

Рис. 3. Зависимость  $\sigma_T/\bar{T}$  (а) и  $\eta'$  (б) от  $\rho$  при  $n = 1,1$ :

1 —  $S/s = 5$ ; 2 —  $10^3$ . Отрезки на оси абсцисс, выделенные пунктиром, указывают диапазоны размеров, при которых величина  $LQ\eta$  не может быть равна единице

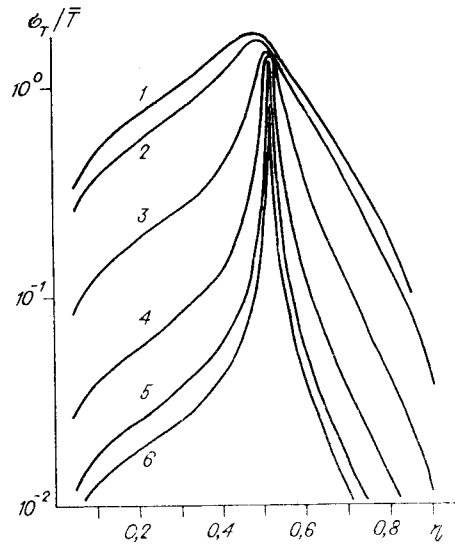


Рис. 4. Зависимость  $\sigma_T/\bar{T}$  от  $\eta$  при  $n = 1,3$ ;  $\rho = 7$  ( $Q = 3,85$ ;  $L = 0,5$ ):  
1 —  $S/s = 5$ ; 2 — 10; 3 —  $10^2$ ; 4 —  $10^3$ ; 5 —  $5 \cdot 10^3$ ; 6 —  $10^4$

ных монодисперсных сферических непоглощающих частиц. Они позволяют представить характер трансформации ОСШ в зависимости от апертуры считывания, оптических свойств и концентрации неоднородностей. Наличие поглощения, полидисперсности, неоднородности структуры частиц и других факторов приводит к усложнению расчетов параметров  $Q$  и  $L$ , входящих в формулу (1). Однако если свойства частиц таковы, что возможно выполнение условия  $QL\eta = 1$ , то характер описанных закономерностей сохраняется.

#### СПИСОК ЛИТЕРАТУРЫ

1. О'Нейл. Введение в статистическую оптику.— М.: Мир, 1966.
2. Иванов А. П., Лойко В. А. Оптика фотографического слоя.— Минск: Наука и техника, 1983.
3. Иванов А. П., Предко К. Г. Оптика люминесцентного экрана.— Минск: Наука и техника, 1984.
4. Danty I. C., Shaw R. Density and granularity of clustered distributions of monosized opaque dots // JOSA.— 1982.— 72, N 5.— P. 662.
5. Фризер Х. Воспроизведение фотографического изображения.— М.: Мир, 1978.
6. Вендровский К. В., Вейцман А. И. Фотографическая структурометрия.— М.: Искусство, 1982.
7. Domb C. Statistics of correlated events // Phil. Mag. (7).— 1950.— 41, N 321.— P. 969.
8. Коке Д., Смит В. Теория восстановления.— М.: Сов. радио, 1967.
9. Twersky V. Electromagnetic scattering. In multiple scattering of waves by periodic and by random distributions/Ed. P. L. E. Uslenghi.— N. Y.: Academ. Press, 1978.
10. Twersky V. Transparency of pair-correlated random distributions of small scatterers with applications to the cornea // JOSA.— 1975.— 65, N 5.— P. 524.
11. Дик В. П., Иванов А. П., Лойко В. А. Особенности ослабления излучения моно-съемом дискретных рассеивателей // Журн. прикл. спектроскопии.— 1987.— 47, № 3.

Поступило в редакцию 18 февраля 1988 г.

УДК 681.31.001.8

И. Е. ЛОБОВ, В. Е. МЕЖОВ, Ю. А. ЧЕВЫЧЕЛОВ  
(Воронеж)

#### ПОДГОТОВКА ИСХОДНЫХ ДАННЫХ В ПОДСИСТЕМЕ ФУНКЦИОНАЛЬНО-ЛОГИЧЕСКОГО МОДЕЛИРОВАНИЯ

Анализ процесса функционально-логического проектирования цифровых схем показывает, что этап подготовки исходных данных является узловым моментом, который во многом определяет качество проектирования. Удобство подготовки и

компактность представления данных, обеспечение максимально возможной информационной полноты и легкости изменения их структуры и содержания, близкие разработчикам,— вот основные требования, которым должна удовлетворять подсистема подготовки исходной информации.

**Модульный принцип описания схем.** При разработке схем большой размерности (до 100 000 транзисторов) все более важным становится использование модульной организации данных, которая лежит в основе иерархического метода проектирования.

Под модулем ниже понимается логически законченная часть информации, оформленная неким каноническим образом как единое целое (задано имя, определены входы (выходы) линий схемы, способы хранения, обращения, копирования, изменения, удаления).

Использование модульности в рамках рассматриваемой системы подготовки данных заключается в следующем: любая схема или ее часть оформляется как модуль. Простейшие модули-схемы описываются через типовые элементы системы функционально-логического моделирования ПРАЦИС-ТМ, называемые примитивами (примитив — математическая модель логического (булевого) вентиля, функционального матричного элемента и т. д.). Более сложные схемы могут включать в себя как примитивы, так и модули, ранее введенные в библиотеку. При описании модулей выделяются линии, являющиеся внешними входами и выходами модуля. После логической верификации программой моделирования структурная модель модуля помещается в библиотеку. В состав библиотеки могут включаться модули-схемы, соответствующие известным микросхемам, или любые оригинальные схемы. Это позволяет создавать библиотеку модулей, выполняющих практически любую необходимую функцию, логика работы которых не вызывает сомнения. При синтезе или анализе сложных схем используются такие, как суперпозиции известных функций, скоммутированных соответствующим образом из более простых модулей заменой формальных параметров фактическими именами линий схемы, являющихся входами и выходами модулей.

При моделировании схем, содержащих вложенные модули, пользователю предоставляется возможность контролировать любые линии схемы: внешние, соединяющие модули, а также внутренние линии вложенных модулей. Кроме того, при моделировании схем с неисправностями пользователь может потребовать не генерировать неисправности внутри специализированных модулей, например, являющихся микросхемами, что естественно при моделировании работы ТЭЗов.

**Организация библиотек логического описания схем.** Как уже отмечалось выше, основной структурной единицей системы является модуль. Это обеспечивает, во-первых, более четкое представление структуры разрабатываемых схем, во-вторых, резко уменьшает число потенциальных ошибок описания, так как, оформив отдельные фрагменты схемы в виде модулей, пользователь может сократить объем описания всей схемы в 5—10 раз. Наиболее эффективной формой хранения описания модулей является пользовательская библиотека модулей (библиотека проекта) — файл с прямым доступом. Структура такого файла выглядит следующим образом:

оглавление — список зарегистрированных в библиотеке модулей с указанием начала описания;

описание модуля — часть файла, где в определенном формате хранится систематизированная по установленным правилам информация.

Как правило, библиотека проекта содержит только те модули, которые задействованы для целевой схемы. Здесь необходимо сделать замечание о представлении содержания описания модуля, которое должно помочь разработчикам структуры и технологии понять и оценить задачи друг друга и те требования и ограничения, которые они налагают на систему. Необходимо так представлять содержание модуля, чтобы помочь разработчику описать поведение будущей системы и упростить ее функционально-логическое проектирование, а технологу (топологу, разработчикам кристалла, платы) оценить возможности ее физической реализации в выбранном элементном базисе. Например, при проектировании вентиляемых матриц часто используются функциональные примитивы — триггеры, регистры, счетчики. Моделирование на функциональном уровне осуществляется сравнительно быстро и, как правило, используется на ранних стадиях проектирования, однако более точные результаты (с учетом гонок фронтов, выбросов) могут быть получены на вентиляльном уровне, т. е. необходимо двойное представление модуля, имеющего одно и то же функциональное назначение. Замена одного содержания другим осуществляется в интерактивном режиме на соответствующем этапе разработчиком. Таким образом, на базе описания функциональных примитивов через простейшие вентили либо их комбинации формируется библиотека модулей-примитивов, которая используется для представления модели схемы в вентиляльном базисе. Например, модель схемы может представляться в примитивах базового матричного кристалла.

Подобные рассуждения приводят к выводу о целесообразности создания,ряду с библиотеками примитивов, общей системной библиотеки подсистемы функционально-логического проектирования, т. е. библиотеки, в которой содержится всесторонняя информация о базе проектирования: микросхемах, функциональных примитивах, вентилях.

В пакете программ ПРАЦИС-ТМ реализована следующая процедура формирования и использования системных библиотек. С помощью схемного редактора соз-

даются описания модулей: микросхем — файл MICRO. VEN, функциональных примитивов — FPRIM. VEN, примитивов на вентилях — GPRIM. VEN. Используя команды ОС MBC, создаются библиотеки: микросхем, модулей, примитивов, модулей функциональных примитивов.

Извлеченный из системной библиотеки модуль редактируется (в случае необходимости), транслируется и заносится в библиотеку проекта — MODUL. LIB. В процессе эксплуатации пакета ПРАЦИС-ТМ разработчиками происходит заполнение системных библиотек новыми модулями.

**Язык МОС (язык модульного описания схем).** В настоящее время существует достаточно много специализированных входных языков, представляющих описание функциональных и логических схем цифровой аппаратуры: МОЛК, МОДИС, ФОРА, АЛГОРИТМ И СТРУКТУРА, ЛЮТОС [1—3].

Язык МОС, рассматриваемый в данной работе, ориентирован на описание элементного состава схем, способа соединения элементов, перечня входных и выходных контактов. Входной язык МОС представляет собой расширенную версию языка описания схем системы ПРАЦИС-2 [4] с развитыми средствами иерархических ссылок на вложенные друг в друга модули.

Основными элементами языка МОС являются: командно-ключевые слова, определяющие последующую структуру описания; примитивы; модули; идентификаторы; атрибуты примитивов (задержки переключений и их силовые характеристики); служебные символы.

Прежде чем приступить к рассмотрению синтаксиса языка МОС, необходимо отметить два способа описания связей компонентов (элементов): покомпонентный и с явным заданием связей [2]. Первый способ более приемлем в силу его компактности, поэтому он удобен для описания устройств, состоящих из большого числа компонентов с большим числом связей. В силу изложенных причин в пакете программ ПРАЦИС-ТМ использован покомпонентный способ описания связей.

Приведем перечень команд языка МОС с кратким их описанием:

МО[ДУЛЬ] — задается имя текущего модуля;  
НА[ЗНАЧЕНИЕ] — задается текст, поясняющий назначение текущего модуля;  
ВХ[ОДЫ] — указываются внешние входы модуля;  
ВЫ[ХОДЫ] — указываются внешние выходы модуля;  
ДН — отмечается наличие внешних двунаправленных линий;  
МС — отмечается наличие неявных монтажных соединений;  
РА[ИГ] — задается групповое имя модулей, которые будут использоваться как микросхемы;  
ЗА[ДЕРЖКИ] — задаются задержки элементов;  
БИ[БЛИОТЕКА] — задаются мнемонические имена модулей, ранее введенные в библиотеку модулей, которые будут использоваться в описании настоящего модуля;  
ПЛ[М] — описываются программируемые логические матрицы, используемые в текущем модуле;  
КОМ[МУТАЦИЯ] — описываются коммутации элементов модуля;  
НО[ВТОРИТЬ] — используется для компактного описания коммутаций регулярных логических структур;  
КОН[ЕЦ] — признак конца текущего модуля.

*Примечание.* В квадратных скобках указаны необязательные символы.

Пять команд: МОДУЛЬ, ВХОДЫ, ВЫХОДЫ, КОММУТАЦИЯ, КОНЕЦ — являются обязательными для описания каждого модуля. Остальные команды используются по мере необходимости. Порядок следования команд в блоке описания модуля фиксирован и приведен выше.

Рассмотрим синтаксис команд языка МОС:

<МОДУЛЬ> ::= <ИМЯ МОДУЛЯ>  
<ИМЯ МОДУЛЯ> ::= <СИМВОЛ 1>, ..., <СИМВОЛ 12>

При описании элементов языка могут использоваться любые символы КОИ-7. Все имена, определяемые пользователем, могут содержать от 1 до 12 символов, не являющихся служебными.

<НАЗНАЧЕНИЕ> ::= <СИМВОЛ>, ..., <СИМВОЛ>  
<ВХОДЫ> ::= <СПИСОК ВХОДОВ>  
<СПИСОК ВХОДОВ> ::= <ИДЕНТИФИКАТОР>, ..., <ИДЕНТИФИКАТОР>  
<ИДЕНТИФИКАТОР> ::= <ИМЯ ЛИНИИ> ИЛИ <ИМЯ МНОГОРАЗРЯДНОЙ ШИНЫ>  
<ИМЯ МНОГОРАЗРЯДНОЙ ШИНЫ> ::= <ИМЯ ЛИНИИ> [<ИНДЕКС N>]  
<РАЗДЕЛИТЕЛЬ><ИНДЕКС K>:<ИМЯ  
ЛИНИИ> [<ИНДЕКС T>]  
<ИНДЕКС N> ::= <НАЧАЛЬНЫЙ НОМЕР ЛИНИИ ШИНЫ>  
<НАЧАЛЬНЫЙ НОМЕР ЛИНИИ ШИНЫ> ::= <ЦЕЛОЕ ЧИСЛО><ЦИФРА>

<ЦЕЛОЕ ЧИСЛО> ::= 1 : 2 : 3 : 4 : 5 : 6 : 7 : 8 : 9 : <ЦЕЛОЕ ЧИСЛО><ЦИФРА>  
 <ЦИФРА> ::= 1 : 2 : 3 : 4 : 5 : 6 : 7 : 8 : 9 :  
 <РАЗДЕЛИТЕЛЬ> ::= <,>  
 <ИНДЕКС К> ::= <КОНЕЧНЫЙ НОМЕР ЛИНИИ ШИНЫ>  
 <КОНЕЧНЫЙ НОМЕР ЛИНИИ ШИНЫ> ::= <ЦИФРА> [<ЦЕЛОЕ ЧИСЛО>]  
 <ИНДЕКС Т> ::= <НОМЕР ЛИНИИ ШИНЫ>  
 <НОМЕР ЛИНИИ ШИНЫ> ::= <ЦИФРА> [<ЦЕЛОЕ ЧИСЛО>]  
 <РАНГ> ::= <ИМЯ>  
 <ЗАДЕРЖКИ> ::= <ЗАДЕРЖКИ ПРИМИТИВ 1>, ..., <ЗАДЕРЖКИ ПРИМИТИВ N>  
 <ЗАДЕРЖКИ ПРИМИТИВА> ::= <ИМЯ ПРИМИТИВА><НОМЕР ТИПА>  
                                 [<СИЛОВЫЕ ХАРАКТЕРИСТИКИ>]  
                                 <СПИСОК ЗАДЕРЖЕК>  
 <ИМЯ ПРИМИТИВА> ::= <СИМВОЛИЧЕСКОЕ ИМЯ ПРИМИТИВА>  
 <НОМЕР ТИПА> ::= <ЦЕЛОЕ ЧИСЛО><ЦИФРА>  
 <СИЛОВЫЕ ХАРАКТЕРИСТИКИ> ::= <СИЛА 0><РАЗДЕЛИТЕЛЬ><СИЛА X>  
                                 <РАЗДЕЛИТЕЛЬ><СИЛА 1><РАЗДЕЛИТЕЛЬ>  
                                 <СИЛА Z>  
 <СИЛА><\*> ::= <S> : <D> : <R> : <Z>  
 <\*> ::= <0> : <X> : <1> : <Z>  
 <СПИСОК ЗАДЕРЖЕК> ::= <Т01><Т10>  
 <Т01> ::= <ЦЕЛОЕ ЧИСЛО><ЦЕЛОЕ ЧИСЛО>...<ЦЕЛОЕ ЧИСЛО>

Отметим, что <СИМВОЛИЧЕСКОЕ ИМЯ ПРИМИТИВА> является одним из 80 имен примитивов (модулей цифровых элементов), которые в своем составе содержат наиболее употребляемые имена элементов (И, ИЛИ ... МУЛЬТИПЛЕКСОРЫ ... ИЗУ ... МИКРОПРОЦЕССОРЫ). Используемые силовые характеристики, определяющие мощностные параметры сигналов, таковы: S — мощность источника для логической единицы либо шины заземления для логического нуля; D — мощность сигнала в открытом транзисторе; R — мощность сигнала на резисторе; Z — сигнал высокоомисапсного состояния. Что касается задержки элемента, то здесь рассматриваются временные характеристики элемента при его переключении с низкого уровня сигнала на высокий и наоборот:

<БИБЛИОТЕКА> ::= <ВЛОЖЕННЫЙ МОДУЛЬ 1>  
                                 <ВЛОЖЕННЫЙ МОДУЛЬ N>  
 <ВЛОЖЕННЫЙ МОДУЛЬ> ::= [<ЭКВИВАЛЕНТНОЕ ИМЯ М>] <ИМЯ МОДУЛЯ>  
 <ЭКВИВАЛЕНТНОЕ ИМЯ М> ::= <ИМЯ>  
 Здесь <ЭКВИВАЛЕНТНОЕ ИМЯ> выбирается пользователем для ссылки на модуль в разделе коммутация. Если указать только <ИМЯ МОДУЛЯ>, то оно совпадает с <ЭКВИВАЛЕНТНЫМ ИМЕНЕМ>.  
 <КОММУТАЦИЯ> ::= <ОПИСАТЕЛЬ 1>  
                                 <ОПИСАТЕЛЬ N>  
 <ОПИСАТЕЛЬ> ::= <ОПИСАТЕЛЬ ПРИМИТИВА> : <ОПИСАТЕЛЬ МОДУЛЯ> :  
                                 <ОПИСАТЕЛЬ СТРУКТУРЫ ПОВТОРИТЬ>  
 <ОПИСАТЕЛЬ ПРИМИТИВА> ::= <ИМЯ ПРИМИТИВА><НОМЕР ТИПА>  
                                 <ВЫХОДЫ><РАЗДЕЛИТЕЛЬ><ВХОДЫ>  
 <ВЫХОДЫ> ::= <СПИСОК ВЫХОДОВ><ЧИСЛО ВЫХОДОВ><СПИСОК ВЫХОДОВ>  
 <ЧИСЛО ВЫХОДОВ> ::= <ЦИФРА><ЦЕЛОЕ ЧИСЛО>  
 <РАЗДЕЛИТЕЛЬ> ::= <,>  
 <ВХОДЫ> ::= <ЧИСЛО ВХОДОВ><СПИСОК ВХОДОВ>  
 <ОПИСАТЕЛЬ МОДУЛЯ> ::= <ЭКВИВАЛЕНТНОЕ ИМЯ М> [<РАЗДЕЛИТЕЛЬ>  
                                 <ИМЯ ЭЛЕМЕНТА>] <КОММУТАЦИЯ ВЫВОДОВ>  
 <РАЗДЕЛИТЕЛЬ> ::= <,>  
 <КОММУТАЦИЯ ВЫВОДОВ> ::= <НЕЯВНОЕ ПОЗИЦИОНИРОВАНИЕ> ИЛИ  
                                 <ЯВНОЕ ПОЗИЦИОНИРОВАНИЕ>  
 <НЕЯВНОЕ ПОЗИЦИОНИРОВАНИЕ> ::= <СПИСОК ВЫХОДОВ М>  
                                 <РАЗДЕЛИТЕЛЬ> <СПИСОК ВХОДОВ N>  
 <СПИСОК ВЫХОДОВ М> ::= <ИДЕНТИФИКАТОР> 1, ..., <ИДЕНТИФИКАТОР>N  
 <РАЗДЕЛИТЕЛЬ> ::= <,>  
 <СПИСОК ВХОДОВ> ::= <ИДЕНТИФИКАТОР 1> ... <ИДЕНТИФИКАТОР N>

<ЯВНОЕ ПОЗИЦИОНИРОВАНИЕ> :: = <КОНТАКТ>... <КОНТАКТ>  
 <КОНТАКТ> :: = <ФОРМАЛЬНОЕ ИМЯ ВЫВОДА><РАЗДЕЛИТЕЛЬ>  
                   <ИДЕНТИФИКАТОР N>  
 <ФОРМАЛЬНОЕ ИМЯ ВЫВОДА> :: = <ИМЯ>  
 <РАЗДЕЛИТЕЛЬ> :: = <=>

Параметры <СПИСОК ВЫХОДОВ>, <СПИСОК ВХОДОВ> содержат списки имен выходов (входов) в том порядке, в котором они были определены в этом модуле ранее. Параметр <ФОРМАЛЬНОЕ ИМЯ ВЫВОДА> — имя, присвоенное по команде ВХОДЫ (ВЫХОДЫ) при описании вложенного модуля как внешнего.

Одной из особенностей языка МОС является возможность использования средств графического представления информации. Наличие графики позволяет провести графическое кодирование схемы и ее синтез, создать файлы описания и графическое изображение схемы. Команды для работы с графическими элементами позволяют проводить перекомпиановку элементов, их слияние и удаление. Сложный графический элемент (схема, система, узел) представляется в виде композиции более простых графических объектов из заданного набора. Разработаны средства создания графической библиотеки типовых элементов (логических и цифровых примитивов) и системная библиотека (микросхем и модулей), которые находятся в полном соответствии с ранее описанными аналогичными текстовыми библиотеками. Графическое представление данных — полный эквивалент текстового ввода.

Элементы подготовки графической информации:

<ПРОСТЫЕ ЭЛЕМЕНТЫ ГРАФИКИ> :: = <ТОЧКА> : <ОТРЕЗОК ПРЯМОЙ>  
 <ОТРЕЗОК ПРЯМОЙ> :: = <ТОЧКА><ТОЧКА>  
 <ТОЧКА> :: = <ТИП> : <КООРДИНАТЫ ПОЛОЖЕНИЯ ТОЧКИ ВИЗИРА ГД>  
 <ГРАФИЧЕСКИЕ МОДЕЛИ ЭЛЕМЕНТОВ И ФРАГМЕНТОВ> :: =  
 <ИМЯ ПОДПРОГРАММЫ, ГЕНЕРИРУЮЩЕЙ ИЗОБРАЖЕНИЕ ЭЛЕМЕНТА ИЛИ  
 ФРАГМЕНТА ГПП>  
 <ГПП> :: = <БУКВА><ЦИФРА>  
 <ОРИЕНТАЦИЯ> :: = <ЦИФРА>

Здесь же определены логические элементы и элементарные компоненты схемы (резистор, емкость, транзистор и т. п.):

<ЭЛЕМЕНТ СХЕМЫ> :: = <ЛОГИЧЕСКИЙ ЭЛЕМЕНТ> : <ЭЛЕМЕНТАРНЫЙ  
                   КОМПОНЕНТ> : <МОДУЛЬ>  
 <ЛОГИЧЕСКИЙ ЭЛЕМЕНТ> :: = <ОДИН ИЗ ПРИМИТИВОВ>  
 <ЭЛЕМЕНТАРНЫЙ КОМПОНЕНТ> :: = <ЕМКОСТЬ> : <РЕЗИСТОР> : <ДИОД> :  
                   <ТРАНЗИСТОР>  
 <МОДУЛЬ> :: = <ИМЯ МОДУЛЯ><АТТРИБУТЫ МОДУЛЯ>  
 <АТТРИБУТЫ МОДУЛЯ> :: = <ВЫХОД><РАЗДЕЛИТЕЛЬ><ВХОДЫ>

При применении графического ввода для синтеза функционально-логической схемы предварительно формируется графическая библиотека элементов или фрагментов. С помощью простых графических построений трассируются необходимые связи между элементами схемы с указанием точек излома и узлов трассы, соответствующих связям. Трансляция обеспечивает формирование массивов коммутации для последующего их использования.

Команды и программы контроля, редактирования и сопровождения языка МОС позволяют осуществлять работу с текстовой и графической информацией, обеспечивают вывод информации, интерфейс к программам подсистемы функционально-логического проектирования.

В заключение необходимо отметить, что рассмотренные выше средства и методы описания схем реализованы в версии ОС МОС «Электроника 82», позволяют просто и наглядно создавать математические модели сложных цифровых устройств, обеспечивая при этом функционирование подсистем моделирования, топологического проектирования, программ контроля режима и подготовки тест-программ. Кроме того, разработано программное обеспечение, выполняющее автоматическую трансляцию описания схемы в графическую форму для последующего документирования. Настоящая версия языка МОС успешно эксплуатируется в процессе проектирования вентиляльных матриц и ТЭЗов при разработке ЭВМ.

### СПИСОК ЛИТЕРАТУРЫ

1. Ландау И. Я. Применение ЦВМ для проектирования ЦВМ.— М.: Энергия, 1974.
2. Автоматизированное проектирование цифровых устройств/Под ред. С. С. Бадулина.— М.: Радио и связь, 1981.
3. Глушков В. М., Капитонова Ю. В., Летачевский А. А. Автоматизация проектирования вычислительных машин.— Киев: Наук. думка, 1975.