

ГПЗм2

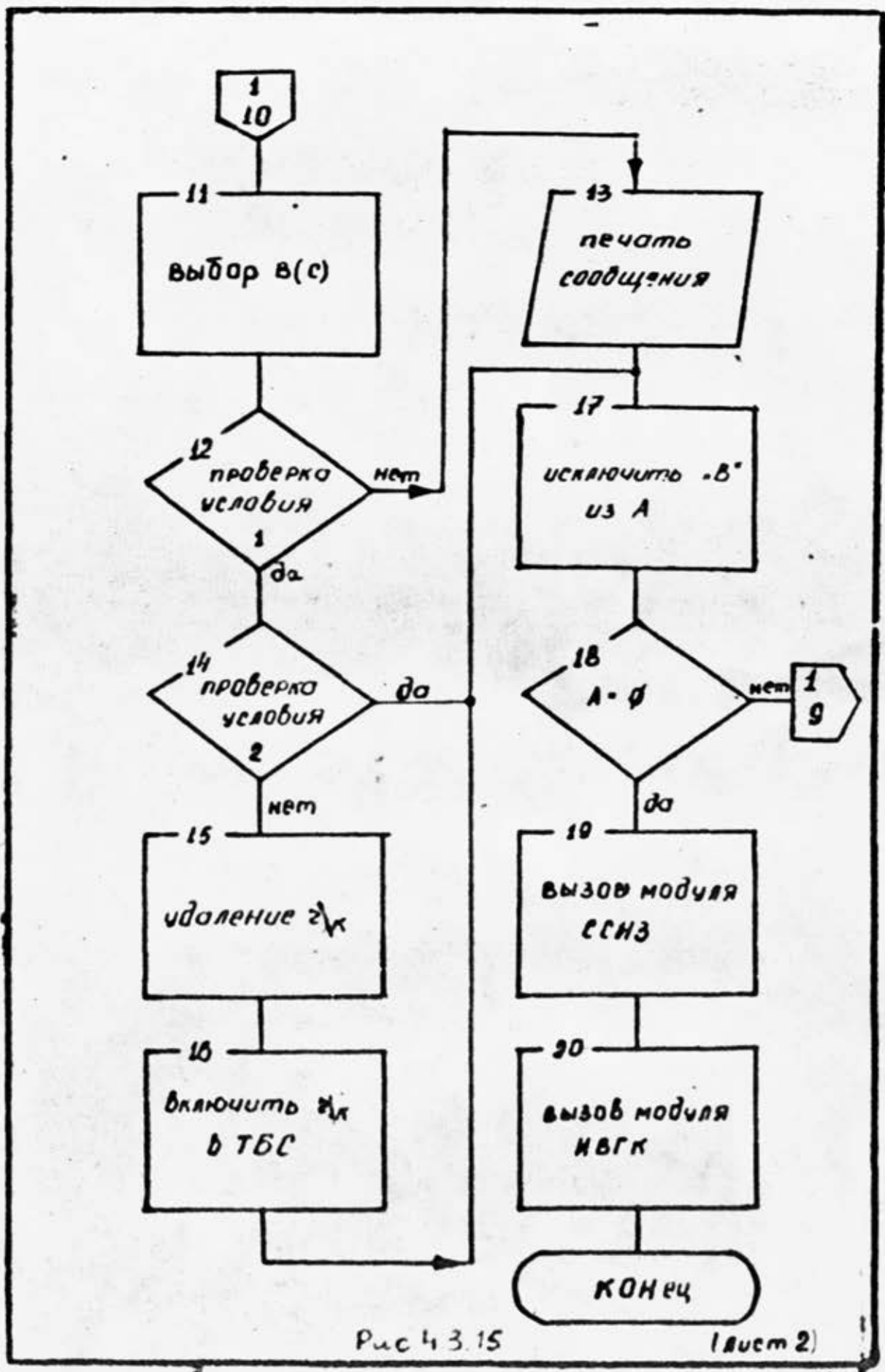


Рис 4.3.15

(лист 2)

7553
Т/ГЗен2

"в";

ССНЗ - сокращения сильного нижнего замыкания;

УВГР - исключения конституент из графа R-интерпретации.

Опишем отдельные блоки алгоритма:

Бл. 5. Сообщается в конституэнтах из СК, не входящих в А

Бл. 11. Поиск "в" на слое с номером "с".

Бл. 12. Проверяется, что $K(v) \subset Z(v)$.

Бл. 13. Сообщается о конституэnte, не удовлетворяющей условия блока 12.

Бл. 14. Проверяется, совпадают ли $Z(v)$ и $K(v)$.

Бл. 15. Исключаются из соответствующих записей термины из "Z" не попавшие в К.

Бл. 19, 20. Модули, вызываемые этими блоками, описаны в данном пакете.

4.3.17. Описание алгоритма модуля расширения множества разрешающих термов.

Алгоритм предназначен для расширения множества разрешающих термов в указанных специнтерпретируемых конституэнтах графа R-интерпретации.

Модуль вызывается управляющей программой пакета.

Входная информация - список конституэнт с их човыми разрешающими термами. Вводится самим модулем по форме 5.

Блок-схема алгоритма приведена на Рис. 4.3.16. В блок-схеме приняты следующие обозначения:

СК - список конституэнт с разрешающими термами;

А - часть СК, включающая конституэнты, определенные в ГР;

с - максимальный номер слоя конституэнт из А;

$A(c)$ - элементы А, отвечающие слову с;

$A(c)$ - выражения конституэнт $A(c)$;

$B(c)$ - термины, входящие в $A(c)$;

d - новые разрешающие термины, не совпадающие со старыми для $A(c)$;

ТБР - таблица добавленных разрешающих термов;

Т - объединение ТБР и имен конституэнт из А.

Опишем отдельные блоки алгоритма.

В-3-17

ТПТЗкн2

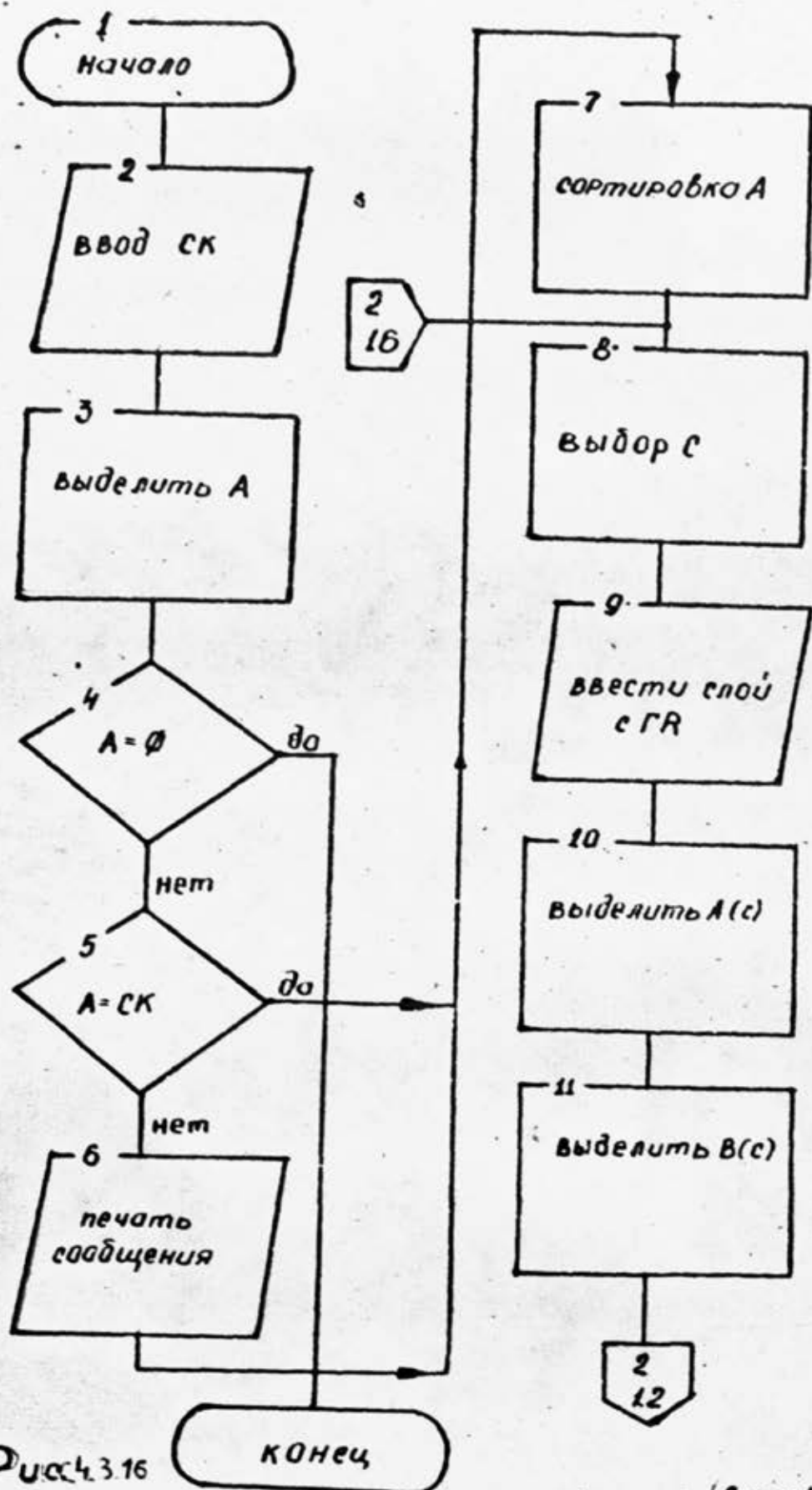


Рис. 4.3.16

(Лист 1)

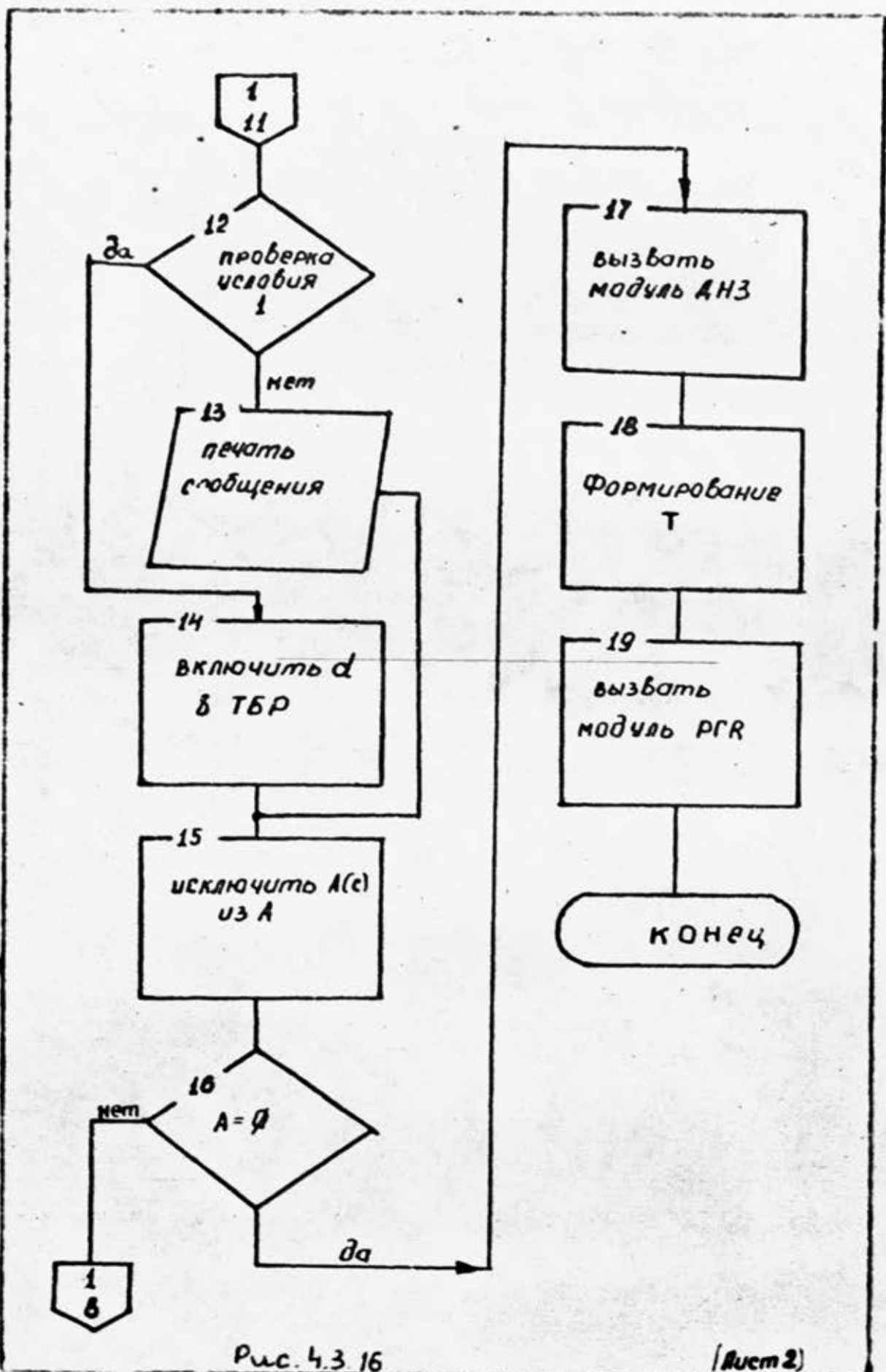


Рис. 4.3 16

(Лист 2)

ТПТЗЕН2

Бл. 6. Сообщается, какие элементы СК не попали в А.

Бл. 7. Сортировка А по предварительно определенным номерам слов.

Бл.12. Проверяется, что разрешающие термы А(с) являются термами В(с).

Бл.13. Сообщается, что А(с) не удовлетворяет условию I.

Бл.17,19. Модули ДНЗ и РГР описаны в данном пакете.

4.3.18. Описание алгоритма модуля изменения операционного заполнения.

Модуль предназначен для изменения операционного заполнения ГР и ГР. Вызывается модуль управляющей программой пакета, указывающей, что должно производиться одно из трех:

- расширение операционного замыкания;
- сокращение операционного замыкания;
- замена операционного заполнения.

В модуле используется модуль операционного заполнения графа из пакета R-интерпретации.

Ввиду простоты алгоритма блок-схема его не приводится. Алгоритм начинает работу с выбора указанного варианта работы. В случае, если указанный вариант не совпадает ни с одним из перечисленных выше, печатается сообщение, и алгоритм заканчивает работу. После выбора варианта производится анализ задания:

- при сокращении и замене проверяется, определены ли указанные конститутенты в графе и привязаны ли им идентификаторы Alg;

- при расширении проверяется, определены ли в графе указанные конститутенты и, что им не сопоставлены идентификаторы Alg.

Из задания выбирается часть, которая удовлетворяет этим условиям. Если эта часть содержит какие-то элементы, то работы алгоритма продолжается.

Для непосредственного расширения или замены операционного заполнения используется модуль операционного заполнения. При сокращении последовательно читаются вершины графа и очищаются поля, заполненные идентификаторами Alg.

4.3.19. Описание алгоритма модуля изменения индивидуальных имен конститuent

Модуль предназначен для изменения одного или нескольких имен конститuent в массиве индивидуальных имен.

Модуль вызывается управляющей программой пакета, которая указывает, что должно производиться сокращение или расширение имен в массиве индивидуальных имен.

Ввиду простоты алгоритма блок-схема его не приводится.

Алгоритм начинает работу с проверки задания.

1. Проверяется, что задано: расширение или сокращение. Это можно реализовать с помощью ТРЖК, которое вводится так же, как и в других модулях пакета.

2. Проверяется, что список имен удовлетворяет требованиям варианта работы алгоритма:

- при сокращении указанные индивидуальные и стандартные имена должны быть в массиве индивидуальных имен;
- при расширении вводимые индивидуальные и стандартные имена не должны быть в массиве индивидуальных имен.

После успешной проверки задания производится пополнение, сокращение части имен массива.

4.3.20. Описание алгоритма модуля изменения сокращенных схемных кодов

Модуль предназначен для изменения одного или нескольких сокращенных схемных кодов в массиве сокращенных схемных кодов.

Модуль вызывается управляющей программой пакета, которая указывает, что должно производиться: сокращение или расширение сокращенных схемных кодов.

Алгоритм данного модуля аналогичен алгоритму модуля изменения индивидуальных имен конститuent. В связи с этим возможно объединение этих двух модулей в один.

Блок-схема алгоритма не приводится.

4.4. Характеристика информационных массивов ППП ВИ
и связь с другими пакетами.

4.4.Г. Пакет внесения изменений использует часть массивов пакетов формирования ГРС и R - интерпретации, перевода представлений.

В пункте 4.4 для них описаны только особенности применения. Из пакета формирования ГРС использованы массивы: ОПСХ, РСД, ОТОТ, ТИИТ, КОПС. Из пакета R - интерпретации использованы массивы: ГК, *Rel*, ЭНАР, КРВГ. В связи с тем, что в ППП ВИ применены модули пакета R - интерпретации, то и все массивы, с которыми они работают, использованы в этом пакете. Из пакета перевода представлений использованы массивы ИНД, МСК.

В пакете ВИ введены следующие массивы:

- ИОПСХ - массив задания для изменения ОПСХ;
- ИБРС - массив задания для изменения базового рода структуры, дополнения;
- ИОТОТ - массив задания для изменения отображения ;
- РРЕЛ - массив задания для расширения множества *Rel* ;
- СОРЕЛ - массив задания для сокращения множества *Rel* ;
- ИРРТ - массив задания для расширения множества разрешающих термов;
- ИРСК - массив задания для расширения множества специнтерпретируемых конститuent;
- ИСРТ - массив задания для сокращения множества разрешающих тармов;
- ИСМОД - массив задания для изменения операционного заполнения ГР ;
- ИИНД - массив задания на изменение массива ИНД;
- ГАММА - массив задания на изменение имен в выражениях конститuent БРС или дополнения.
- ИМСК - массив задания на изменение массива МСК.

4.4.2. Массив ИОПСХ

Описание пунктов 1-6 совпадает с описанием соответствующих пунктов массива ОПСХ.

7. Получение массива ИОПСХ.

Массив получается в результате введения в БД задания по изменению операционной схемы.

8. Использование массива ИОПСХ.

Массив применяется как входной для модулей РОПСХ, СОПСХ, ЗОП.

9. Использование массива ИОПСХ в других пакетах.

Данный массив в других пакетах не используется.

4.4.3. Массив ОПСХ

Описание пунктов 1-6 см. 1.4.1.

7. Получение массива ОПСХ.

Первоначально массив ОПСХ получается из пакета ГРС. В результате работы модулей РОПСХ, СОПСХ, ЗОП массив обновляется.

8. Применение массива ОПСХ.

Массив является входным и выходным для перечисленных модулей.

Обновленный массив передается в пакет ГРС с кодом нового варианта.

9. Использование массива ОПСХ в других пакетах.

Массив используется в пакете формирования ГРС при выборе операции, а также в пакете перевода представления

4.4.4. Массив ИЕРС

Описание пунктов 1-6 совпадает с описанием соответствующих пунктов массива РСД см. 1.4.2

16.2.26

ТПТЗкн2

7. Получение массива ИЕРС.

Массив получается в результате ввода с внешних носителей задания на изменение базовых родов структур и дополнения.

8. Использование массива.

Массив используется как входной для модулей ИЕРС, ЗИК, ИОТН.

9. Применение массива в других пакетах.

Массив в других пакетах не используется.

4.4.5. Массив РСД.

Описание пунктов 1-6 см 1.4.2.

7. Получение массива

Первоначально массив РСД получается из пакетов формирования ГРС.

Массив обновляется в результате работы модулей УПД, ИЕРС, ЗИК, ИОТН.

8. Использование массива.

Массив используется как входовой и выходной при переходе к новому варианту.

9. Использование массива в других пакетах .

Массив используется пакетами:

- формирования ГРС;
- R₁ - интерпретация;
- перевода представления.

4.4.6. Массив ИОТН.

Описание пунктов 1-6 совпадает с описанием пунктов 1-6 массива ОТОТ.

7. Получение массива.

Массив получается в результате ввода с внешних носителей задания на изменение массива ОТОТ.

15-2-М
ТПЗем2

8. Использование массива.

Массив используется как входной для модуля ИОТН.

9. Использование в других пакетах.

В других пакетах не используется.

4.4.7. Массив ОТОТ.

Описание пунктов 1-6 см. 1.4.3.

7. Получение массива.

Первоначально массив получается из пакета формирования ГРС.

Массив обновляется модулем ИОТН.

8. Использование массива.

Массив является входным и выходным для модуля ИОТН.

9. Использование массива в других пакетах.

Массив используется модулями пакета формирования ГРС при выполнении операции.

4.4.8. Массив ТИНТ.

Описание пунктов 1-6 см. 1.4.4.

7. Описание массива.

Массив первоначально получается из пакета формирования ГРС. В дальнейшем он подвергается модификации УПП.

8. Использование массива.

Массив корректируется УПП при переходе к новому варианту.

9. Использование другими пакетами.

Массив ТИНТ используется в пакете перевода представления.

15-2-76

ТПТЗ м 2

Массив является выходным для пакета формирования ГРС.

4.4.9. Массив КОПС.

Описание пунктов 1-6 см. 1.4.5.

7. Применение массива.

Массив первоначально передается из пакета формирования ГРС и модифицируется УПЛ ВИ.

8. Использование массива КОПС.

Массив обновляется при переходе к новому варианту в режиме внесения изменений.

9. Использование массива в других пакетах.

Массив используется в пакете ГРС при выборе операции, которую можно выполнить.

4.4.10. Массив ГК.

Описание пунктов 1-6 см. 2.4.2.

7. Получение массива ГК.

Массив ГК первоначально получается из пакета R - интерпретации. Текущие ГК, ГР модифицируются модулями ССНЗ, Р *del* . *с del*, РАК, РСК, СРТ, РРТ, РТК, ИВГК.

8. Использование массива.

Массив ГК является входным и выходным для указанных модулей. Массив используется при изменении ГК, ГР перечисленными в п.7 модулями. Этот же массив используется для хранения ГК в ~~расширенном~~ виде.

9. Использование другими пакетами.

Массив ГК используется в пакете R - интерпретации, где первоначально получается для задания процесса R - интерпретации.

4.4.11. Массив Rel.

Описание пунктов 1-6 см. 2.4.3.

7. Получение массива.

Массив *Rel* первоначально получается из пакета *Rel* интерпретации и подвергается изменениям в результате работы модулями *СОНЗ*, *АНЗ*, *PRel*, *CRel*, *PAK*, *PSK*, *СРТ*, *РРТ*, *РГР*, *УВГР*.

8. Использование массива

Массив является входным и выходным для ПВЧ.

Массив используется при проверке задания на работу модулей, указанных в п.7. Массив обновляется модулями указанными в п.7.

9. Использование другими пакетами.

Используется пакетом *R* - интерпретации при получении информации о множестве вершин ГК и пакетом перевода представлений при формировании запроса по переводу представлений.

4.4.12. Массив PPEL A.

Описание пунктов 1-6 совпадает с описанием пунктов 1-6 массива РИПТ.

7. Получение массива.

Массив получается в результате ввода с внешних носителей задания на расширение множества *Rel*.

8. Использование массива.

Массив является входным для модуля *PRel*.

9. Использование в других пакетах.

В других пакетах не используется.

4.4.13. Массив COREL.

Описание пунктов 1-6 совпадает с описанием пунктов

I-6 массива РИИТ.

7. Получение массива СОРЕЛ.

Массив получается в результате ввода с внешних носителей задания на сокращение множества *Rel*.

8. Использование массива СОРЕЛ.

Массив является входным для модуля СОРЕЛ.

9. Использование в других пакетах.

В других пакетах не используется.

4.4.14 Массив ИРСК.

Описание пунктов I-6 совпадает с пунктами I-6 описания массива СИИТ.

7. Получение массива ИРСК.

Массив вводится с внешних носителей модулем РСК.

8. Использование массива ИРСК.

Массив используется как входной модулем РСК. Массив содержит задание для работы модуля.

9. Использование другими пакетами.

В других пакетах массив не используется.

4.4.15. Массив ИРРТ.

Описание пунктов I-6 совпадает с пунктами I-6 описания массива СИИТ.

7. Получение массива ИРРТ.

Массив вводится с внешних носителей модулем РРТ.

8. Использование массива ИРРТ

Массив используется как входной модулем РРТ. Массив содержит задание для работы модуля.

9. Использование другими пакетами.

В других пакетах массив не используется.

15-2-76

ТПТЗки2

4.4.16. Массив ИСРТ.

Описание пунктов 1-6 совпадает с пунктами 1-6 описания массива СМНТ.

7. Получение массива ИРРТ.

Массив вводится с внешних носителей модулем СРТ.

8. Использование массива ИСРТ.

Модуль используется как входной модулем СРТ. Массив содержит задание для работы модуля.

9. Использование другими пакетами.

В других пакетах массив не используется.

4.4.17. Массив ИСМОД.

Описание пунктов 1-6 совпадает сописанием пунктов 1-6 массива СМОД.

7. Получение массива ИСМОД.

Массив вводится с внешних носителей модулем ИОЗ.

8. Использование массива.

Массив используется как входной модулем ИОЗ. Массив содержит задание для работы модуля.

9. Использование другими пакетами.

В других пакетах массив не используется.

4.4.18. Массив ЗНАЗР.

Описание пунктов 1-6 см. 2.4.8.

7. Получение ЗНАЗР.

Массив получается из пакета R_1 - интерпретации.

8. Использование ЗНАЗР.

Массив подвергается изменениям УПП при переходе к новому варианту.

9. Использование другими пакетами.

Массив ЗНАЧР используется в пакете R_1 - интерпретации для хранения значений R_1 - интерпретированных конституэнт.

4.4.19. Массив КРВГ.

Описание пунктов 1-6 см. 2.4.9.

7. Получение КРВГ.

Массив получается из пакета R - интерпретации.

8. Использование КРВГ.

Массив используется для определения конститuent, получивших R - интерпретации. Массив обновляется УПП при переходе к новому варианту.

9. Использование другими пакетами.

Массив КРВГ используется в пакете R - интерпретации при работе модуля автоматической R - интерпретации.

4.4.20. Массив ИИИД.

Описание пунктов 1-6 совпадает с описанием пунктов 1-6 массива ИИД.

7. Получение массива ИИИД.

Массив вводится с внешних носителей модулем ИЗИИИМ.

8. Использование массива ИИИД.

Массив применяется как входной для модуля ИЗИИИМ. Массив содержит задание для работы модуля.

9. Использование массива в других пакетах.

В других пакетах массив не используется.

4.4.21. Массив ИМСК.

Описание пунктов 1-6 совпадает с описанием пунктов 1-6 массива ИМСК.

7. Получение массива ИМСК.

Массив вводится с внешних носителей модулем ИССК.

8. Использование массива ИМСК.

Массив применяется как входной модулем ИССК. Массив содержит задание для работы модуля.

9. Использование массива в других пакетах.
В других пакетах массив не используется.

4.4.22. Массив ИНД.

Описание пунктов 1-6 см. 3.4.1.

7. Получение массива ИНД.

Массив первоначально получается из пакета перевода представления. В пакете БИ он модифицируется модулем ИЗНИМ.

8. Использование массива ИНД.

Массив применяется как входной и выходной для модуля ИЗНИМ. Число записей массива изменяется указанным модулем.

9. Использование массива в других пакетах.

Массив применяется в пакете перевода представления при переводе конститuenty из одного представления в другое.

4.4.23. Массив ИСК

Описание пунктов 1-6 см. 3.4.2.

7. Получение массива ИСК.

Массив первоначально получается из пакета перевода представления. В пакете БИ он модифицируется модулем ИССК.

8. Использование массива ИСК.

Массив применяется как входной и выходной модуля ИССК.

9. Использование массива в других пакетах.

Массив применяется пакетом перевода представления при переводе конститuent из одного представления в другое.

4.4.24. Массив ГАММА.

1. Имя.

Выбор имени ограничивается требованиями содержать

идентификаторы проекта и варианты.

2. Информация, содержащаяся в записи массива.

В массиве содержатся записи 2-х типов. Каждая запись 1-го типа соответствует одной или нескольким последовательно расположенным конституэнтам, в выражениях которых производятся замены имен.

Каждая запись 2-го типа содержит имя базисного рода структуры или дополнения.

3. Организация массива.

Массив имеет записи 2-х типов. Запись 1-го типа - переменной длины, запись 2-го типа - фиксированной длины.

4. Метод доступа - последовательный или индексно-последовательный.

5. Максимальный размер 50x2x48 записей.

6. Структура записи.

Структура записи 1-го типа показана на рис. 4.3.1.

Признак записи 1-го типа	Имя 1-го конституэнта	Имя 2-го конституэнта	Замена именное	Замена именная	Замена именная	Замена именная
	:	:	:	:	:	:
	конституэнт	конституэнт	именное	именное	именное	именное
	:	:	:	:	:	:
	имя	имя	именное	именное	именное	именное

Рис. 4.3.1.

Структура записи 2-го типа показана на рис. 4.3.2.

Признак записи 2-го типа	:	Имя базисного рода структуры, дополнения
	:	имя

Рис. 4.3.2.

7. Получение массива ГАММА.

Массив вводится в БД после проверки правильности перфорации.

8. Использование массива ГАММА в ППП ВМ.

Массив используется как входной модулем ЗИК для замены имен конститuent.

9. Использование массива ГАММА в других пакетах.

Массив в других пакетах не используется.

4.5. Ограничения и возможности расширения ППП.

Пакет реализует в настоящее время внесение изменений в следующую входную информацию:

- операционную схему;
- базисные рода структур, дополнения;
- отождествляющее отображение η ;
- граф конституэнт, граф R -интерпретации;
- значение R -интерпретации;
- индивидуальные имена конституэнт;
- сокращенные схемные коды.

В пакете предусмотрена возможность расширения на разных уровнях:

- внесение изменений в новые пакеты;
- расширение возможностей внесения изменений в существующие пакеты с подключением новых модулей;
- расширение возможностей ЭИ, указанных в п.4.3. модулей тела пакета.

В частности, возможна реализация более сложных видов изменения, чем расширение, сокращение, замена. Эти расширения можно делать без программирования пакета.

Для обеспечения этих возможностей в пакете используются таблицы ТИЭИ, ТРИК и не описанная программа модификации этих таблиц.

Эта программа должна уметь добавлять новые строки, сдвигать строки, корректировать поля: наименования, адрес 1, адрес 2, описание ТИЭИ, ТРИК см. п.п.4.2., 4.3.).

В настоящее время можно задать ключ (см. п.4.3.), только указав его значение индивидуально (явно или неявно).

Более удобна система иерархично организованных ключей, когда задание одного ключа определяет значения ключей, подчиненных данному. Это позволит более широко использовать ключи в алгоритмах, что, в свою очередь, увеличит гибкость пакета.

В настоящее время при переходе к новому варианту, уделяется часть информации, которая изменится, а остальная часть копируется. Такой подход может привести к большому объему информации в системе. Для его уменьшения можно усложнить

логику системы и не копировать массивы, а использовать старые. При этом необходимо следить, чтобы не использовать ту часть старой информации, которая должна быть изменена.

4.6. Средства программного контроля и контроля результатов.

В пакете ВИ контроль распределен между управляющей программой, модулями тела пакета и БД.

Контроль УПП. УПП при получении задания проверяет, что:

- задание может быть выполнено модулями пакета;
- задание соответствует состоянию системы.

Контроль модулей тела пакета. При получении задания макромодуль проверяет, что модуль, реализующий заданное изменение, включен в макромодуль. Дальнейший контроль макромодуля совпадает с контролем модуля.

При сокращении и замене проверяется, что соответствующая входная информация определена, а при режиме расширения, что информация, определенная заданием, еще не включена в систему. Модуль проверяет задание на непротиворечивость. Например, при расширении операционной схемы проверяется, что ее расширение не приводит к циклам.

Контроль БД. Из средств контроля БД в пакете ВИ используются:

- средства контроля перфорации;
- средства контроля базисных родов структур, дополнения, которые используются после внесения изменения в базисные роды структур, дополнения.