

**ВЫЧИСЛИТЕЛЬНАЯ  
ТЕХНИКА  
СОЦИАЛИСТИЧЕСКИХ  
СТРАН**

Межправительственная комиссия  
по сотрудничеству социалистических стран  
в области вычислительной техники



# ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА СОЦИАЛИСТИЧЕСКИХ СТРАН

Сборник статей

Издается с 1977 г.

*Выпуск 15*

Под общей редакцией М. Е. Раковского

**OldPC.ru**

**5050**

**музей компьютеров**



МОСКВА  
«ФИНАНСЫ И СТАТИСТИКА»  
1984

УДК 681.3  
ББК 32.97  
В94

*Редакционная коллегия сборника: М. Е. Раковский — главный редактор (Координационный центр МПК), Л. Варга (ВНР), Н. В. Горшков (Совет по комплексному обслуживанию СВТ), И. Корж (ЧССР), Е. Н. Мельникова — ответственный секретарь (Координационный центр МПК), Б. Н. Наумов (Совет главных конструкторов СМ ЭВМ), С. Пашев (НРБ), С. Пашковский (ПНР), В. В. Пржиялковский (Совет главных конструкторов ЕС ЭВМ), Ю. П. Селиванов — ответственный редактор (СССР), А. Е. Фатеев (СССР), Х. Чоппе (ГДР).*

Тематика сборника — проблемы разработки современной микроэлектронной элементной базы для ЕС ЭВМ и СМ ЭВМ. Рассматривается деятельность Совета по этим вопросам, основные проблемы и перспективы развития средств микроэлектроники для ЭВМ. Освещаются вопросы использования микропроцессорной техники для АСУ различного уровня.

Для специалистов по разработке, применению и эксплуатации средств ЕС ЭВМ и СМ ЭВМ, для руководителей и организаторов работ.

В 240500000—054 93—84  
010(01)—84

# I

## Международное сотрудничество социалистических стран в области вычислительной техники

---

УДК 681.325.65.181.5

### МИКРОЭЛЕКТРОННАЯ ЭЛЕМЕНТАРНАЯ БАЗА ДЛЯ СРЕДСТВ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

**Ю. В. Терехов**, инженер (Координационный центр МПК по ВТ)

На развитие вычислительной техники особенно большое влияние оказало появление интегральных схем (ИС), созданных в результате бурного развития микроэлектроники.

Полупроводниковые логические ИС широко применяются при создании ЭВМ и средств вычислительной техники (СВТ) различного назначения. Это объясняется тем, что в сравнении с логическими элементами на дискретных электронных компонентах полупроводников ИС дешевле, обладают более высокой надежностью, имеют малые массу и габариты. Высокие технико-экономические характеристики полупроводниковых ИС достигаются за счет групповой технологии их изготовления, небольшого удельного расхода материалов, проведения всех технологических операций в условиях высокой чистоты, жесткого контроля окружающей среды и хода технологического процесса.

За 20-летний период развития микроэлектроники степень интеграции ИС была увеличена с нескольких десятков до сотен тысяч логических элементов на кристалл, рабочие частоты возросли в сотни раз, значительно повысилась надежность ИС.

В результате совершенствования и развития технологии и схемотехники ИС создана функционально полная система изделий микроэлектронной элементной базы (МЭБ) для средств вычислительной техники различного назначения (логические ИС, микропроцессоры, запоминающие устройства, аналоговые ИС и т. п.).

Учитывая важность развития микроэлектронной элементной базы для СВТ, в Межправительственной комиссии по сотрудничеству социалистических стран в области вычислительной техники в 1981 г. создан новый орган — Совет по МЭБ. Основные направления деятельности Совета по МЭБ определены соответствующими соглашениями между странами — членами СЭВ.

Главной задачей Совета по МЭБ является организация многостороннего сотрудничества в области разработки и производства перспективных микроэлектронных изделий, необходимых для создания технических средств ЕС и СМ ЭВМ на уровне мировых стандартов в области вычислительной техники.

В странах — участницах Соглашения в области разработки, производства и применения вычислительной техники создана большая номенклатура интегральных схем различного назначения. С целью оптимизации номенклатуры ИС, упрощения эксплуатации и ремонта технических средств, создаваемых на основе ИС, секцией специалистов по микроэлектронной элементной базе Совета по МЭБ разработан документ «Микросхемы интегральные. Единый перечень ИС, рекомендуемых для применения в СВТ. Редакция 6-82».

Единый перечень представляет собой функционально полную систему ИС и больших интегральных схем (БИС) для вычислительной техники и включает следующие основные типы схем: логические ИС; БИС микропроцессоров; БИС запоминающих устройств; БИС универсальных вентиляльных матриц; интерфейсные ИС для связи ЭВМ с объектами управления, передачи и сбора информации; БИС цифроаналоговых и аналого-цифровых преобразователей; ИС для вторичных источников питания.

Основной тенденцией в развитии изделия микроэлектроники являются увеличение функциональной сложности за счет роста степени интеграции и повышение быстродействия ИС. Однако создать ИС с максимальной степенью интеграции и наивысшим быстродействием, используя только какой-либо один вид схемотехники и технологии, практически невозможно, так как в зависимости от условий применения СВТ одновременно могут возникнуть требования малой потребляемой мощности, работоспособности в широком диапазоне температур и т. п. Такое многообразие требований к ИС выполняется за счет использования различных технологических и схемотехнических решений при их изготовлении.

В производстве цифровых ИС наиболее широкое распространение получили ТТЛ, КМОП, МОП и ЭСЛ схемы.

ЭСЛ ИС обладают наиболее высоким быстродействием (время задержки 1—2 нс/вентиль\*), но при этом потребляют значительную энергию от источника питания (20 мВт/вентиль), что ограничивает максимально достижимую степень интеграции этого типа ИС до 300—350 вентиль/кристалл.

ТТЛ ИС имеют меньшее быстродействие, чем ЭСЛ ИС (время задержки — 3—10 нс/вентиль, энергия, потребляемая от источников питания, — 2—10 мВт/вентиль, степень интеграции — до нескольких тысяч вентилю на кристалл).

Для этого типа схем характерно сохранение работоспособности в широком интервале рабочих температур от —60 до +125°C. Технологи и схемотехника МОП позволяют достигать наиболее высокой степени интеграции — до нескольких сотен тысяч вентилю на

---

\* Вентиль — простейший двухходовой логический элемент.

кристалл. С другой стороны, этот тип ИС уступает ТТЛ по быстродействию, нагрузочным способностям, диапазону рабочих температур (от  $-10$  до  $+70^{\circ}\text{C}$ ). КМОП-схемы обеспечивают самое минимальное потребление энергии из всех рассмотренных типов ИС, имеют быстродействие на уровне МОП-схем, но более сложны в изготовлении и уступают им по достигаемой степени интеграции.

**Структура «Единого перечня ИС, рекомендуемых для применения в СВТ. Редакция 6-82»**

| Номер функциональной группы | Функциональное назначение                          | Количество типов ИС, шт. | Основные технические характеристики   |
|-----------------------------|--|--------------------------|---|
| 1                           | Логические ИС                                      | 295                      | Время задержки, нс/вентиль:<br>КМОП — 30<br>ТТЛ — 10<br>ЭСЛ — 2                                     |
| 2                           | Запоминающие устройства                            | 64                       | Информационная емкость, бит/кристалл:<br>ОЗУ      ПЗУ<br>1 024    1 024<br>4 096    4 096<br>16 384 |
| 3                           | Микропроцессорные БИС                              | 72                       | Разрядность — $2 \times n$ , 8 и 16<br>Быстродействие — $100 \div \div 2000$ тыс. оп./с             |
| 4                           | Аналоговые ИС                                      | 74                       | Операционные усилители различного назначения  |
| 5                           | Цифроаналоговые и аналого-цифровые преобразователи | 7                        | Разрядность — 8, 10 и 12  |
| 6                           | Интерфейсные ИС                                    | 37                       | Магистральные усилители, приемники, формирователи   |

В Единый перечень включены серии ИС и БИС, выполненные по всем основным видам технологии. Состав и основные параметры изделий МЭБ, включенных в Единый перечень, приведены в таблице.

Особое внимание в странах — участницах Соглашения уделяется развитию номенклатуры БИС микропроцессоров (БИС МП), обес-

печивающих создание дешевых массовых средств вычислительной техники.

В номенклатуру МЭБ включены следующие основные комплекты БИС МП:

8-разрядные  $n$ -МОП БИС МП с фиксированной и наращиваемой разрядностью, предназначенные для создания контроллеров, 8- и 16-разрядных микроЭВМ;

биполярные ТТЛ ДШ и ЭСЛ БИС с наращиваемой разрядностью, кратной  $2 \times n$ ,  $4 \times n$ ,  $8 \times n$ , предназначенные для создания на их основе быстродействующих контроллеров, микроЭВМ и мульти-микропроцессорных систем;

16-разрядные  $n$ -МОП БИС МП и однокристалльная 16-разрядная микроЭВМ, предназначенные для создания одноплатных микроЭВМ, в системах управления технологическими процессами, контрольно-измерительной аппаратуре и других системах обработки цифровой информации.

Номенклатура полупроводниковых БИС ЗУ охватывает основные разновидности запоминающих устройств современных ЭВМ, в их числе:

сверхбыстродействующие регистровые ЗУ, выполненные по биполярной ТТЛ и ЭСЛ технологиям с информационной емкостью 16—64 бита/кристалл и временем выборки 10—30 нс;

БИС для оперативных ЗУ ЭВМ — статические с информационной емкостью 1 и 4 Кбит/кристалл и динамические емкостью 4, 16 и 64 Кбит/кристалл;

БИС ПЗУ со стандартными прошивками информационной емкостью 256—8192 бита/кристалл, применяемые в качестве генераторов символов;

быстродействующие ТТЛ БИС ПЗУ, программируемые потребителем пережогом плавких перемычек с информационной емкостью 1 и 4 Кбит/кристалл и временем выборки менее 100 нс;

МДП БИС ПЗУ с возможностью многократной перезаписи информационной емкостью 4, 8 и 16 Кбит/кристалл.

Серия стандартных цифровых ИС, БИС МП и БИС ЗУ дополнена несколькими типами базовых матричных кристаллов (БМК).

БМК — это новое направление в создании цифровых специализированных ИС. Известно, что специально спроектированные ИС для конкретного узла ЭВМ обладают наилучшими параметрами в отношении быстродействия и не имеют избыточности. Но такой подход к созданию ИС зачастую оказывается неприемлемым из-за их малой применяемости, а следовательно, и высокой стоимости. БМК представляет собой базовый кристалл, в котором сформирована матрица стандартных логических элементов (вентили, триггеры, усилители и т. п.), их соединение определяется шаблонами металлизации.

При создании конкретных типов ИС на основе БМК меняется только схема соединения элементов, поэтому даже при небольшой тиражности ИС материальные затраты на ее изготовление невелики.

Для передачи сигналов управления от ЭВМ к объектам управления, к исполнительным механизмам и устройствам сбора инфор-

мации от объектов управления разработан и включен в Единый перечень ряд ИС;

операционные усилители средней и высокой точности;

4-, 8- и 16-канальные ключи и коммутаторы токов и напряжений;

быстродействующие 8-, 10- и 12-разрядные БИС ЦАП и АЦП.

В целях уменьшения габаритов и увеличения КПД источников питания создана номенклатура ИС для вторичных источников (напряжение  $\pm 5$ ,  $\pm 12$ ,  $\pm 15$  В и токи нагрузок до 1,5 А).

В соответствии с планами работ и концепцией развития перспективных моделей ЕС и СМ ЭВМ важнейшими направлениями развития изделий МЭБ являются:

увеличение степени интеграции, быстродействия и надежности;

создание 16- и 32-разрядных микропроцессорных комплектов, совместимых по системе команд СМ ЭВМ. В целях повышения эффективности применения разработанных комплектов БИС МП планируется их дополнение развитой номенклатурой интерфейсных БИС и БИС для устройств связи с объектами;

увеличение информационной емкости БИС ЗУ до 256 Кбит/кристалл и создание БИС с организацией на 4, 8 и 16 разрядов, увеличение быстродействия и снижение потребляемой мощности;

дальнейшее развитие номенклатуры базовых матричных кристаллов на основе ТТЛ, *n*-МОП, КМОП и ЭСЛ технологий, увеличения их быстродействия, повышения функциональной сложности и увеличения количества выводов. Широкое применение базовых матричных кристаллов требует создания высокопроизводительных систем автоматизированного проектирования БИС на основе БМК, тесного взаимодействия между заказчиками и изготовителями БИС.

Даже простое перечисление тенденций развития МЭБ позволяет сделать вывод о том, что основной элементной базой перспективных СВТ будут служить большие и сверхбольшие интегральные схемы (СБИС), содержащие десятки и сотни тысяч вентилях на кристалле.

При создании и массовом применении БИС и СБИС необходимо учитывать их особенности:

разработка каждого типа БИС, а тем более СБИС — трудоемкий и сложный процесс, требующий участия высококвалифицированных специалистов различного профиля (математиков, системотехников, схемотехников, конструкторов, технологов и т. п.). Это обуславливает необходимость жесткой унификации, единых технических и технологических решений;

для создания и производства БИС необходимо дорогостоящее прецизионное автоматизированное технологическое исследовательское и контрольно-измерительное оборудование и широкая номенклатура особо чистых материалов и химических реактивов, обеспечивающих устойчивое получение субмикронных размеров. Эти проблемы могут быть решены только путем быстрого кооперирования стран СЭВ в области создания, производства и обеспечения взаимных поставок специального технологического оборудования и особо чистых материалов для производства изделий МЭБ;



особое внимание следует уделять повышению надежности БИС и СБИС, в том числе за счет применения в БИС методов резервирования, использования корректирующих кодов и т. п. Необходимо совершенствовать методы прогнозирования эксплуатационной надежности разрабатываемых изделий МЭБ в зависимости от воздействия факторов окружающей среды, а также методы сбора и анализа информации об отказах изделий МЭБ.

Указанные особенности разработки и применения БИС и СБИС требуют дальнейшего углубления сотрудничества стран, ускоренного решения вопросов стандартизации и унификации номенклатуры БИС.

УДК 681.3.06

### **ОРГАНИЗАЦИЯ РАБОТ ПО РАЗВИТИЮ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**М. Е. Раковский**, канд. техн. наук (Координационный центр МПК по ВТ)

**Т. В. Жукова**, инженер (Координационный центр МПК по ВТ)

Сегодня особенно ярко вырисовывается необходимость консолидации сил всех программистов социалистических стран для коренного изменения положения с разработкой программного обеспечения. Можно без преувеличения утверждать, что эффективность использования вычислительной техники в странах социалистического лагеря в настоящее время зависит от обеспеченности программными средствами для различных областей применения и качества этих средств.

Несмотря на большое количество разработанных пакетов прикладных программ (ППП), их все же недостаточно для удовлетворения потребностей того широкого круга пользователей, которые применяют сейчас вычислительную технику. В силу этого значительное количество пользователей вынуждено заниматься самостоятельной разработкой программных средств, что сопровождается значительными затратами, но не дает нужного эффекта.

Стоимость программного обеспечения продолжает неуклонно расти [1].

Учитывая, что жизненный цикл технических средств вычислительной техники на данном этапе значительно короче жизненного цикла программного обеспечения, задача социалистических стран, объединяемых Межправительственной комиссией по вычислительной технике (МПК по ВТ), состоит в разработке глубоко продуманной философии производства прикладного программного обеспечения, которая бы гарантировала социалистические страны от

непроизводительных затрат в этой области и служила скорейшему обеспечению потребителя необходимыми ППП. Не следует забывать, что за прошедшие годы в странах накоплен большой объем программного обеспечения, однако полной и глубокой ревизии этих программ не сделано.

По имеющимся в Координационном центре МПК по ВТ данным в социалистических странах — участницах Соглашения имеется более 12 тыс. программных средств для различных сфер применения, в том числе около 1 тыс. — в НРБ, примерно 1,3 тыс. — в ВНР, около 4,5 тыс. — в ГДР, более 6 тыс. — в СССР. Из этого количества 238 пакетов прикладных программ, представляющих наиболее современные программные средства, разработаны органами МПК по ВТ и прошли совместные (международные) испытания на 1 января 1983 г. 127 ППП включены в Совместный фонд прикладных программ (СФПП).

Прикладное программное обеспечение можно разбить на ряд основных групп:

- программные средства общего назначения (включают системы управления базами данных (СУБД), информационно-поисковые и информационно-справочные системы (ИПС и ИСС), программные средства подготовки и ввода данных);

- программные средства технологии программирования;

- программные средства для расширения возможностей операционных систем;

- программные средства методоориентированных расчетов (методы оптимизации, статистические методы, методы сетевого планирования и управления, математические методы, методы моделирования);

- программные средства функционального назначения для АСУ в промышленности (техническая подготовка производства, управление основным производством, бухгалтерский учет и управление финансами, управление кадрами и др.);

- программные средства для применения в непромышленных отраслях;

- программные средства систем автоматизации проектирования (САПР) в строительстве, энергетике, машиностроении и других областях.

К началу 1983 г. среднее количество повторных применений ППП, разработанных в рамках МПК по ВТ, составило около 50, а общее число применений — около 7 тыс. При этом свыше 200 повторных применений имеют более 5% ППП. Данные о применяемости различных групп ППП приведены в таблице.

Наивысшую применяемость имеют программные средства общего назначения и пакеты методоориентированных расчетов, так как они могут использоваться в самых разных областях народного хозяйства. Широко применяются программные средства АСУ в промышленности.

Чтобы удовлетворить большее количество пользователей ВТ в программных средствах, необходимо интенсивно внедрять пакеты

прикладных программ общего назначения и ППП, реализующие разнообразные вычислительные методы. Необходимо также повысить уровень типизации функциональных программных средств. Раз-

| Группа программных средств (ПС)                           | Количество ППП, для которых учтено число применений | Количество применений | Среднее число повторных применений |
|---|---|-----------------------|------------------------------------|
| ПС общего назначения                                      | 22  | 1 904                 | 86                                 |
| ПС технологии программирования                            | 1   | 14                    | 14                                 |
| ПС для расширения возможностей операционных систем        | 9   | 54                    | 6                                  |
| ПС методоориентированных расчетов                         | 36  | 2 402                 | 66                                 |
| ПС функционального назначения для АСУ в промышленности    | 42  | 2 083                 | 49                                 |
| ПС функционального назначения для непромышленных отраслей | 16  | 103                   | 6                                  |
| ПС систем автоматизации проектирования (САПР)             | 12  | 41                    | 3                                  |
| Всего   | 138   | 6 601                 | 48                                 |

работку этих групп ППП следует распределить среди специализированных коллективов программистов. При этом особое значение приобретает задача максимальной стандартизации структур данных, особенно их машинное представление в различных типовых применениях вычислительной техники.

Структура национальных фондов прикладных программ и совместных разработок в рамках МПК по ВТ не имеет резких отличий. Сейчас основную долю (25—55%) составляют ППП функционального назначения для автоматизированных систем управления в промышленности и непромышленных отраслях народного хозяйства, более 10% — ППП методоориентированных расчетов, около 10% — программные средства общего назначения для АСУ и в среднем 3,5% — программные средства технологии программирования. Анализ фондов показал, что во всех странах ведутся разработки аналогичных программных средств. По имеющимся данным более 70% работ по созданию программных средств функционально дублируют друг друга [2]. Очевидно, что это ведет к увеличению и без того больших затрат на программное обеспечение.

Чтобы избежать дублирования работ как внутри стран, так и в рамках сотрудничества, надо улучшить информирование разработчиков и пользователей программных средств о готовых и ведущихся разработках, повысить качество разрабатываемых программных средств, улучшить их сопровождаемость. В рамках МПК по ВТ необходимо совершенствовать систему планирования разработки программного обеспечения, в том числе проработать вопрос о специализации стран на разработке определенных групп программных

средств. Экономическому совету совместно с соответствующими заинтересованными органами необходимо начать разработку положения и проекта соглашения о специализации.

Планирование создания программных средств надо осуществлять с учетом заинтересованности стран в покупке программных продуктов, развивать кооперирование при разработке программных средств. Необходимо изучить и распространить опыт создания крупных дву- и многосторонних международных организаций по типу Болгаро-советского научно-исследовательского и проектного института «Интерпрограмма».

Решение проблем создания программных средств невозможно без обеспечения разработчиков современными версиями операционных систем, поскольку они несут в себе ряд важных элементов технологии программирования, например средства тестирования, отладки и т. д. Однако поставка и сопровождение операционных систем не налажены должным образом.

Комплексное применение вычислительной техники в 70-е годы привело к тому, что затраты на программное обеспечение по отношению к затратам на технические средства значительно возросли и имеют тенденцию к дальнейшему увеличению. Повышение удельного веса программного обеспечения в стоимости вычислительных систем обуславливается, во-первых, резким увеличением объема программ, требуемых для современных систем, и, во-вторых, низкими темпами роста производительности труда программистов.

Производительность труда программистов отличается при разной их квалификации и разных условиях работы. Количество специалистов, занятых производством прикладных программ в социалистических странах, необоснованно велико и по экспертной оценке составляет 150 тыс. человек. При этом основной объем программных средств (около 80%) во всех странах создается пользователями вычислительной техники, работа которых, как правило, организована слабо, и производительность труда остается низкой. Производители ЭВМ и специализированные организации разрабатывают и поставляют программные средства ограниченной номенклатуры в больших количествах, и они имеют, как правило, крупные коллективы высококвалифицированных программистов. Расходы на создание программных средств у них значительно ниже, чем у пользователей. Поэтому необходимо концентрировать силы программистов в крупных специализированных организациях, обеспечивая их самыми современными техническими средствами и технологиями.

Разработка программных средств традиционными методами не обеспечивает необходимого качества программного продукта, вследствие чего затраты на его сопровождение значительно превышают стоимость разработки. Начинается изменение процесса программирования: создание программных средств из искусства превращается в отрасль промышленного производства. Перевод на промышленную основу процессов разработки и сопровождения программного обеспечения является, несмотря на специфику про-

цесса программирования, реальной и своевременной задачей. Ее решение включает в комплексе организационные, методологические, структурные, технологические задачи и задачи создания аппаратных средств.

Остановимся на одной из проблем — создании промышленной технологии разработки программных средств. Понятие технологии программирования (ТП) охватывает все содержание процесса создания программных средств от появления потребности в программном средстве до полного прекращения его использования, т. е. на всех стадиях жизненного цикла ПС. Сюда относятся анализ и проектирование ПС, его реализация (кодирование, отладка, тестирование, документирование) и использование (эксплуатация, сопровождение, модификация). Уровень ТП определяет некоторую профессиональную культуру деятельности специалистов и имеет своей целью обеспечение заданного качества ПС и заданного уровня производительности труда специалистов. Технология программирования должна соответствовать следующим требованиям:

- повышение производительности каждого программиста на всех стадиях разработки программного продукта;

- организация планирования и управления работой коллектива программистов, повышение эффективности их взаимодействия, создание возможности глубокого распараллеливания разработки при работе над большими и сверхбольшими программными системами;

- создание понимаемой пользователем документации и упрощение эксплуатации ПС;

- упрощение процессов верификации и отладки программ, выявление и устранение ошибок на возможно более ранних стадиях разработки;

- упрощение процесса внесения изменений и поправок;

- упрощение процесса переноса созданных программных средств на новую техническую базу.

Вопросами технологии программирования в рамках МПК по ВТ занимается Совет по применению средств вычислительной техники. Цель этой работы — объединить усилия социалистических стран для ускорения решения проблем развития и внедрения промышленной технологии создания программных средств.

Основные направления деятельности в этой области:

- формирование научно-теоретического базиса технологии создания программных систем;

- создание технологических комплексов, включающих методологические и инструментальные средства, обеспечивающие процесс разработки программных продуктов для различных областей применения;

- разработка методов организации и управления производством программных средств с использованием ЭВМ.

Основным результатом сотрудничества по технологии программирования явилось достижение единства взглядов стран — участниц Соглашения по ВТ; был разработан и принят документ «Общая концепция технологии программирования», а также разработаны

и согласованы «Обобщенная модель цикла жизни программных продуктов», «Словарь терминов технологии», «Классификатор проблем технологии», дополнения к методике испытаний ППП с учетом особенностей технологических средств и материал «Учет, передача и использование технологии программирования».

Разработан ряд технологических средств: «Технологический комплекс программиста (ТКП-1)» (СССР, испытан в 1981 г.), ППП «Спектр», «Контролер», «Библиотекарь» (СССР, испытаны в 1982 г.), «Технологические средства программирования в интерактивном режиме (АНСВЕР)», «Внедрение методологии Джексона» и «Инструментальные средства в рамках технологии АРТ (МОЗ — АРТ)», (ВНР, испытаны в 1983 г.). К концу пятилетки страны — участницы Соглашения уже будут иметь технологические средства, разработанные на основе общей концепции технологии программирования.

На следующем этапе ставится проблема перевода на контрактную основу наиболее актуальных разработок и обмена технологическими разработками, обеспечение необходимыми нормативными и методическими материалами.

Проблемы программного обеспечения требуют безотлагательно-го решения. Необходимо осуществлять ряд мероприятий для совершенствования разработки и применения программных средств: улучшить систему информации пользователей о разработанном и разрабатываемом программном обеспечении как внутри стран, так и между ними;

осуществлять централизованное финансирование разработки ряда программных средств для нужд всех стран — участниц Соглашения;

продолжить развитие технологии программирования с целью создания единых технологических средств в рамках МПК по ВТ. Эти работы необходимо вести в тесной связи с фундаментальными исследованиями в области программирования;

создать в странах — участницах Соглашения условия, стимулирующие развитие сопровождения программного обеспечения как НОТО стран, так и разработчиками, а также разработать комплекс необходимых нормативных документов;

разработать организационные мероприятия, направленные на совершенствование поставки и сопровождения операционных систем и программных средств, расширяющих возможности операционных систем.

#### ЛИТЕРАТУРА

1. Громов Г. Р. **Промышленность обработки данных.** — Зарубежная радиоэлектроника, 1982, № 10, с. 26.
2. Глушков В. М. **Фундаментальные исследования и технология программирования.** — Программирование, 1980, № 2, с. 5.

## II

# Технические средства вычислительной техники

---

УДК 681.3:621.3.049.77

### ОСНОВНЫЕ НАПРАВЛЕНИЯ РАЗВИТИЯ ВЫСОКОПРОИЗВОДИТЕЛЬНЫХ ЭВМ ОБЩЕГО НАЗНАЧЕНИЯ НА ОСНОВЕ БИС

Ю. С. Ломов, канд. техн. наук (СССР)

Повышение производительности вычислительных систем — одна из важнейших задач развития вычислительной техники. Однако само понятие «производительность» меняется в зависимости от изменения функций и назначения ЭВМ. Если раньше шла речь о производительности ЭВМ при решении научно-технических задач, то для современных ЭВМ общего назначения требуется учитывать также производительность при решении задач информационно-логического и экономического характера, т. е. появилась необходимость обеспечивать равную или примерно равную производительность для широкого класса задач. Универсализация, безусловно, накладывает ограничения на достижение максимальной производительности ЭВМ. Эти ограничения могут быть преодолены путем создания спецпроцессоров для наиболее распространенных классов научно-технических задач и подключения их к ЭВМ общего назначения. Таким образом, в рамках дальнейшей универсализации ЭВМ возможна и необходима специализация некоторых функций.

Производительность ЭВМ определяется прежде всего быстродействием ее элементов, а также организацией их соединения и взаимодействия, т. е. структурой ЭВМ [1,2].

Структура ЭВМ за последнюю четверть века эволюционно развивалась в направлении оптимизации взаимодействия узлов и блоков, совершенствования организации вычислительного процесса и расширения функций ЭВМ, не выходя за рамки принципов организации вычислительного процесса, сформулированных фон Нейманом, когда процесс обработки информации носит последовательный характер, т. е. команды извлекаются из памяти и выполняются последовательно одна за другой в программно заданном порядке.

Структура ЭВМ общего назначения однозначно определяет и способы ее оптимизации по критерию отношения «производительность — стоимость»:

сокращение длительности машинного цикла за счет уменьшения числа каскадов в логической цепи обработки;

увеличение емкости и быстродействия буферной и основной памяти;

введение совмещения выполнения операции (конвейерной обработки) в процессоре;

уменьшение количества циклов на одну операцию за счет совершенствования алгоритмов выполнения команд в процессоре и увеличения степени параллельности обработки.

Для рассматриваемой нами структуры пределом повышения производительности является выполнение одной команды за один машинный цикл. В наиболее высокопроизводительных ЭВМ общего назначения достигнут уровень выполнения одной команды за три машинных цикла. Это показывает, что за счет оптимизации структуры можно ожидать увеличения производительности ЭВМ общего назначения в три раза. Этот процесс достаточно сложен и протекает относительно медленно. Кроме того, намечившийся переход на элементную базу четвертого поколения (БИС и СБИС) несколько затормозит усложнение структуры, поскольку на первом этапе одновременное решение двух сложнейших научно-технических проблем весьма затруднительно. Поэтому в ближайшее время следует ожидать, что развитие ЭВМ общего назначения пойдет не столько по пути развития внутренней структуры процессора, сколько по пути:

расширения состава операций;

увеличения емкости и совершенствования организации оперативной памяти;

развития системы ввода-вывода;

расширения средств диагностики и восстановления;

организации многопроцессорных и многомашинных вычислительных систем обработки информации;

создания проблемно-ориентированных спецпроцессоров.

Создание ЭВМ общего назначения, в том числе ЕС ЭВМ, привело к глубокой стандартизации принципов работы и архитектуры ЭВМ. В этих условиях дальнейшее совершенствование архитектуры и повышение производительности ЭВМ общего назначения с учетом значительного и дорогостоящего задела по операционным системам и прикладным программам будут в ближайшие годы при переходе к четвертому поколению носить эволюционный характер и опираться прежде всего на физико-технологические достижения в области элементной базы и электронного конструирования.

Учитывая быстрое развитие микроэлектроники, можно прогнозировать дальнейший рост производительности ЭВМ общего назначения в ближайшие 15—20 лет без существенного изменения их структуры [3]. Такое повышение скорости обработки информации может быть достигнуто только при условии решения комплекса



сложнейших научно-технических проблем, прежде всего связанных с переходом на новую технологию. Четвертое поколение ЭВМ ориентировано на широкое применение в качестве элементной базы БИС, поэтому оценка возможностей и путей повышения их быстродействия с учетом физических и конструктивных ограничений представляет значительный научный и практический интерес для проектирования высокопроизводительных ЭВМ.

При выполнении логических элементов в виде интегральной полупроводниковой схемы совершенствование технологии и уменьшение геометрических размеров компонентов, как правило, сопровождается увеличением количества логических элементов на кристалле (т. е. повышением степени интеграции). При уменьшении размеров элементов и повышении степени интеграции из-за тепловых ограничений приходится во столько же раз снижать потребляемую каждым элементом мощность, в результате чего задержка переключения логических элементов в составе интегральной схемы с максимальной степенью интеграции практически не изменяется, несмотря на уменьшение их размеров и снижение энергии переключения [4]. Таким образом, сам по себе факт совершенствования технологии и уменьшения размеров элементов в БИС с максимальной степенью интеграции еще не означает повышение быстродействия логических элементов.

Повышение быстродействия полупроводниковых элементов в составе БИС может быть достигнуто за счет повышения удельной мощности на элементах, применения эффективных методов охлаждения и увеличения допустимой мощности рассеивания на кристалле, а также за счет совершенствования схемотехники и технологии, снижения амплитуды рабочих сигналов и использования новых полупроводниковых материалов. Все эти казалось бы противоречивые требования могут быть реализованы в случае использования матричной структуры размещения логических элементов БИС на базовых матричных кристаллах. Базовые матричные кристаллы позволяют решить и проблему расширения номенклатуры БИС высокопроизводительных ЭВМ, которая возникает при повышении степени интеграции логических элементов.

БИС, созданные на основе базовых матричных кристаллов, обеспечивают высокую информационную плотность, короткий цикл изготовления, низкую стоимость и высокое быстродействие по сравнению с заказными логическими схемами и микропроцессорами.

Однако повышение быстродействия самих БИС не приводит однозначно к повышению быстродействия ЭВМ, поскольку быстродействие ЭВМ определяет величина системной задержки, равная сумме задержки прохождения сигнала в логических элементах и в линиях связи. Возникает задача оптимального согласования параметров БИС с параметрами конструкции ЭВМ более высокого уровня, т. е. задача оптимального электронного конструирования высокопроизводительных ЭВМ. Таким образом, рост степени интеграции элементов, рост количества контактов, рост удельного тепловыделения и

быстродействия БИС приводит к необходимости повышения плотности монтажа и трассировочной способности многослойных печатных плат (МПП) [5,6]. Следует ожидать, что технология МПП в усовершенствованном виде сохранится и для ЭВМ четвертого поколения. Для того чтобы поднять производительность ЭВМ, необходимо сократить задержки в межсоединениях элементов при повышении степени интеграции БИС, что в свою очередь повлечет как увеличение количества слоев многослойных печатных плат, так и повышение трассировочной способности каждого слоя, а также увеличение плотности монтажных отверстий.

Несмотря на усложнение технологии, следует ожидать дальнейшего снижения стоимости центральной (электронной) части ЭВМ, так как исходя из тенденций развития микроэлектронной элементной базы стоимость БИС растет гораздо медленнее, чем степень интеграции, а развитие структуры ЭВМ в ближайшее время не требует значительного увеличения функционального объема электронного оборудования процессора. Еще в большей степени будет снижено отношение «стоимость — производительность», поскольку технология ЭВМ четвертого поколения, основанная на применении БИС, резко увеличивает производительность ЭВМ.

Высокий уровень технологической сложности ЭВМ четвертого поколения ставит перед разработчиком еще две взаимосвязанные задачи — повышение уровня автоматизации проектирования ЭВМ и создание системы бездефектного проектирования БИС широкой номенклатуры. Резкий рост степени интеграции элементов на БИС сильно усложняет функциональное проектирование и практически исключает возможность внесения изменений в процессе наладки из-за большой длительности цикла проектирования и изготовления БИС. Для решения этой проблемы необходимо разработать и внедрить мощную систему автоматизации на всех этапах проектирования, наладки опытных образцов и изготовления ЭВМ.

Таким образом, рассматривая перспективу развития высокопроизводительных ЭВМ общего назначения и проблемы, возникающие при этом, можно сделать ряд важных выводов.

1. На первом этапе создания ЭВМ четвертого поколения резкое улучшение основных технико-экономических параметров будет достигнуто за счет совершенствования элементно-конструктивной базы на основе БИС и СБИС. Можно ожидать, что в ближайшие 15—20 лет производительность ЭВМ возрастет на один-два порядка за счет развития микроэлектронной технологии.

2. Развитие микроэлектронной элементной базы неизбежно приведет к необходимости непрерывного совершенствования уровня технологии электромонтажа и конструкции ЭВМ.

3. Необходимым условием реализации достижений микроэлектроники при создании высокопроизводительных ЭВМ является создание новой, более совершенной технологии проектирования, наладки и изготовления ЭВМ на основе автоматизации всех этапов.

OldPC.su

5 0 5 0

музей компьютеров

4. Несмотря на высокий уровень технологической сложности ЭВМ четвертого поколения, следует ожидать снижения стоимости центральной части ЭВМ и, в еще большей степени, снижения отношения «стоимость — производительность».

#### ЛИТЕРАТУРА

1. Глушков В. П. **Научные проблемы развития вычислительной техники.** — Вестник АН СССР, 1976, № 2, с. 27—44.
2. Пржиялковский В. В., Ломов Ю. С., Файзулаев Б. Н. **Проблемы и пути технической реализации высокопроизводительных ЭВМ на основе БИС.** — Управляющие системы и машины, 1980, № 6, с. 15—23.
3. Ломов Ю. С., Файзулаев Б. Н. **Прогноз развития технической базы высокопроизводительных ЭВМ.** — Вопросы радиоэлектроники. Сер. ЭВТ, 1981, вып. 5, с. 37—45.
4. Файзулаев Б. Н., Первов А. С. **Взаимосвязь предельного быстродействия и степени интеграции БИС.** — В кн.: Микроэлектроника и полупроводниковые приборы. — М.: Радио и связь, 1979, вып. 4, с. 149—156.
5. Чунаев В. С. **Вероятное направление развития полупроводниковых БИС для быстродействующих ЭВМ.** — Микроэлектроника, 1977, т. 6, вып. 2, с. 99—107.
6. Вербицкий Г. Г., Уинклер П., Хейнинг Ф. В. **Повышение плотности монтажа компьютеров более чем в 15 раз.** — Электроника, 1979, т. 52, № 16, с. 37—45.

УДК 621.3.049.77:325.65

### **БИС НА ОСНОВЕ БАЗОВОГО МАТРИЧНОГО КРИСТАЛЛА — ЭЛЕМЕНТНАЯ БАЗА ВЫСОКОПРОИЗВОДИТЕЛЬНЫХ ЭВМ ЕДИНОЙ СИСТЕМЫ**

**Б. Н. Файзулаев**, д-р техн. наук (СССР)  
**Н. М. Малярский**, канд. техн. наук (СССР)

Требование совместимости ЭВМ Единой системы определяет тенденцию к максимальному сохранению структуры ЭВМ и принципов работы, а резервы повышения производительности ЭВМ за счет усложнения функциональной структуры отдельных узлов основных обрабатывающих устройств практически исчерпаны, поэтому основным путем повышения производительности старших ЭВМ Единой системы, в которых предельно используются возможности элементной базы, является увеличение системного быстродействия элементной базы.

Увеличение системного быстродействия элементной базы невозможно без существенного сокращения задержек сигналов в межсоединениях, а следовательно, без увеличения плотности компоновки элементов в конструкциях ЭВМ.

При типовом значении задержки распространения на элементарную функцию в интегральных схемах (ИС) серии 500, равном 2 нс, и средней степени интеграции на корпус ИС 10—15 вентилях систем-

ная задержка переключений элементов в базовых конструкциях высокопроизводительных ЭВМ типа ЕС-1060 равна 4—5 нс. Для повышения производительности ЭВМ необходимо соответственно уменьшать значение системной задержки переключений элементов. Сокращение системной задержки может быть достигнуто повышением быстродействия логических элементов и снижением задержек сигналов в межсоединениях за счет использования больших интегральных схем (БИС) и более совершенных конструкций ЭВМ [1, 2].

Совершенство технологических процессов в МОП и биполярной технологии привело к широкому производству БИС микропроцессорного типа и БИС запоминающих устройств. В высокопроизводительных ЭВМ общего назначения, какими являются ЭВМ Единой системы, применение БИС до последнего времени ограничивалось только устройствами, имеющими регулярную структуру, такими, как оперативная и сверхоперативная память, а также некоторыми узлами периферийных устройств, в которых возможно применение унифицированных БИС микропроцессорного типа.

Появление матричных БИС (МаБИС) открывает новый этап в развитии вычислительной техники, который должен привести к сокращению сроков и затрат на разработку БИС, к дальнейшему росту производительности и увеличению надежности ЭВМ.

При проектировании и использовании МаБИС первоначально разрабатывается и изготавливается базовый матричный кристалл (БМК), содержащий некоторое количество регулярно расположенных полупроводниковых структур. Каждая повторяющаяся полупроводниковая структура (макроячейка) может содержать одну или несколько логических схем либо набор компонентов, на основе которого может быть выполнена целая серия различных логических элементов (библиотека логических элементов).

На основе БМК проектируются различные типы МаБИС, при этом функциональная настройка макроячеек и матрицы в целом осуществляется с помощью специально проектируемых межэлементных соединений. Для различных типов МаБИС переменными являются слои металлизации межсоединений, при этом комплект фотомасок для изготовления собственно БМК не изменяется. Этот метод проектирования заказных БИС дает возможность решить основную проблему реализации широкой номенклатуры БИС для высокопроизводительных ЭВМ Единой системы.

Основной задачей создания МаБИС широкой номенклатуры на основе БМК является задача «бездефектного» проектирования. Так как МаБИС для высокопроизводительных ЭВМ Единой системы, как правило, не являются функционально законченными частями ЭВМ, то правильность функционирования таких БИС может быть проверена достаточно хорошо только в комплексе ЭВМ при взаимодействии с другими БИС. С этой целью при разработке на БИС высокопроизводительных ЭВМ Единой системы предусматривается разработка не только традиционных методов проверки правильности проектирования и функционирования БИС (моделирование на

этапе проектирования БИС и ЭВМ, контроль функционирования при изготовлении и т. д.), но и физических моделей ЭВМ на основе типовых элементов замены (ТЭЗ), являющихся прототипами для построения ЭВМ на БИС. Такие ЭВМ могут быть переходными моделями в развитии технических средств ЕС ЭВМ и одновременно схемно-функциональными прототипами и физическими моделями будущих ЭВМ Единой системы на БИС.

Создание схемно-функциональных прототипов ЭВМ на БИС позволит проверить структурно-функциональные решения, отработать систему микродиагностических тестов и новые технические средства, расширяющие и наращивающие функции ЕС ЭВМ.

Создание переходных моделей-прототипов обеспечивает практически «бездефектный» технологический переход к созданию ЭВМ на основе БИС по принципу реализации «ТЭЗ → БИС» и позволяет провести физическое и функциональное моделирование БИС в составе ЭВМ на отработанном программном обеспечении, выверить таблицы соединений на всех конструктивных уровнях, отработать систему проектирования тестов и средства контроля БИС.

Основные ограничения, накладываемые на проектирование конструктивных и функциональных узлов ЭВМ прототипа, и в первую очередь на проектирование ТЭЗ, определяются возможностями их дальнейшей реализации на основе БМК и основными параметрами БМК. К таким ограничениям относятся функциональный объем (число эквивалентных вентиляей), количество контактов для логических связей, количество входных и выходных выводов корпуса, контролепригодность схемы БИС и т. д.

ЭВМ на основе БИС может разрабатываться и изготавливаться параллельно с ЭВМ прототипа, причем возможна одновременная разработка конструкции ЭВМ на основе БИС, разработка основной памяти устройства электропитания и других устройств ЭВМ на БИС. (Только МаБИС и многослойные печатные платы с БИС проектируются и изготавливаются после окончательной отработки ЭВМ прототипа.) Такое совмещение позволит значительно сократить время создания ЭВМ.

Конструктивно-технологическая база современных высокопроизводительных ЕС ЭВМ и система автоматизированного проектирования базируются на использовании ЭСЛ ИС серии 500 и ТЭЗ со следующими характеристиками:

среднее количество эквивалентных вентиляей в ИС — 10–15;

типовое значение задержки переключения вентиля в ИС — 2 нс;

среднее значение функциональной емкости ТЭЗ — 500 эквивалентных вентиляей (максимальное значение функциональной емкости 1200—1500);

максимальное количество контактов разъема ТЭЗ, используемых для логических связей, — 90.

Электроника процессора современных высокопроизводительных ЭВМ Единой системы размещается в типовой стойке, количество типов ТЭЗ — 200—300, общее количество ТЭЗ — 500—700.

Для технического перехода от ЭВМ прототипа к ЭВМ на основе БИС при одновременном повышении быстродействия элементной базы необходимо использовать БМК с функциональными характеристиками, аналогичными характеристикам ТЭЗ.

функциональная емкость — до 1500 эквивалентных вентиляей;  
 количество выводов корпуса под логические связи — не менее 90;

библиотека функций основных макроячеек, аналогичная библиотеке ИС серии 500.

При разработке технической базы ЭВМ на основе БИС повсеместно должен соблюдаться принцип пропорционального «сжатия» временной диаграммы (по конструктивным уровням процессора, по системной задержке элементной базы и запоминающих устройств). При компоновке функциональных схем ЭВМ прототипа проводится оценка

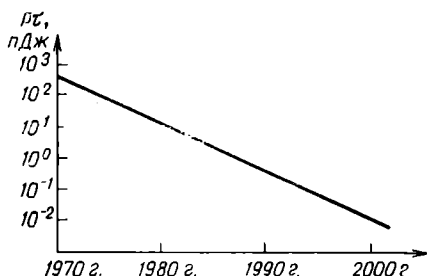


Рис. 1. Прогноз снижения энергии переключения  $P\tau$

возможного коэффициента «сжатия» временной диаграммы при переходе на МаБИС с учетом конструктивной реализации логических цепей, определяющих такт синхронизации.

Конструктивно-технологическая база ЭВМ на основе БИС должна в сравнении с конструктивно-технологической базой современных ЭВМ Единой системы иметь более высокую плотность печатного монтажа и компоновки логических элементов в единице конструктивного объема, а также более высокую удельную мощность источников электропитания, высокую стабильность и равенство волновых сопротивлений линий связи на различных конструктивных уровнях.

Применение новой микроэлектронной элементной базы на основе матричных БИС и совершенствование базовых конструкций ЕС ЭВМ позволит создать следующее поколение процессоров ЕС ЭВМ со значительно более высокой производительностью.

Основным параметром, характеризующим уровень технологии логических элементов, является энергия переключения ( $P\tau$ ), которая представляет собой произведение мощности на задержку переключения и всесторонне определяет качество интегральных логических элементов. За последние двадцать лет величина энергии переключения полупроводниковых логических элементов снизилась почти на три порядка, уменьшаясь ежегодно в 1,5—2 раза (рис. 1). Причем прогнозируется дальнейшее снижение энергии переключения примерно с тем же темпом до уровня  $10^{-2}$  пДж к 2000 г. [3].

Производительность процессора ЭВМ непосредственно определяется системной задержкой логических элементов  $\tau_c$ :

$$\tau_c = \tau_{лз} + \tau_{св}, \quad (1)$$

где  $\tau_c$  — усредненное время системной задержки на один каскад логической цепи;

$\tau_{лэ}$  — средняя задержка на каскад логического элемента (схемная задержка);

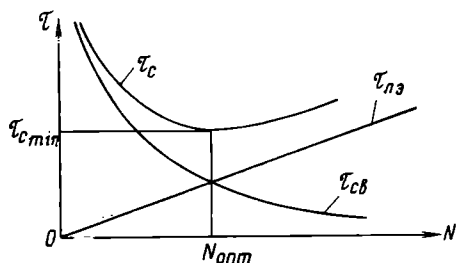


Рис. 2. Зависимость системной задержки  $\tau_c$  от степени интеграции элементов  $N$  при заданной энергии переключения  $P\tau = \text{const}$

некотором оптимальном значении степени интеграции, когда время задержки в логических элементах  $\tau_{лэ}$  становится равным времени задержки в соединениях между логическими элементами  $\tau_{лэ}$  (рис. 2):

$$\tau_{лэ} = \tau_{св}. \quad (2)$$

Любое другое соотношение, когда  $\tau_{лэ} \ll \tau_{св}$  или  $\tau_{лэ} \gg \tau_{св}$ , не соответствует оптимальному выбору параметров элементной и конструктивной базы и не обеспечивает предельно достижимого быстродействия процессора ЭВМ для данного технологического уровня развития элементов и электромонтажа. По мере совершенствования микроэлектронной технологии и техники электромонтажа снижение задержки переключения логических элементов  $\tau_{лэ}$  должно сопровождаться соответствующим снижением конструктивных задержек в линиях связи  $\tau_{св}$ . Поскольку конструктивная задержка  $\tau_{св}$  снижается обратно пропорционально росту степени интеграции БИС, условие (2) можно записать в виде:

$$K_F = K_N, \quad (3)$$

где  $K_F$  — коэффициент роста быстродействия логических элементов;

$K_N$  — коэффициент роста степени интеграции БИС.

При заданной энергии переключения  $P\tau$  и мощности на кристалле  $P_{кр}$  для матричных БИС справедливо следующее фундаментальное соотношение между быстродействием  $\tau_{лэ}$  и степенью интеграции  $N$  логических элементов на кристалле:

$$\frac{P_{кр} \cdot \tau_{лэ}}{N} \approx P\tau \quad (4)$$

и, следовательно,

$$K_N \cdot K_F \approx K_{P\tau} \cdot K_P, \quad (5)$$

где  $K_{P\tau}$  — коэффициент снижения энергии переключения;  
 $K_P$  — коэффициент роста мощности на кристалле.

Используя соотношения (3) и (5), связывающие параметры элементной и конструктивной базы ЭВМ, нетрудно получить теоретическую оценку роста быстродействия процессоров при оптимальном электронном конструировании ЭВМ:

$$K_F = \sqrt{K_{P\tau} \cdot K_P},$$

где  $K_F$  — коэффициент повышения системного быстродействия процессора ЭВМ за счет элементной базы и конструкции.

Учитывая, что энергия переключения логических элементов уменьшается ежегодно в среднем в 1,5 раза [3], можно прогнозировать темпы снижения системной задержки в процессорах высокопроизводительных ЭВМ, т. е. темпы роста их быстродействия для конструкции с воздушным и жидкостным охлаждением (рис. 3).

С ростом степени интеграции и быстродействия БИС возникает ряд конструктивно-технологических проблем;

увеличения количества контактов БИС, повышения плотности электромонтажа БИС, роста удельного тепловыделения БИС, контролепригодности;

расширения номенклатуры БИС и уменьшения их повторяемости в ЭВМ.

Наиболее серьезной задачей на пути технической реализации БИС для высокопроизводительных ЭВМ является «бездефектность» проектирования БИС, т. е. окончательная отработка сложных функциональных схем до изготовления опытных образцов БИС и ЭВМ в целом. Если в машинах первого, второго и третьего поколений можно было легко вносить изменения в сложные функциональные схемы ЭВМ в процессе их наладки, то с переходом на БИС такая возможность практически исключается из-за длительного процесса перепроектирования и изготовления БИС. Это ставит перед разработчиками высокопроизводительных ЭВМ четвертого поколения задачу «бездефектного» проектирования БИС широкой номенклатуры на основе базовых матричных кристаллов и автоматизации всех этапов разработки и изготовления.

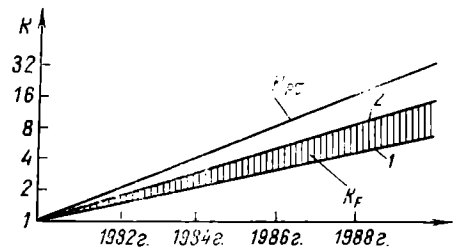


Рис. 3. Прогноз роста быстродействия процессоров высокопроизводительных ЭВМ:

- 1 — воздушное охлаждение БИС ( $K_P=1$ );  
 2 — жидкостное охлаждение БИС ( $K_P=4$ )



## ЛИТЕРАТУРА

1. Пржиялковский В. В., Ломов Ю. С., Файзулаев Б. И. Проблемы и пути технической реализации высокопроизводительных ЭВМ на основе БИС. — Управляющие системы и машины, 1980, № 6, с. 15—23.
2. Файзулаев Б. Н. Проблема быстродействия элементной базы ЭВМ. — В кн.: Микроэлектроника и полупроводниковые приборы. — М.: Радио и связь, 1981, вып. 6, с. 5—30.
3. Keyes R. W. The Evolution of Digital Electronics Towards VLSI. IEEE Journ. of Solid. — State Circuits, 1979, vol. 14, No 2, p. 193—201.

УДК 681.327.28

### ПОВЫШЕНИЕ БЫСТРОДЕЙСТВИЯ И КОЭФФИЦИЕНТА ПРОГРАММИРУЕМОСТИ БИПОЛЯРНЫХ ППЗУ СЕРИИ КР556

Ю. И. Щетинин, канд. техн. наук (СССР)

Среди устройств обработки данных серийного производства трудно найти такие, в которых не использовались бы программируемые постоянные запоминающие устройства (ППЗУ) серии КР556. Они применяются в генераторах алфавитно-цифровых символов и преобразователях кода, для хранения и табличного поиска данных и выполнения математических операций. Появление микропроцессорных комплектов БИС расширяет сферу применения ППЗУ [1].

Электронной промышленностью выпускаются микросхемы ППЗУ типа КР556РТ4 и КР556РТ5 или в модифицированном варианте с индексом А.

Микросхема КР556РТ4А представляет собой функционально законченное постоянное запоминающее устройство с электрическим программированием емкостью 1024 бита с организацией 256 четырехразрядных слов. Основными узлами схемы являются (рис. 1): запоминающая матрица, адресные формирователи, дешифраторы строк и столбцов, мультиплексоры, усилители считывания и схемы программирования. Кроме того, на кристалле содержатся непоказанные на функциональной схеме двухуровневые стабилизированные источники опорных напряжений, дополнительные поля матрицы для контроля дешифраторов и резервная словарная шина для повышения выхода годных микросхем в процессе производства.

Запоминающая матрица выполнена в виде 33-многоэмиттерных транзисторов (МЭТ) по 34 эмиттера в каждом. Включение транзисторов по схеме с общим коллектором позволяет разместить все транзисторы матрицы в одном изолированном кармане кристалла, что существенно уменьшает занимаемую матрицей площадь.

Программируемые элементы, выполненные из пленки никеля и хрома толщиной 300А, включены между эмиттерами транзисторов матрицы и разрядными шинами. Наличие плавкой перемычки соот-

ветствует логическому нулю для КР556РТ4А и логической единице для КР556РТ5А на выходе усилителя считывания.

В [2] показано, что наибольшая неустойчивость перемычки, а следовательно, и наилучший процесс программирования отмечен

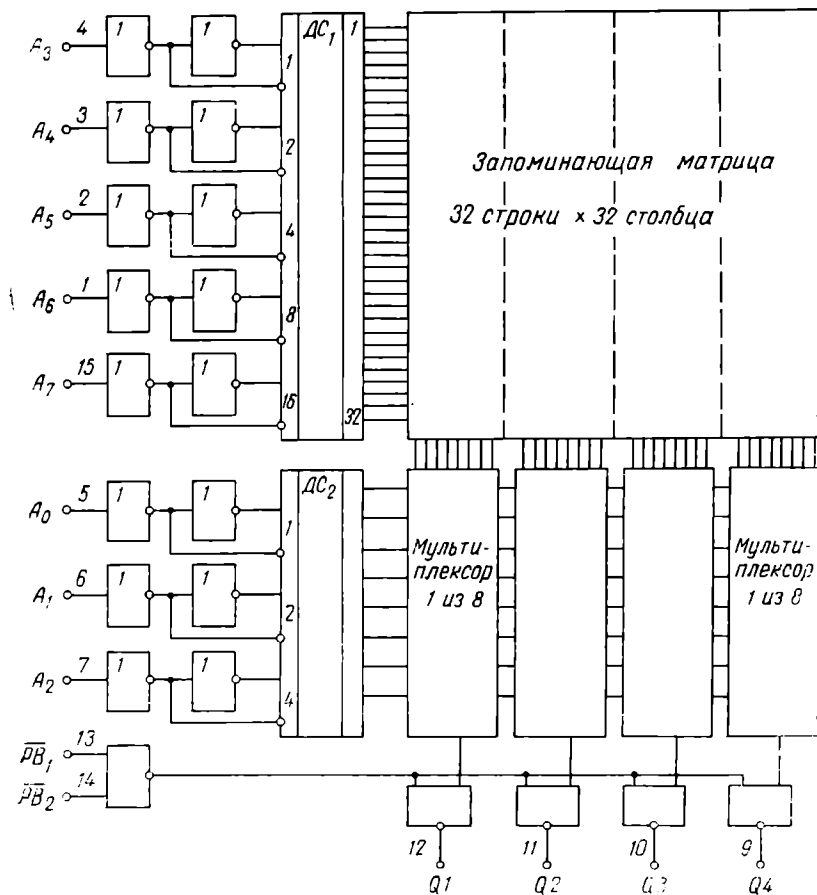


Рис. 1. Функциональная схема ППЗУ емкостью 1024 бита типа КР556РТ4А:

$A_0$ — $A_7$  — адресные входы;  $PB_1$  и  $PB_2$  — входы разрешения выборки;  $ДС_1$  — дешифраторы строк;  $ДС_2$  — дешифраторы столбцов;  $Q_1$ — $Q_2$  — выходы 1—4-го разрядов

для плавкой перемычки с прямолинейными боковыми поверхностями. Прямоугольная форма перемычки позволила с использованием 3—4-микронных размеров реализовать транзисторную ячейку размером 900 мкм<sup>2</sup>, что, например, для КР556РТ4А в 2,5 раза меньше, чем в первоначальном варианте ППЗУ. Конструкция фрагмента матрицы ППЗУ с указанной ячейкой приведена на рис. 2. Размер активной части плавкой перемычки составляет 3×3 мкм<sup>2</sup>. Ширина

перемычки определяется возможностями фотолитографии по никелю и хрому, а длина перемычки соответствует расстоянию между соседними алюминиевыми дорожками нижнего слоя металлизации.

В новой ячейке ППЗУ сведены к минимуму площади активных и пассивных полупроводниковых областей, а также впервые используются два базовых контакта на один бит, благодаря чему вдвое

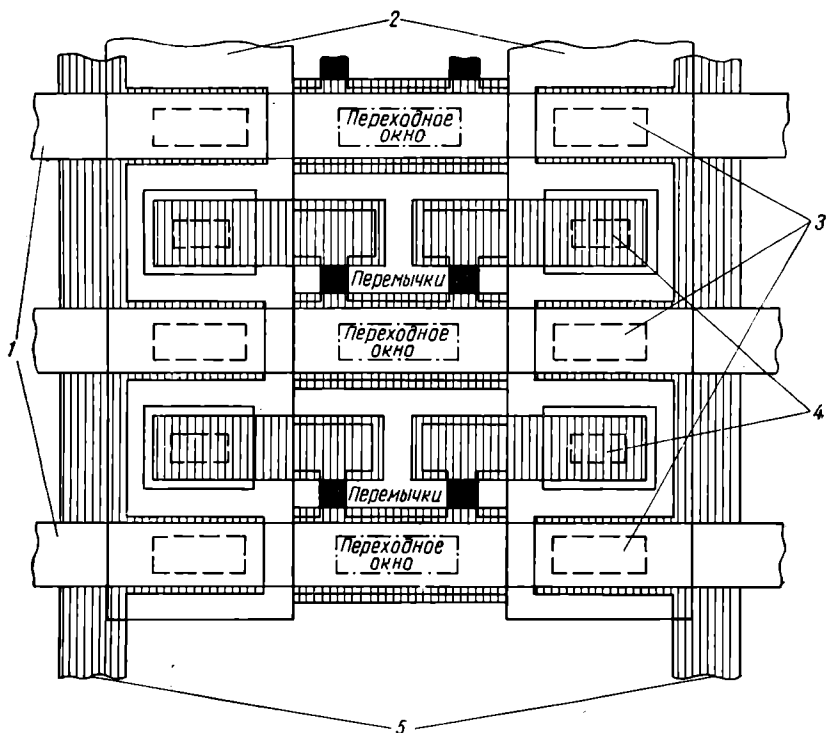


Рис. 2. Конструкция фрагмента матрицы:

- 1 — разрядные шины; 2 — базовые области транзистора; 3 — контакт к базе;  
4 — контакт к эмиттеру; 5 — словарные шины

повышается эффективность использования эмиттера. Удалось создать ячейку, которая при размерах 4 мкм позволяет реализовать ППЗУ емкостью до 16 Кбит, а при переходе на размеры 2—3 мкм — до 64 Кбит. Дополнительная защита плавкой перемычки межслойным диэлектриком значительно повышает надежность ППЗУ в условиях повышенной влажности.

Адресные формователи служат для получения прямых и инверсных значений адресного кода. Они выполнены по схеме входного вентиля типа ТТЛ с диодами Шоттки с входным эмиттерным повторителем на *p-n-p* транзисторе [3].

Запись информации в схему представляет собой избирательное разрушение плавких перемычек. До программирования все эмиттеры транзисторов запоминающей матрицы соединены с разрядными

шинами и, следовательно, при обращении по любому адресу на все выходы схемы выдается одинаковая информация. Это создает определенные трудности при контроле схемы в условиях производства, так как не позволяет проверить правильность работы всех функциональных узлов схемы, в частности адресных формирователей, дешифраторов строк и столбцов, мультиплексоров. Для контроля этих узлов матрица содержит дополнительную строку и два дополнительных столбца, которые используются при разбраковке кристаллов на пластине [4] и недоступны потребителю.

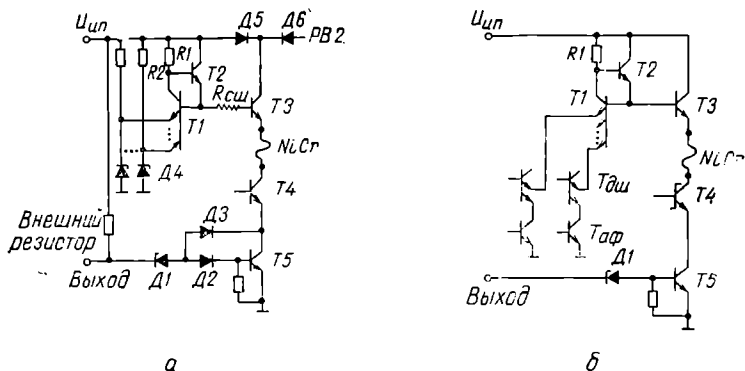


Рис. 3. Тракт программирования ППЗУ типа КР556РТ4 (а) и КР556РТ4А (б)

В ППЗУ последняя операция, определяющая выполняемую функцию, реализуется разработчиком аппаратуры. Такой операцией является программирование, точность соблюдения режимов которого обеспечивает высокий выход годных после данной операции и безотказную работу.

На рис. 3 приведены тракт программирования микросхемы КР556РТ4А и для сравнения — тракт программирования микросхемы КР556РТ4. Транзисторы Т1, Т2 и резистор R1 реализуют один из 32 элементов дешифратора строк. Входное сопротивление дешифратора определяется номиналом резистора R1, а выходное сопротивление почти на два порядка меньше благодаря включению транзистора Т2 по схеме эмиттерного повторителя. Коллекторная цепь транзистора матрицы Т3 микросхемы КР556РТ4 в режиме считывания получает напряжение от шины питания  $U_{пит}$  через диод Д5. Чтобы предотвратить насыщение транзистора Т3, которое возникает из-за большого падения напряжения на сопротивлении тела коллектора, на него подается напряжение 15 В от вывода разрешения выборки РВ2 через диод Д6. Модернизированная микросхема КР556РТ45 отличается от серийно выпускаемой следующим:

снижено сопротивление тела коллектора Т3, что позволило уменьшить подаваемое на коллектор Т3 напряжение до 12,5 В, т. е. до величины напряжения, подаваемого на вывод питания в режиме программирования, и отсоединить вывод РВ2 от матрицы. Хотя на-

пряжение 15 В теперь не требуется, однако его наличие на входе РВ2 программированию не мешает. Это позволяет программировать микросхемы КР556РТ4А на аппаратуре для программирования КР556РТ4 без какой-либо доработки;

в тракте программирования КР556РТ4А диффузионные словарные шины заменены на металлические; устранено сопротивление  $R_{сш}$ ;

снижено напряжение логического нуля, подаваемого в режиме

**Режим программирования микросхем  
КР556РТ4А, КР556РТ4**

| Режим программирования | Длительность импульса, мкс | Количество импульсов |
|------------------------|----------------------------|----------------------|
| Нормальный             | 25                         | 1 000                |
| Форсированный          | 10                         | 100                  |
| Дополнительный         | 25                         | 40—100               |

программирования на базу выходного транзистора мультиплектора Т4, что позволило исключить диоды Д2, Д3 и перевести программирующий транзистор Т5 в режим насыщения;

уменьшено сопротивление тела коллектора транзисторов Т4 и Т5;

транзистор Т3 выполнен с двумя базовыми контактами, что повышает коэффициент усиления при больших плотностях тока и снижает падение

Примечание. Во всех режимах скважность импульсов не менее 10; фронт импульсов — 300—500 нс.

напряжения на переходе база — эмиттер;

повышено напряжение на входе дешифратора строк. Для схемы КР556РТ4 это напряжение определялось стабилизаторами на выходе адресных формирователей. На рис. 3, а показан один из них на элементах  $R2$ , Д4. Выходное напряжение такого стабилизатора составляет 7,6 В, оно зависит от напряжения пробоя перехода база—эмиттер, на котором выполнен стабилитрон Д4. Для схемы КР556РТ4А входное напряжение дешифратора равно 9 В, оно представляет собой сумму напряжений пробоя перехода база—эмиттер  $T_{дш}$ , падения напряжения на прямосмещенном переходе база — эмиттер и напряжения логического нуля адресного формирователя  $T_{аф}$ . Только одно это усовершенствование позволяет увеличить ток через программируемую перемычку почти на 20 мА.

В целом при модернизации микросхем ППЗУ емкостью 1 Кбит и 4 Кбит программирующий ток увеличился с 35—40 до 70—80 мА.

При программировании устанавливается адрес выбранного слова, на выход выбранного разряда задается ток 3—6 мА и подается напряжение 12,5 В на шину питания. Информация о длительности и количестве подаваемых импульсов приведена в таблице.

Форма подаваемых импульсов и их взаимное расположение представлены на рис. 4.

Наличие дополнительного режима при программировании требует пояснения. Как уже упоминалось, ППЗУ серии КР556 выполняется на основе никель—хромовых плавких перемычек. Этот материал характеризуется малой длиной свободного пробега электро-

нов (40 А), а сопротивление пленки толщиной более 100 А практически не отличается от объемного сопротивления сплава. Малая длина свободного пробега электронов не позволяет им приобрести момент количества движения, достаточный для перемещения атомов никеля и хрома, что исключает разрушение пленки за счет электромиграции.

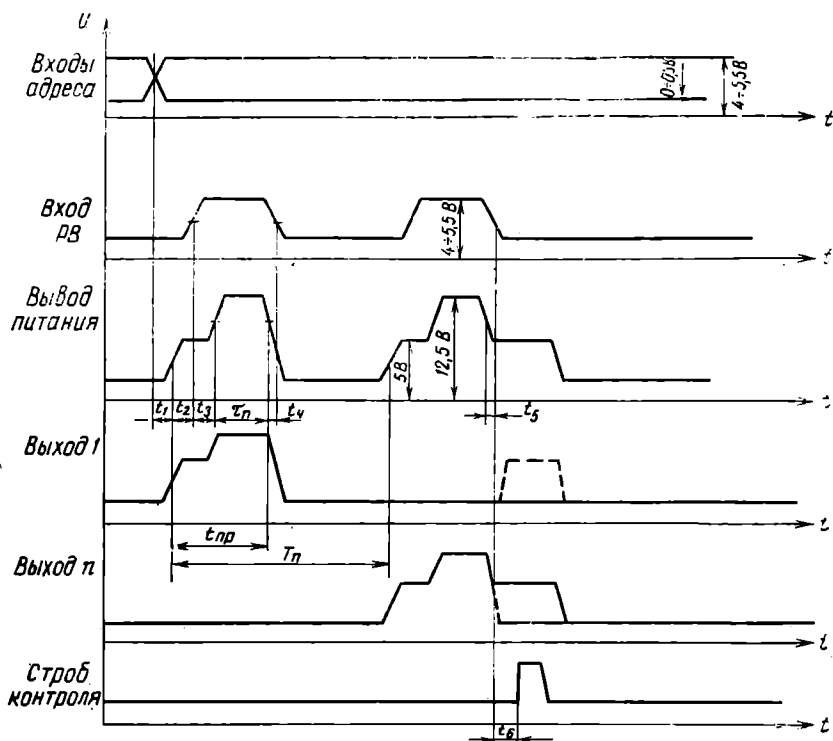


Рис. 4. Форма импульсов, подаваемых при программировании микросхем КР556РТ4А и КР556РТ5А:

$t_n$  — длительность импульса программирования, равная 25 мкс;  $t_1$  — время предварительного установа адресных входов относительно напряжения питания;  $t_2$  — время предварительного установа напряжения питания относительно запрета выборки;  $t_3$  — время предварительного установа запрета выборки относительно напряжения программирования;  $t_4$  — время сохранения запрета выборки относительно напряжения питания;  $t_5$  — время задержки импульса разрешения выборки относительно напряжения питания при контроле;  $t_6$  — время задержки строба контроля;  $t_1 - t_6 > 1$  мкс;

$T_n$  — период следования импульсов;  $\frac{T_n}{t_n} > 10$  — скважность программирующих импульсов

Основным механизмом программирования никель-хромовых плавких перемычек при достаточной мощности программирующего импульса является плавление с последующим образованием пустот под действием сил поверхностного натяжения и движения ионов в электрическом поле. Температура плавления сплава никель-хром

составляет около  $1450^{\circ}\text{C}$ . Если коэффициент усиления выходного транзистора мультиплексора мал и программирование осуществляется слаботочным импульсом, то данная температура не будет достигнута, а программирование будет осуществляться медленно за счет окисления никель-хрома кислородом. При этом перемишка, воспринимаемая усилителем считывания при нормальных условиях как запрограммированная, с увеличением чувствительности усилителя считывания при более высоких температурах или при увеличении напряжения питания может восприниматься как незапрограммированная. Для борьбы с этим явлением, а также для реализации ускоренного режима тренировки запрограммированных перемишек методикой программирования микросхем серии КР556 предусматривается подача дополнительных 40—100 импульсов после фиксации усилителем момента пережигания (дополнительный режим). Микросхемы КР556РТ4А и КР556РТ5А содержат схему повышения чувствительности усилителя при программировании, что позволяет сократить количество тренировочных импульсов до нескольких единиц и повысить скорость программирования. Использование этих возможностей требует незначительной доработки программатора.

В результате модернизации ППЗУ повысились программируемость микросхем и их быстродействие. Типовое значение программируемости микросхемы КР556РТ4 составляет 70%, микросхемы КР556РТ5 — 80%. Специальные испытания программируемости новых микросхем на выборке по 1000 образцов показали программируемость выше 97%. Это означает, что разработчики радиоаппаратуры могут программировать ППЗУ группами по 16—32 прибора после установки на печатные платы.

Совершенствование конструкции ячейки памяти, исключение диффузионных каналов и уменьшение площади активных компонентов позволили уменьшить время выборки адреса для КР556РТ4А с 70 до 45 нс и для КР556РТ5А — с 70 до 50 нс в диапазоне температур от  $-10^{\circ}\text{C}$  до  $+70^{\circ}\text{C}$ .

Учитывая единое конструктивное исполнение, полную взаимозаменяемость по статическим параметрам, более высокий коэффициент программируемости и более высокую надежность микросхемы КР556РТ4А и КР556РТ5А, можно считать, что использование новых микросхем позволит повысить быстродействие, надежность и снизить себестоимость устройства обработки информации.

#### ЛИТЕРАТУРА

1. Березенко А. П., Корягин Л. Н., Щетинин Ю. И. **Микропроцессорный комплект биполярных БИС.** — В кн.: Микроэлектроника и полупроводниковые приборы. — М.: Радио и связь, 1977, вып. 2, с. 80—94.
2. Щетинин Ю. И., Штейнгард З. А., Ивацов Е. Н., Быстрова В. Б. **Некоторые механизмы программирования элементов ППЗУ на основе тонких пленок.** — В кн.: Микроэлектроника и полупроводниковые приборы. — М.: Радио и связь, 1981, вып. 6, с. 121—132.

3. Щетинин Ю. И., Березенко А. И., Базанов В. И., Марков Б. А. Схемотехника микропроцессорного комплекта БИС ТТЛ с диодами Шоттки. — В кн.: Микроэлектроника и полупроводниковые приборы. — М.: Радио и связь, 1979, вып. 4, с. 56—62.
4. Щетинин Ю. И., Митина В. А. Проектирование и контроль интегральных полупроводниковых ПЗУ с произвольной таблицей истинности. — В кн.: Микроэлектроника и полупроводниковые приборы. — М.: Радио и связь, 1976, вып. 1, с. 40—50.

УДК 621.3.049.77:681.3

## МАТРИЧНЫЕ БИС ДЛЯ СРЕДСТВ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

З. Знаменачек, инженер (ЧССР)  
Я. Штепанек, инженер (ЧССР)  
И. Лакатош, инженер (ЧССР)

При современном развитии технологии полупроводникового производства на одной интегральной схеме можно реализовать большие логические комплексы, в результате возникают принципиальные проблемы межсоединений активных элементов интегральной схемы по требованию пользователя. На этапе проектирования интегральной схемы с низкой степенью интеграции эта проблема решена путем создания «универсальных элементов», которые с учетом избыточности стали широко употребляться пользователями, что оказало влияние на экономичность серийного производства. Однако это решение не эффективно для БИС и СБИС, так как вместе с ростом функциональной сложности схем быстро понижается их универсальность. Проблему можно решить с помощью:

микропроцессоров или программируемых схем, когда ориентация переносится в область программного обеспечения;

заказных интегральных схем, где внутренняя структура определяется требованиями заказчика.

Первый способ требует большого количества дополнительных средних интегральных систем, что быстро снижает эффект от применения БИС и СБИС. Второй способ подходит только для использования схем с большой серийностью производства (минимум 10 тыс. шт. ИС в год), так как при этом расходы, возникшие на этапе проектирования заказной ИС, могут быть включены в цену интегральных схем. В этом отношении полузаказные интегральные схемы — матричные БИС (МаБИС) — наиболее целесообразное решение прежде всего в области вычислительной техники, где нужен большой ассортимент ИС при малой серийности их производства. Ниже рассматривается использование этого решения в развитии новых прогрессивных средств вычислительной техники в Исследовательском институте вычислительной техники в г. Жилина и описание такого элемента.



МаБИС представляют собой универсальный элемент, программируемый фотошаблоном. Программирование осуществляет заказчик, предлагая схемы собственных матриц. Базовый кристалл универсальный (он уже разработан и изготовлен), и при его разработке надо выдержать его конфигурацию, т. е. его морфологическое схемное решение. Проектирование конкретной БИС сводится только к проектированию системы межсоединений, при помощи которой заказчик будет осуществлять коммутацию логических элементов и преобразователей ввода-вывода, которые образованы стандартными операциями на базовом кристалле. Поскольку проектирование БИС в этом случае сходно с проектированием печатных плат, предполагается, что в этой области уже есть средства автоматизированного проектирования. Кроме того, использование МаБИС намного снижает затраты, необходимые на разработку, поскольку при этом требуется только проектирование фотошаблонов на уровне межсоединений, и одновременно позволяет намного сократить сроки отдельных этапов разработки логической схемы.

Следовательно, при помощи МаБИС можно получить такие специализированные ИС, которые до сих пор с экономической точки зрения не могли быть реализованы из-за их малой серийности. Базовый кристалл МаБИС выпускается большими сериями, в результате чего повышается эффективность производства этих схем. Проект реализуется при помощи средств САПР у заказчика, который в зависимости от имеющегося у него технологического оборудования может выполнять и заказные операции.

МаБИС представляют собой перспективный элемент для разработки новых технических средств ВТ, для которых характерно применение большого числа типов ИС. Этим разрешаются противоречия между изготовителем ЭВМ и изготовителем БИС в вопросах ассортимента и серийности производства отдельных типов БИС.

Базовый кристалл — пластина-полуфабрикат, на которой с помощью стандартных операций образована полная вертикальная структура отдельных элементов (коллектор, база, эмиттер, диод, резистор), включая выводы на уровне первого слоя металлизации. Базовый кристалл состоит из нескоммутированных логических элементов входных и выходных преобразователей и «пространства» для реализации межсоединений. Для заказчика базовый кристалл — универсальный элемент.

*Стандартные операции* — все технологические операции, которые осуществляются независимо от заключительной функции ИС. При использовании МаБИС к стандартным операциям относятся массовые операции образования эпитаксиального слоя с помощью процессов диффузии и ионной имплантации для создания контактов «полупроводник — металл» и первого слоя металлизации.

Границей между стандартными и заказными операциями является образование первого слоя металлизации, обеспечивающего возможность долгосрочного хранения этого полуфабриката. Заказчик

разрабатывает: проект логической схемы и его логическое моделирование, тест проверки работоспособности схемы и топологический проект схемы межсоединений.

В зависимости от степени технологического оснащения заказчик может разработать морфологию схемы межсоединений до этапа выдачи данных для управления генератором изображений, а также выполнить у себя окончательные операции на базовом кристалле, который он получает от изготовителя как полуфабрикат.

*Массовые операции* реализуются с помощью фотошаблонов, которые разрабатывает заказчик во время проектирования топологии соединительной металлизации. К массовым операциям относятся: травление первого металлического слоя — 1-й фотошаблон; создание и травление изоляционного слоя — 2-й фотошаблон; образование и травление второго слоя металлизации — 3-й фотошаблон; образование и травление защитного слоя — стандартный фотошаблон.

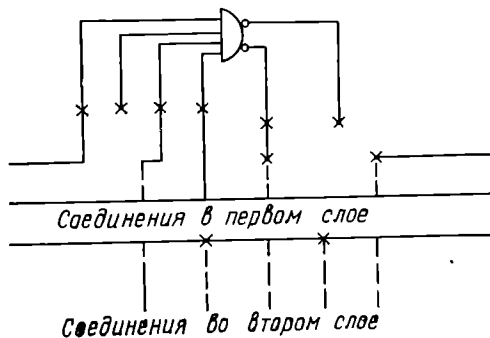


Рис. 1. Топология логического элемента

*Индивидуальные операции* — это контрольно-монтажные операции, выполняемые на каждой пластине (контроль на зондовой установке, резка и ломка пластин, сборка, герметизация и заключительный контроль интегральной схемы). Они реализуются с помощью испытательного теста и монтажной схемы, разработанной заказчиком.

Проектирование топологии (рис. 1) для МаБИС представляет собой проектирование соединения отдельных элементов базового кристалла в символической форме. Основой топологического проекта являются:

- логическая схема или описание межэлементных соединений;
- размещение отдельных элементов базового кристалла и их выводов;
- схема соединительных каналов, созданных в слоях первой и второй металлизации;

- схема запрещенных зон в первом слое металлизации, в которых находятся элементы базового кристалла и разводка питания.

От проектирования морфологии (реальной топологии) зависит действительная структура отдельных функциональных областей каждого элемента и межэлементных соединений в соответствии с нормами заданной технологии (рис. 2). При создании конкретной интегральной схемы специалист не работает над морфологическим проектом, выполненным на этапе проектирования базового кристалла. На основе морфологического проекта базового кристалла реализуются фотошаблоны отдельных технологических слоев. Мор-

фологический проект межэлементных соединений задается фактической шириной промежутков между элементами, размером контактных площадок, формой переходов и т. д., поскольку в топологическом проекте эти данные выражены в символической форме.

Созданные до настоящего времени МаБИС для СМ ЭВМ будут внедряться постепенно:

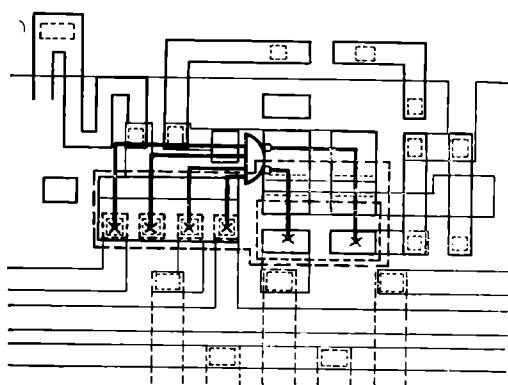


Рис. 2. Морфология (реальная топология) логического элемента

в трех поколениях. Типовыми представителями отдельных поколений являются матрицы с 200, 1000 и 5000 вентилями на базовом кристалле. Каждое новое поколение интегральных схем характеризуется более высоким уровнем технологии. Кроме того, 1-е поколение МаБИС использует изопланарную технологию, 2-е поколение — технологию изопланар-2, 3-е поколение

требует планаризации поверхности кристалла и литографии, позволяющей создавать элементы шириной 2—3 мкм. Технические параметры отдельных поколений интегральных схем приведены ниже.

|  | 1-е поколение | 2-е поколение | 3-е поколение |
|--|---------------|---------------|---------------|
| Количество вентиляей, шт.                    | 200           | 1 000         | 5 000         |
| Задержка, нс                                 | 2,6           | 1,8           | 1,5           |
| Задержка на соединительной линии, нс         | 0,8           | 0,7           | 0,5           |
| Потребляемая мощность ячейки, мВт            | 1             | 0,42          | 0,2           |
| Количество выводов, шт.                      | 38            | 86            | 180           |
| Габариты базового кристалла, мм <sup>2</sup> | 2,8×3,6       | 4,5×5,0       | 7,0×7,0       |
| Напряжение питания, В                        | 5             | 2 ; 5         | 2 ; 5         |

МаБИС 1-го поколения позволяют разработчикам средств вычислительной техники заменять СИС, трудные в поставке, модифицировать эти элементы и создавать большие схемные комплексы, основной частью которых являются структуры, основанные на соединении обычных СИС и ИС. Одновременно при создании МаБИС 1-го поколения будут созданы автоматизированные рабочие места проектировщиков и осуществлена отладка программного обеспечения, а также объединение автоматизированных рабочих мест с технологическими линиями производства интегральных схем и т. д.

Целью 2-го и 3-го поколений МаБИС является создание БИС, работающих самостоятельно, в соединении, а также как логические схемы для СБИС процессорного типа.

В связи с тем, что одной из целей МаБИС 1-го поколения является замена трудных в поставке СИС, интегральная схема должна обеспечивать первичную совместимость и сборку в любой стандартизованный керамический или пластмассовый корпус. Поэтому каждая контактная площадка имеет входной и выходной преобразователи, а схема содержит стабилизатор, снижающий напряжение питания с 5 В до 2—3 В для нужд вентиляционной матрицы.

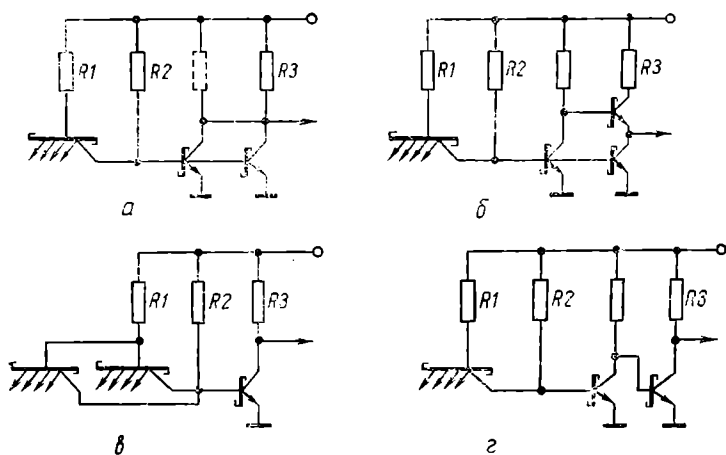


Рис. 3. Схема соединения логических элементов:  
 а — «4И—НЕ» с открытым коллектором; б — «4И—НЕ» с активным выходом; в — «8И—НЕ»; г — «4И»

У МаБИС 2-го и 3-го поколений требования на совместимость с любыми образцами не обязательны, поэтому ИС разработаны с двойным напряжением питания с разделением входно-выходных преобразователей так, что 30% контактных площадок имеют входные преобразователи, 30% — выходные и 40% — входные и выходные преобразователи.

Матрица состоит из логических элементов «4И—НЕ» с открытым коллектором, расположенных так, чтобы соединением двух соседних ячеек можно было создать логический элемент «8И—НЕ» с открытым коллектором или элемент «4И—НЕ» с активным выходом. Другим соединением элементов основной схемы можно получить логический элемент «4И».

Принципиальная схема отдельных соединений приведена на рис. 3. Эти варианты соединения одинаковые у ИС 1-го и 2-го поколений МаБИС. Разрабатывается схема соединений ячейки 5000-вентильного поля. Вентиль 1-го и 2-го поколений МаБИС выполнен в так называемой низкоуровневой ТТЛШ логике, что позволяет получить лучшие динамические свойства, чем у стандартных ТТЛШ. При разработке схемы вентиля 1-го поколения не были известны точные параметры отдельных переходов транзисторов, поэтому была выбрана схема вентиля, содержащая резистор  $R_2$  (см. рис. 3),

поскольку она более устойчива к технологическим разбросам в сравнении со схемой, в которой  $R2$  является составной частью резистора  $R1$ .

При проектировании БИС и СБИС требуется вычислительная техника, еще в большей мере это относится к проектированию МаБИС, где нужный уровень производительности труда в процессе проектирования можно достичь только с помощью средств и методов автоматизированного проектирования, которые предназначены для печатных плат. В Исследовательском институте вычислительной техники (г. Жилина) это проектирование осуществляется при помощи системы ISAN1, которая разработана в институте для печатных плат. Система ISAN1 построена на мини-ЭВМ СМ-4/20 и интерактивном графическом терминале СМ-7405. Большой набор программ, включая программы для автоматизированного размещения и трассировки, которые имеются в системе ISAN1, дают возможность значительно сократить время проектирования.

До начала проектирования конкретных МаБИС формируется рабочий набор данных, который содержит нужные данные о базовом кристалле матрицы, прежде всего информацию о количестве и форме отдельных ячеек, координатах выводов ЛЭ, преобразователях ввода-вывода, запрещенных зонах и т. п. Из этих данных в дисковой памяти образуется рабочий набор данных, с помощью которого на экране дисплея можно получить в графическом виде понятную для человека информацию об основном кристалле матрицы. Этот набор данных образует основу проектируемой МаБИС на базовом кристалле данного типа. Для каждого типа базового кристалла создаются самостоятельные наборы данных.

Последовательность проектирования конкретных МаБИС можно разделить на четыре этапа. На первом этапе рабочий набор дополняется необходимой информацией. Взаимное соединение ячеек матрицы в системе проектирования ISAN1 можно определить двумя способами. По первому способу осуществляется проект соединения элементов логической схемы, которая представлена на рис. 3. Таблица межсоединений после загрузки в ЭВМ дополняется данными рабочего набора файла матрицы.

Программное обеспечение системы ISAN1 дает возможность разработчику образовать логическую схему проектируемой БИС также при помощи интерактивной графики (второй способ) и записать ее в памяти ЭВМ. При этом способе таблица межсоединений образуется прямо из данных, которые описывают логическую схему.

На втором этапе ячейки матрицы «присоединяют» к конкретным логическим элементам, которые образуют проектируемую логическую схему, так как позиции отдельных ячеек матрицы заданы и их невозможно перемещать. Результатом этого этапа является проект оптимального «соединения» ячеек матрицы, при котором обеспечивается минимальная длина соединительных линий.

На третьем этапе при помощи автоматизированных и полуавтоматизированных (диалоговых) режимов работы проектируются от-

дельные слои соединения между схемными элементами проектируемой МаБИС. В результате образуется набор данных, который описывает трассировку межсоединений. В наборе данных закодирована вся информация, необходимая для создания морфологической схемы. Преобразование данных в морфологическую схему и генерирование лент управления для генератора изображений производится в системе проектирования изготовителя интегральных схем.

Задачей четвертого этапа проектирования МаБИС является генерирование образования полных наборов данных для БИС. На этом этапе надо образовать два разных набора данных. Один из них предназначен для логического моделирования, задачей которого является определение ошибок, возникших на этапе разработки ИС. (В НИВТ г. Жилина моделирование осуществляется на мини-ЭВМ СМ-4/20 при помощи программы MRM.) Второй набор данных, описывающих спроектированную ИС, используется для ввода в систему изготовителя БИС, с помощью которой изготавливаются фотошаблоны первого и второго слоя металлизации и фотошаблоны для обработки изоляционного слоя между слоями металлизации.

Небольшой объем статьи не позволяет подробно описать все работы, связанные с проектированием и применением МаБИС. Сфера сотрудничества организаций, принимающих участие в разработке МаБИС 2-го и 3-го поколений, расширится с подключением институтов объединения ЗАВТ.

В дальнейшем предполагается значительное расширение области применения МаБИС, прежде всего за счет внедрения больших логических комплексов на одном базовом кристалле. Это еще больше увеличит роль МаБИС благодаря дальнейшему повышению степени интеграции, уменьшению размеров устройств (что является важным параметром для мини- и микроЭВМ), сокращению потребляемой ими мощности и повышению их надежности.

УДК 621.3.049.77.001.57

#### **СИСТЕМА ДЛЯ ЛОГИЧЕСКОГО МОДЕЛИРОВАНИЯ ЦИФРОВЫХ СХЕМ**

**Р. Динев**, инженер (ИРБ)  
**Кр. Мукарева**, инженер (ИРБ)  
**М. Петров**, инженер (ИРБ)

**Непрерывное повышение степени интеграции привело к тому, что проектирование больших интегральных схем (БИС) без использования вычислительной техники стало невозможным. Созданы сложные комплексы для автоматизированного проектирования БИС. Важное место в этих комплексах занимают системы логического моделирования.**

Рассматриваемая ниже система логического и функционального моделирования МОС ИС HIL (HARDWARE IMAGE LANGUAGE) разработана Институтом микроэлектроники в г. Софии.

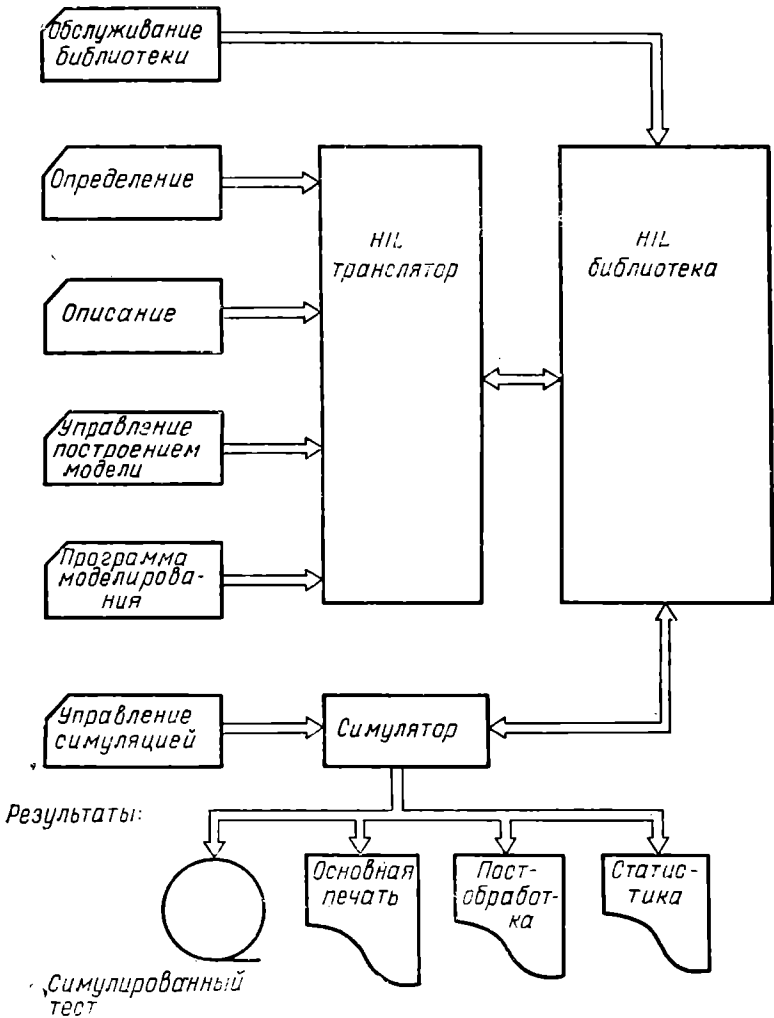


Рис. 1. Общая структурная схема системы HIL

При создании структуры системы преследовалась цель — обеспечить максимальные удобства для пользователя: гибкий и удобный входной язык, оптимальное применение модульного принципа проектирования сверху вниз и снизу вверх, быстрдействие программ, сохранение промежуточных результатов и т. д.

Система спроектирована таким образом, чтобы предоставить инженерам-схемотехникам некоторые средства, используемые про-

граммистами (рис. 1). Эти средства включают в себя: язык для описания и управления моделированием БИС, компилятор, редактор связи для построения модели схемы, симулятор и библиотеку, в которой хранится полная информация процессов.

**Организация базы данных.** Основными критериями при проектировании базы данных системы служили максимально быстрый доступ к отдельным элементам данных, хранение возможно большего количества данных для целей моделирования, для реализации модульного принципа проектирования и освобождения пользователя от проблем, связанных с сопровождением базы данных.

Для операционной системы база данных системы НИЛ представляет собой файл с непосредственным доступом, называемый библиотекой системы HILLIB. Физическая запись в этом файле равна логической и занимает одну дорожку, содержащую 7294 байта.

Для программистов каждая дорожка имеет 18 сегментов данных по 400 байт и «дорожечный» справочник о 18 секторах (рис. 2). Сегмент представляет собой максимальную физическую единицу, которая может быть выделена для данных. Для достижения большей экономии и гибкости при расположении информации каждый файл формируется в системе в виде набора секторов. Каждый сектор может формировать от 1 до 18 сегментов данных на одной дорожке.

При обработке входной информации и при моделировании генерируются и обрабатываются данные, значительно различающиеся по объему и характеру. Элементные данные системы НИЛ структурируются в отдельные секции HILLIB, где они распределяются в один или несколько файлов. Организация библиотеки (рис. 3) — иерархическая, и доступ к ней до уровня элемента набора осуществляется посредством главного справочника, справочника секции и «дорожечного» справочника. Справочники секций, где хранятся файлы с размером, большим размера одного сегмента, содержат информацию о цепочке секторов, необходимых для хранения каждого файла. Такая иерархическая организация не противоречит логическому распределению данных, так как главный справочник и справочники секторов также представляют собой файлы.

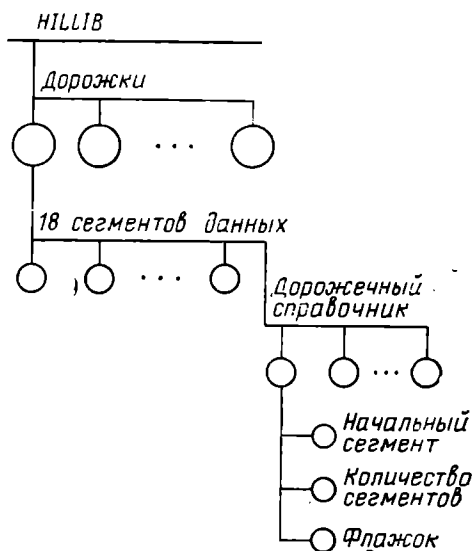


Рис. 2. Физическое распределение библиотеки HILLIB



Основной проблемой при организации библиотеки является потеря пространства памяти при обновлении и стирании файлов. В данном случае эта проблема решена посредством ввода стека с информацией о свободных секторах. Из него берутся секторы для создания всех файлов системы, кроме файла главного справочника, который фиксируется при создании библиотеки. При стирании файлов освобожденные секторы возвращаются в стек. В тех случаях, когда стек переполнен или исчерпан, обслуживающая процедура

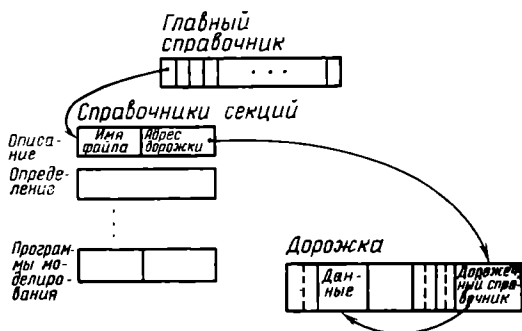


Рис. 3. Иерархическая организация и схема доступа к данным в библиотеке HILLIB

использует специальные поля в дорожечных справочниках. Такой метод работы обеспечивает автоматическое уплотнение и облегчает обслуживание библиотеки.

При создании библиотечного файла в специальную секцию программы записывается и информация о базовых логических элементах. Кроме того, существует набор стан-

дартных программ для записи и вызова их из библиотеки, что облегчает программирование системы.

**Транслятор системы ИЛ.** Для ввода в систему логической структуры ИС и условий ее моделирования создан входной язык [1]. Он состоит из четырех разделов (рис. 4).

Раздел «Описание». В этом разделе указывается структура схемы (составные элементы и связи между ними). Существуют следующие составные элементы:

- вентили — ИЛИ, ИЛИ—НЕ, И, И—НЕ и т. д.;
- специфические МОП элементы — дву- и однонаправленный ключ, тактируемые вентили, монтажное ИЛИ;
- функциональные элементы — набор традиционных и специфических триггеров, ПЗУ, ЗУПВ;
- подсхемы — схемы, которые уже описаны и могут участвовать в новом описании в качестве составного элемента.

Этот раздел работает со следующими операторами:

оператор размещения входов (IP), выходов (OP) и двунаправленных (HP) выводов схемы;

описательный оператор для составного элемента, указывающий его функцию и связи по входу и выходу (такое двойное описание усложняет подготовку элемента, но обеспечивает его эффективную проверку);

оператор задания величины задержки составному элементу (исключением является подсхема, где присвоение относится к ее элементу) или группе элементов схемы.

Описание транслируется в табличную модель схемы и вместе с другой необходимой справочной информацией записывается в определенную секцию HILLIB. Подсхемы в табличной модели не расширяются, а остаются в том виде, как были запрограммированы.

Раздел «Определение» по своему значению является первичным, так как в нем определяются имена, которые могут быть использованы в описании схем. Здесь потребителю предоставляется возможность расширения набора базовых элементов системы с определением элементов следующих типов: BOOL — булевый элемент с восьмью входами и четырьмя выходами (максимум), функция которого задается в виде булевого уравнения. Транслятор генерирует справочную информацию и таблицу истинности и записывает их в библиотеку HILLIB;

RAM, ROM — элементы памяти, описания которых содержат оператор размещения их выводов, а для ROM — содержание отдельных ячеек. Генерируется аналогичная запись в библиотеку HILLIB;

DLY — определение имен задержек. Каждое имя может содержать информацию о значениях восьми типов задержек (номинальная, минимальная, максимальная, вероятная — в пределах  $\Delta_{\min}$ — $\Delta_{\max}$ , задержка передачи, пассивная, по переднему фронту, по заднему фронту) для 16 выводов элемента (максимум). В специальной секции HILLIB записывается заданная для этого имени информация (можно определять значения всех видов задержек).

Раздел «Построение модели». В этом разделе на базе табличной модели раздела «Описание» строится имитационная модель схемы. В ней подсхемы расширены до уровня составных элементов. Их входы и выходы не показаны и их связи с более высоким уровнем — непосредственные. В построенной имитационной модели для каждого элемента схемы существует эквивалент, содержащий информацию о его выходных сигналах и их распространении. Эта модель и ее справочный файл записываются в библиотеку системы.

С целью дальнейшего развития в этом разделе предусмотрена возможность на базе одного и того же описания интегральной схемы создавать различные модели, изменяя уровень детализации некоторых функций в зависимости от целей моделирования. Например, подсхему из описания можно заменить булевым уравнением.

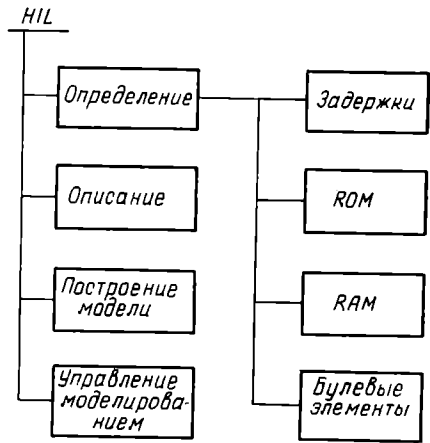


Рис. 4. Входной язык системы HIL

Раздел «Управление моделированием» включает в себя последовательность операторов, называемых «программа для управления моделированием», где пользователь должен:

задать режим моделирования (синхронный/асинхронный);

установить выходы элементов схемы в необходимое состояние 0 или 1 (схема в целом инициализирована с неопределенным состоянием  $x$ );

задать входные воздействия схеме в виде последовательности 0 и 1 и/или в алгоритмическом виде;

включить в случае необходимости некоторые дополнительные возможности, способствующие функциональному анализу логической схемы, верификации и оптимизации проектируемого для этой схемы теста, таких, как слежение за определенными ситуациями в схеме и автоматическое изменение входных воздействий или режимов вывода результатов; обнаружение критических ситуаций и элементов, вызывающих эти ситуации; оценка процесса активизирования схемы; слежение за состоянием активизированных элементов в определенном количестве тактов.

Программа для управления моделированием транслируется в псевдокод, каталогизирующийся в соответствующую секцию библиотеки HILLIB.

**Имитационное моделирование.** Имитационное моделирование данной схемы осуществляется путем последовательной интерпретации каждой инструкции псевдокода. Принцип моделирования, принятый в этой системе, — событийное асинхронное моделирование. Это означает, что реакция схемы на поданное входное воздействие (такт) развивается асинхронно, причем в каждом шаге (микротакте) рассматриваются только те элементы, которые получили изменение входных сигналов. Такт оканчивается при отсутствии элементов с изменением на входе — установлением сигналов в схеме.

Реализованные в схеме моделирующие алгоритмы интерпретируют только среднюю номинальную задержку элементов. Эта задержка имеет следующий смысл: если на вход элемента поступает импульс с продолжительностью, меньшей его задержки, то элемент на него не реагирует (поглощение импульса), но если продолжительность входного импульса больше задержки, то элемент среагирует через промежуток времени, равный задержке.

При синхронном режиме работы модели всем элементам устанавливается значение задержки, равное единице. Значения элементов, которые могут быть прослежены, — 0, 1,  $X$  и  $Z$  доступны только в конце такта. В асинхронном режиме работы модели элементам присваиваются значения их задержек в интервале от 0 до 63. Здесь, кроме упомянутых устойчивых состояний, можно следить за следующими переходными значениями:  $U$  — планируемая единица;  $D$  — планируемый нуль;  $?$  — планируемое  $X$ ;  $T$  — планируемое  $Z$ . Эти значения появляются в пределах такта. За значениями по микротактам организован режим слежения.

Система спроектирована на Ассемблере и ПЛ/2 и может работать в пакетном режиме под управлением ДОС на ЕС ЭВМ, имею-

щих дисковые накопители 29 Мбайт и оперативной памятью минимум 256 Кбайт:

Возможности системы логического моделирования НІЛ:

моделирует интегральные схемы с произвольной логикой, включающие в себя до 7 тыс. логических элементов;

использует гибкий и удобный входной язык;

отличается сравнительно высоким быстродействием;

располагает дополнительными средствами, способствующими анализу и верификации как моделируемых ИС, так и создаваемых для них тестов;

позволяет вводить новые средства для подробного анализа проектируемой ИС.

Система НІЛ используется Институтом микроэлектроники (г. София) с 1981 г. для моделирования схем с произвольной логикой и оптимизации их тестов.

#### ЛИТЕРАТУРА

1. Георгнева А., Фильов К., Мукарева Кр., Динев Р. НІЛ — язык для описания и управления моделированием цифровых схем. — Микроэлектроника и полупроводниковые приборы, 8—10 окт. 1981 г., Ботевград.
2. Динев Р. Организация базы данных в системе НІЛ. — Микроэлектроника и полупроводниковые приборы, 8—10 окт. 1981 г., Ботевград.

### III

## Программное обеспечение ЭВМ

---

УДК 681.3.06

#### ИСПОЛЬЗОВАНИЕ СТРУКТУРНОГО ПРОГРАММИРОВАНИЯ ПРИ ПОДГОТОВКЕ ПРОГРАММ ДЛЯ АВТОМАТИЗИРОВАННЫХ РАБОЧИХ МЕСТ ПРОЕКТИРОВЩИКА, ПОСТРОЕННЫХ НА БАЗЕ СМ-3

Г. Ротхард, инженер (ГДР)

С 1980 г. на народном предприятии «Карл-Цейс-Йена» эксплуатируется опытный вариант автоматизированного рабочего места механика АРМ-М (рис. 1), в конфигурацию которого входит малая ЭВМ СМ-3 с дисковой операционной системой и графические устройства ввода-вывода данных.

При разработке автоматизированного рабочего места АРМ-М была поставлена цель — полностью использовать преимущества метода структурного программирования, созданного с учетом известных теорий [1], [2], [3], и языка программирования Фортран. Этот подход можно разъяснить на основе трех методических принципов.

Первый принцип — проблемный анализ со специфическими средствами отображения (как самостоятельный этап). Связь между замыслом конструктора (графико-описательные инструкции по конструированию) и программистом (Фортран, ОС ЕС) производится через проблемных аналитиков, знающих вычислительную технику. Они фиксируют свои результаты при помощи средств отображения (структурограмма [3], таблица решений [4] и таблица использования диалога [5]), которые, с одной стороны, могут быть проверены конструктором, а, с другой стороны, могут служить программисту «алгоритмом» для практической реализации.

Благодаря этому принципу можно установить контакты в многочисленном, разнородном по составу коллективе разработчиков. Проблемное решение не смешивается с реализацией на ЭВМ, а полученный алгоритм обладает «удобной структурой», т. е. в нем используются типовые компоненты (сохраняются основные структуры в соответствии с рис. 2). Построенная по этому алгоритму программа на Фортране становится также программой с удобной структурой. Одновременно возникает несколько проблем. Чем больше программных возможностей имеет аналитик, тем больше у него искушение сформировать текст на Фортране, так как ему кажется, что он охватывает все стороны проблемы. Однако Фортран не очень удобен для применения в структурном программировании, поэтому

возникают нарушения соответствующих соглашений, вследствие чего иногда нельзя восстановить структурограмму, составленную при помощи текста, написанного на исходном языке.

Второй принцип — иерархическое разбиение проблемы. Проблема анализируется сверху вниз (пошаговое уточнение проблемы — рис. 3), а программирование и отладка программы ведутся

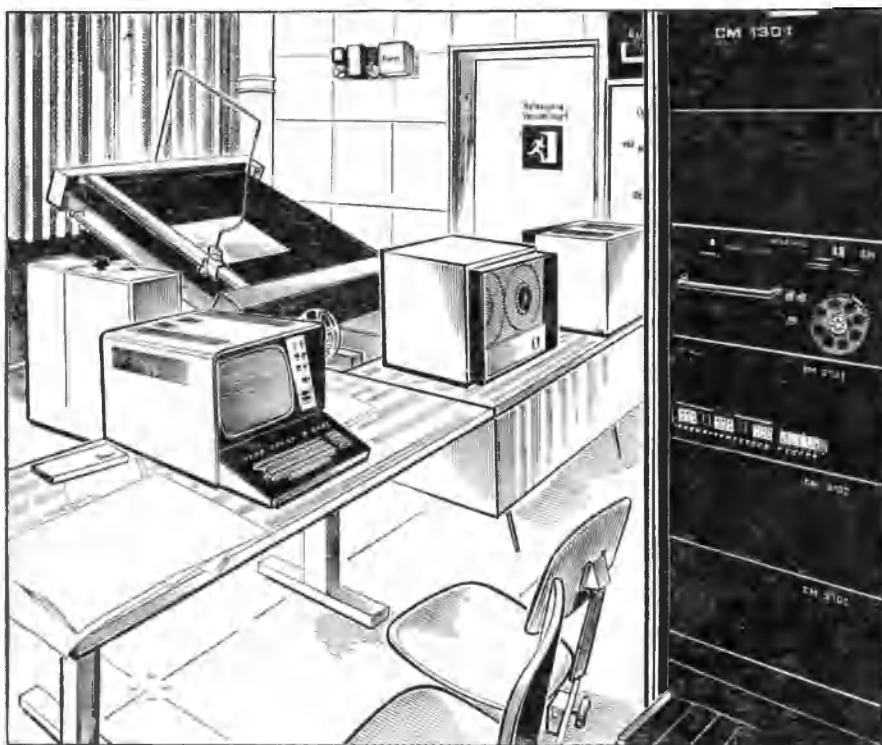


Рис. 1. Автоматическое рабочее место механика АРМ-М

снизу вверх (парасивание на базе основных модулей). При анализе по методу сверху вниз структурограмма является весьма удобным средством представления, обеспечивающим наглядность очередности функциональных блоков, без перекрытия, и их размещение в одном алгоритме, а также взаимосвязь между упрощенным и уточненным вариантами. Помимо этого можно использовать таблицы разрешения для данного вида представления.

Программирование по методу «снизу вверх» предусматривает использование алгоритмов, составленных на основе проблемы, или имеющихся модулей базового программного обеспечения. Написание программы и ее отладка проводится поэтапно. Этот принцип увеличивает наглядность применяемых методов решения, создает

предпосылки для распределения задач и временного перекрытия подзадач и эффективно поддерживает тестирование. Однако при формировании алгоритмов ускоряется износ структурограммы как следствие получения новых сведений из тестирования. Проблема решается с помощью вспомогательного программного обеспечения, описанного ниже.

Разбиение текста программы поддерживается при использовании Фортрана, в первую очередь, техникой формирования подпро-


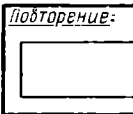
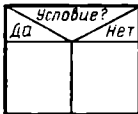
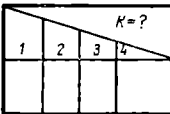
|                    | Цель  | Петля   | Разбор случаев   |  |
|--------------------|---|---|--|--|
| Структурограмма    |  |  |   |  |
|                    |   |   |   |  |
| Пример на Фортране | <pre>Z = -P/2.<br/>W = SORT(Z**2-D)<br/>X1 = Z+W</pre>                            | <pre>DO 1 I=1,4<br/>  X(I)=X(I)+DX<br/>  Y(I)=Y(I)+DY</pre>                       | <pre>IF (A) 2.1.2<br/>1 CALL<br/>  GOTO 3<br/>2 CALL UP2<br/>3 ...</pre>   |  |
|                    |   |   | <pre>GOTO(5,6,7,8).K<br/>5 CALL P1<br/>  GOTO 9<br/>6 CALL P2<br/>  GOTO 9<br/>7 CALL P3<br/>  GOTO 9<br/>8 CALL P4<br/>9 CONTINUE</pre> |  |

Рис. 2. Основная структура структурного программирования

грамм. Так как разбиение на подпрограммы приводит к уменьшению эффективности разработки программы (из-за реализации пересылки данных), то некоторые программисты стараются сформировать подпрограммы объемом в сотню строк. Данную проблему можно в какой-то мере решить, используя для иерархического разбиения алгоритма комментирующие строки.

Третий принцип — программная документация со специфическими средствами представления и вспомогательным программным обеспечением. Если программа вводится в эксплуатацию, то необходимо иметь представление о том, каков срок жизни данной программы и как она сопровождается на этот срок. При этом устраняются имеющиеся в программе ошибки, выполняются согласования (при изменении условий эксплуатации и увеличении производительности).

В данном случае качеству (и больше всего прозрачности) программной документации уделяется особое внимание, так как первичный коллектив разработчиков редко работает вместе в течение длительного времени. Качество документации проявляется прежде всего в подробном и выразительном построении текста на исходном языке при помощи комментирующих строк (минимум: в каждой

подпрограмме словесное описание назначения занимает 2—3 строки в заглавии). Желательно также иметь описание назначения для вторичных функциональных блоков в подпрограмме, описание вход-

СТАРУК -  
(управляющая программа)

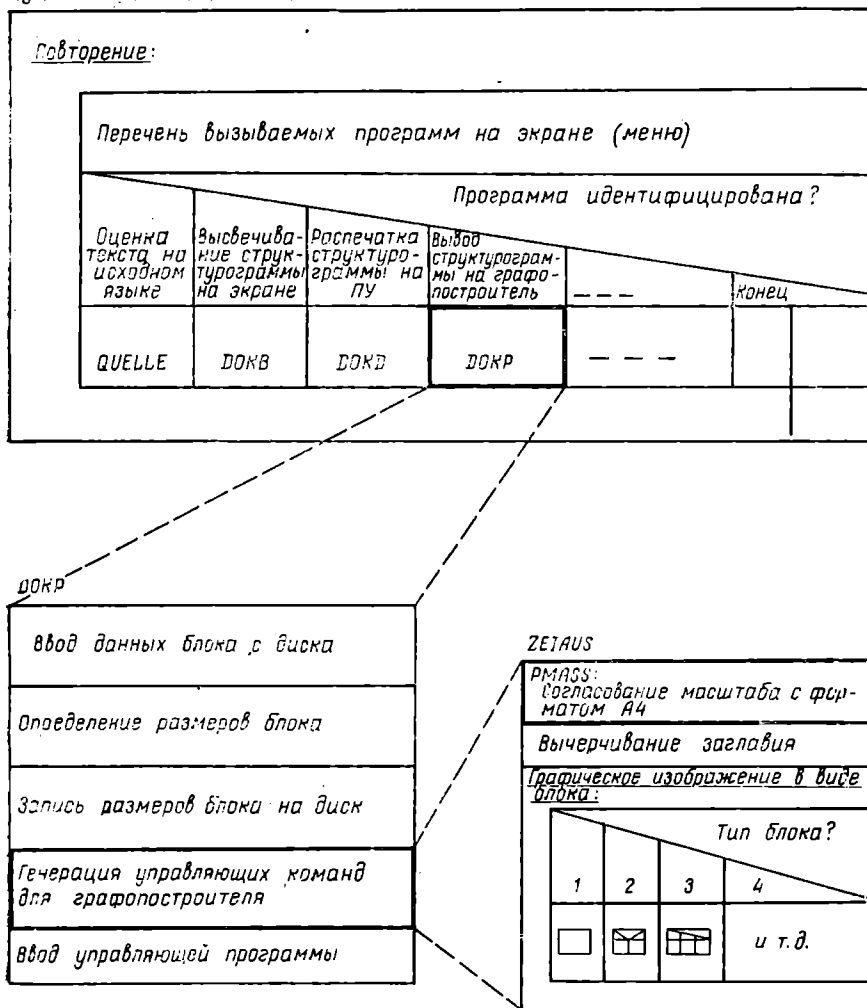


Рис. 3. Оценка по методу Top-Down

ных и выходных данных, специальных кодов и т. д. Кроме того, требуется актуальный алгоритм, записанный для любой подпрограммы в виде структурограмм с легко распознаваемым соответствием с исходным текстом (например, одно и то же обозначение функционального блока, занесение маркеров Фортрана в структу-



программу). Запись должна быть сделана на бланке с приложениями для каждой подпрограммы (сведения об используемых библиотеках, блоках COMMON, вторичных подпрограммах, периферийных устройствах, переменных и др.).

Сопровождение программ требует от автора в течение длительного времени малых затрат на доработку, устранение ошибок и другие изменения в программах; программы приобретают большой срок жизни независимо от автора. Проблемно-ориентированные алгоритмы доступны для других пользователей, у которых установлены иные технические и программные средства. Однако составление программной документации — работа не творческая, поэтому программисты выполняют ее неохотно. Вследствие этого необходимо сформулировать конкретные требования к форме и объему документации, если программа должна применяться в течение длительного срока.

**Вспомогательное программное обеспечение для структурного программирования.** В качестве дополнения к методическим вспомогательным средствам, описанным выше, разработана система программ STRUKT, используемая для автоматического составления блок-схем в виде структурограмм и на основе текста на Фортране. Она обеспечивает непрерывное графическое отображение алгоритма в процессе разработки программ (главным образом на этапе отладки), уменьшает затраты на составление программной документации (нет графических работ), предусматривает широкое применение методов структурного программирования (функции автоматической диагностики). Система STRUKT реализована на АРМ-М. В ней используются Фортран IV, операционная система ДОС для СМ-3, а также программное обеспечение, разработанное для графических изображений и ведения диалога. Система STRUKT работает следующим образом (см. рис. 3). В диалоговом режиме посредством алфавитно-цифрового дисплея запускается программа анализа исходных текстов QUELLE. Она требует идентификации текста на Фортране после введения имени файла и выдает через несколько секунд либо сообщение типа

ПОЗДРАВЛЯЮ!

ИСХОДНЫЙ ТЕКСТ НА ФОРТРАНЕ ... СООТВЕТСТВУЕТ ПРАВИЛАМ СТРУКТУРНОГО ПРОГРАММИРОВАНИЯ И ПОЗВОЛЯЕТ АВТОМАТИЧЕСКИЙ ВЫВОД СТРУКТУРОГРАММЫ

или несколько диагностических сообщений при нарушении установленных системой STRUKT правил (например, возврат, который выполняется не с целью условного повторения). Если нарушений нет, управляющая программа предоставляет возможность использовать несколько программ для вывода результата:

структурограммы на экране дисплея, печатающем устройстве или графопостроителе;

текста на Фортране по выбору с дополнительным обозначением структуры блока;

контрольных распечаток информационных блоков, обрабатываемых ЭВМ.

Имеющаяся в данный момент версия программы STRUKT генерирует графическую часть структурограммы с занесением маркеров операторов. Вручную заносятся шаги и цели решения проблемы. Сплошное повторение исходного текста на данном этапе нежелательно. В будущей версии программы STRUKT предусматривается автоматическая запись содержания комментирующих строк в автоматически вычерченные структурограммы.

**Опыт внедрения новых методов.** Описанные методы структурного программирования осваивались в три этапа:

переквалификация всех сотрудников;

определение единых методов и эффективных форм работы (направление для собственной работы);

разработка вспомогательных средств программного обеспечения STRUKT.

Общеизвестно, что новые методы работы не пробивают сами себе дорогу: самым главным препятствием, часто неосозанным, является необходимость переучивания. Поэтому предубеждение у некоторых старых специалистов подчас больше, чем у новичков (у первых уже выработались свои любимые методические навыки), и у программистов выше, чем у проблемных аналитиков.

Введению структурного программирования способствуют следующие мотивы:

если проблемный аналитик представляет свои алгоритмы в удобной для него форме (структурограммы, таблицы решения), то программисты, на которых возлагается задача дальнейшего уточнения и реализации алгоритмов на ЭВМ, вынуждены понимать и применять данные формы представления;

если имеющиеся в коллективе авторитетные личности признают новые методы, то они активно участвуют в использовании и совершенствовании этих методов, которые для них отождествляются с намеченной целью, и их не останавливают вопросы формы;

если есть время, то нужно проводить взаимный анализ проделанной программистами работы. Это позволит избежать ошибок, вытекающих из неправильного понимания или неэффективного использования методов.

По нашему опыту сотрудники после переквалификации сознательно применяют методы структурного программирования, но не владеют всеми его принципами в одинаковой степени. Например, структурограммы — это знакомые средства, поэтому используются часто, хотя и понимаются иногда как синоним структурного программирования, метод сверху вниз применяется реже, но все-таки есть стремление составить алгоритм и программу в виде одного блока, запись текста на исходном языке проводится сотрудниками часто по-разному.

В настоящее время специалистам предоставляются наряду с методическим материалом «Технология разработки программного

обеспечения для автоматизированных рабочих мест» дополнительные средства программного обеспечения.

#### ЛИТЕРАТУРА

1. Dijkstra E. **The GOTO-Statement considered harmful.** Communications of the ACM 11 (1968) 3, s. 147—148.
2. Gewalt K., Haake G., Pädler W. **Software Engineering: Grundlagen und Technik rationeller Programmentwicklung.** Oldenbourg-Verlag. München, Wien 1977.
3. Schnupp P. **Struktogramme — eine neue Methode der Systemplanung.** On line 12 (1974) 11, s. 736—743.
4. Freitag G. et al. **Einführung in die Entscheidungstabellentechnik.** Verlag Technik, Berlin, 1976.
5. Rothhardt G. **Tabellenmethode zur Dokumentation des Mensch-Maschine-Dialogs.** Rechen technik/datenverarbeitung 16 (1979) 10, s. 14.

УДК 681.3.06

#### ПАКЕТ ПРОГРАММ ОБРАБОТКИ ТЕКСТОВОЙ ИНФОРМАЦИИ

О. А. Яворский, инженер (СССР)  
В. Н. Чернявский, инженер (СССР)

С появлением недорогих и надежных малых ЭВМ и многочисленной номенклатуры устройств ввода, обработки и вывода информации малые ЭВМ стали интенсивно применяться для обработки текстовой информации.

Можно выделить два основных направления в использовании малых ЭВМ для обработки текстов. Одно из них предусматривает создание специализированных программно-технических комплексов, учитывающих особенности обработки текстов. Такие комплексы имеют в своем составе видеотерминалы с функциональной клавиатурой, специализированные операционные системы и прикладные программы, ориентированные на решение задач обработки текстов.

Другое направление состоит в разработке системы автоматизации обработки текстов для пользователей, имеющих серийные ЭВМ. Эти системы должны включать средства настройки на типы используемых терминалов и печатающих устройств, средства изменения языка общения с системой (настройки его на язык пользователя), обучающие средства, которые облегчают освоение системы пользователем-непрограммистом. Примером такой системы является пакет программ обработки текстовой информации ТЕКСТ, предназначенный для автоматизации подготовки текстовых документов и ведения делопроизводства на серийно выпускаемых малых ЭВМ типа СМ-4 под управлением операционной системы ДИАМС-2.

Пакет программ предоставляет пользователю возможность: создания файлов текстовой информации; внесения изменений в файлы (редактирование информации);

форматирования и вывода текстовых документов на печатающее устройство;

поиска данных в соответствии с задаваемыми условиями поиска; организации архивов файлов и поиска в архивах;

организации «почтовых ящиков» пользователя, пересылки в них информации и чтения информации пользователем;

защиты данных от несанкционированного доступа и восстановление файлов в аварийных ситуациях;

настройки пакета на типы применяемых терминалов;

использования сервиса при вводе и редактировании текстов (макетный ввод текстов, пересылка текстов из других файлов, автоматические переносы слов в конце строки, различные режимы редактирования);

общения с пакетом на диалоговом языке, ориентированном на пользователя-непрограммиста;

перевода сообщений и операторов входного языка с русского языка на язык пользователя;

вывода поясняющих сообщений об основных операторах пакета, облегчающих освоение входного языка.

**Входной язык пакета.** Входной язык пакета состоит из набора операторов. Структура оператора имеет вид:

<ИМЯ ОПЕРАТОРА>      <ИМЯ ФАЙЛА>

где <имя оператора> — последовательность символов, различающихся по первым трем символам, а <имя файла> — имя файла пользователя.

В состав языка входят операторы управления пакетом, управления файлами, ввода и редактирования, поиска и вывода данных, архивизации файлов, вывода текстовых документов, форматирования текстовых файлов.

После ввода оператор переходит в командный или диалоговый режим работы. В командном режиме выполнение различных функций инициализируется командами с клавиатуры терминала. В диалоговом режиме оператор выводит на экран терминала вопросы, на которые необходимо дать ответ. Освоение операторов входного языка в пакете ускоряется средствами обучения. По запросу пользователя на экран терминала выводится сообщение о функциях оператора и примерах работы с ним.

**Структура файлов пользователя.** Пакет программ ТЕКСТ предусматривает два типа логической структуры файлов пользователя. Для хранения текстовой информации, преобразуемой в результате обработки в текстовый документ, предназначен текстовый файл, основными структурными элементами которого являются строка, страница и раздел. Строка — последовательность символов. Количество символов в строке не может быть более 132. Страница — последовательность строк. Длина страницы (количество строк) задается пользователем. Раздел — последовательность страниц, начинающаяся с новой страницы. При вводе текста выполняется автоматическое формирование строки и страницы.

Файл документов используется для хранения текстов делопроизводственных документов и характеристик документов. Он состоит из записей, каждая из которых может включать однострочные и многострочные поля (именуемые единицы данных), имеющие самостоятельное смысловое значение. Однострочное поле — это последовательность символов длиной не более 255 символов. Многострочное поле состоит из последовательности строк длиной не более 132 символов. Количество строк многострочного поля не ограничено. Такая структура файла документов позволяет хранить не только тексты документов (многострочные поля), но и некоторые характеристики документа (дата, шифр документа и т. д.) с помощью журнала регистрации документов, который поступает вместе с самими документами.

В состав каждого файла входит также описание файла, различное для текстового файла и файла документов. Описание содержит набор параметров, необходимых для работы операторов пакета. Такими параметрами для текстового файла являются длина строки текста, размеры листа документа и его полей, величина абзаца. В описание файла документов входит перечень имен полей и их характеристик, имя ключевого поля, задание которого ускоряет поиск записей.

**Ввод и редактирование информации в файлах.** Ввод описаний файлов, выполняемый при создании файла, производится с помощью оператора ввода в режиме диалога. После ввода описание может быть отредактировано в любой момент работы с файлом. Текстовый файл вводится с помощью оператора редактирования, включающего широкий набор команд ввода и редактирования текстов. При вводе текста могут использоваться следующие сервисные возможности:

макетирование строки. Макет строки может быть задан пользователем командой ввода макета. Он содержит перечень номеров позиций (начал полей строки), на которые необходимо выполнить отступ от начала строки или в которых необходимо размещать фрагменты текстов, например элементы таблицы. Макет позволяет автоматически продвинуться на нужную позицию строки с помощью клавиш табуляции, изменить длину строки, получая тексты разной ширины, вставлять в текст пустые места для последующего заполнения. Структура, заданная макетом, сохраняется при редактировании;

автоматическое форматирование строк. В пакете ТЕКСТ реализованы три режима форматирования — перенос слова, не помещающегося в конце строки, на другую строку с заполнением пустого места передвижением слов в строке, перенос части слова разбиением его на слоги по правилам русского языка, полуавтоматический перенос, состоящий в задании пользователем точки разделения слова с помощью курсора;

вставка фрагментов текстов из других текстовых файлов и из сформированной части создаваемого файла;

заполнение страниц, содержащих тексты заглавной и заключительной записи листов документа, т. е. постоянных для документа текстов, размещаемых в верхней и нижней части листа;

установка начала нового раздела;

поиск фрагментов текста и редактирование. Поиск в текстовом файле выполняется по координатам текста (страница, строка, символ) и по заданному контекстному образцу, содержащему начало и конец текста или начало и конец текста, который необходимо найти. Поиск также может выполняться последовательным просмотром текстового файла в прямом и обратном направлениях. Редактирование текста состоит в удалении, вставке или замене фрагментов текста. В пакете ТЕКСТ реализовано командное и экранное редактирование. В команде редактирования задаются координатное или контекстное поисковое предписание и операции редактирования. Экранный (курсорный) режим редактирования состоит в перемещении курсора терминала в позицию, с которой изменяется текст, и выполнении редактирования с помощью клавиатуры терминала. В связи с тем, что при редактировании изменяется длина строк, после его завершения выполняется автоматическое переформирование строк в соответствии с заданной длиной строки и структурой макета.

Файл документов формируется оператором ввода. Исходные записи вводятся с терминала или внешних устройств. Допускается ввод значений полей записи как в последовательности, определенной описанием, так и в любой другой последовательности, а также заполнение записей файла по частям. При вводе многострочных полей с терминала используются возможности, предоставляемые оператором редактирования (макетный ввод, формирование строки и др.). При вводе информации с внешних устройств такие возможности не используются. Изменения в записи файла документов вносятся с помощью оператора редактирования.

**Поиск в файле документов** выполняется с помощью оператора поиска и выдачи данных. Задание поискового запроса предусмотрено в режиме диалога. В начале запроса задается перечень имен полей записи, которые необходимо вывести на экран терминала и (или) печатающее устройство. При отсутствии перечня имен полей выводятся значения полей всей записи. Условия поиска представляют собой последовательность строк. Каждая строка состоит из имени поля, на значения которого накладываются условия, и перечня условий вида:

|    |               |    |               |
|----|---------------|----|---------------|
| С  | значение поля | ПО | значение поля |
| С  | значение поля |    |               |
| ПО | значение поля |    |               |
| =  | значение поля |    |               |
| ?  | текст         |    |               |

Первые три условия задают интервал значений — нижнюю и верхнюю границы поиска. Последнее условие определяет поиск по неполному значению поля. Так, если значение поля есть «ГОРОД

МОСКВА, УЛИЦА...» и условие поиска задано как «? МОСКВА», то запись, содержащая приведенное выше значение поля, считается соответствующей условию поиска.

Условия поиска в строке предполагаются связанными между собой логическими ИЛИ, строки — логическим И. Условия, налагаемые на значение поля, могут описываться в нескольких строках. При отсутствии условий поиска выводятся все записи файла.

Если в файле документов не найдены записи, соответствующие поисковому запросу, пользователь может дать задание продолжить поиск в архиве документов, а также задать время поиска, по истечению которого поиск будет приостановлен.

**Форматирование и вывод текстового документа.** Текстовый файл может быть выведен на печатающее устройство в виде текстового документа. Для этого предварительно выполняется операция форматирования, заключающаяся в разделении текста на страницы. Оператор печати выводит на печатающее устройство заглавную и заключительную записи листа документа, текст листа и номер страницы. Предусматривается возможность вывода через страницу для двусторонней печати листа, печать двух страниц с одним номером и другие виды печати.

**Архивизация файлов.** В состав пакета включен оператор архивизации, выполняющий пересылку файла документов или отдельных его записей в архив на магнитных лентах. При необходимости файл документов может быть извлечен из архива для поиска данных. Текстовый файл пересылается в архив полностью.

**Почтовые ящики пользователя.** Пакет программ ТЕКСТ предусматривает возможность пересылки информации из файла документов в рабочие файлы пользователя (почтовые ящики). Адреса почтовых ящиков задаются при вводе записей в файл документов. Пересылаемая информация хранится в почтовых ящиках и может быть прочитана пользователем с помощью оператора печати. Предусмотрен контроль заполнения почтовых ящиков.

**Защита файлов.** Пакет программ ТЕКСТ устанавливает три вида пользователей: администратор пакета, администратор файлов, имеющих один код идентификации пользователя (КИП), и пользователь-прикладник. Администратор пакета управляет настройкой пакета. Администратор КИП регистрирует пользователей, разрешает или закрывает им доступ к файлам. Пользователь-прикладник может вести обработку файлов, которые сформированы этим пользователем или доступ к которым разрешен ему администратором КИП.

**Сохранность файлов.** Сохранность файлов пользователя обеспечивается как средствами пакета, так и средствами операционной системы ДИАМС-2. Этой цели служит системный журнал, в котором регистрируются все изменения информации в дисковой памяти. В аварийных ситуациях по копии дисков и системному журналу выполняется восстановление дисковой памяти. Кроме того, в случае частичного разрушения файла пользователя средствами пакета может быть восстановлена структура файла.

**Настройка пакета.** В пакете ТЕКСТ, функционирование которого должно выполняться на серийных ЭВМ типа СМ-4, предусмотрены средства настройки пакета. В процессе настройки пользователь может:

задать параметры терминалов, применяемых для работы с пакетом;

исключить операторы входного языка, в которых нет необходимости;

включить в состав языка пакета новые операторы, программы реализации которых пишутся на языке операционной системы ДИАМС-2 с учетом системных требований;

изменить сообщения, выдаваемые пакетом в процессе работы пакета. Могут изменяться как текст сообщения, так и язык, на котором оно выводится. Таким же изменениям могут подвергнуться и имена операторов входного языка, включая все служебные слова, из которых они состоят. Это позволяет настроить пакет на язык пользователя, использующий символы кода КОИ-7.

Пакет ТЕКСТ включает также ряд возможностей по управлению файлами: объединение и удаление файлов, вывод на печать служебных и пользовательских каталогов, а также файлов документов.

Описанные функции пакета ТЕКСТ делают возможным его широкое применение в проектно-конструкторских организациях, канцеляриях, различных учреждениях для подготовки текстовых документов и ведения делопроизводства: учета входящей и исходящей документации, пересылки ее исполнителям, контроля исполнения, поиска и архивизации.

Пакет программ ТЕКСТ прошел ведомственные испытания и рекомендован для поставок пользователям. Пользователям он поставляется на магнитном носителе (диске или ленте) с комплектом эксплуатационной документации, обеспечивающей его про верку и выполнение всех функций пакета.

УДК 681.3.06

### **ПАКЕТ ПРОГРАММ ТЕЛЕОБРАБОТКИ ДАННЫХ НА ЕС ЭВМ**

**И. Асталаш**, инженер (ВНР)

**Я. Конц**, инженер (ВНР)

**А. Шураньи**, инженер (ВНР)

В последнее десятилетие в Венгрии стала широко использоваться телеобработка данных на машинах ЕС ЭВМ. Различным требованиям телеобработки отвечает пакет программ \*, который содержит:

монитор телеобработки ШЕДОУ II (SHADOW II);

генератор программ телеобработки ЦПГ (CPG);

---

\* Пакет программ закуплен у английской фирмы ALTERGO с правом распространения в других социалистических странах.



диалоговую систему разработки программ ГУТС (GUTS).

Компоненты пакета можно использовать как самостоятельно, так и одновременно. С помощью монитора телеобработки ШЕДОУ II (сокращенно монитор) можно создавать системы, которые решают задачи в реальном масштабе времени. ЦПГ служит для создания транзакционных программ, для оптимизации их работы и упрощения программирования. Система ГУТС обеспечивает разработку программ диалогового режима, подготовку и работу интерактивных программ как на местных ЭВМ, так и через удаленные конечные абонентские пункты.

## Монитор телеобработки ШЕДОУ II

ШЕДОУ II относится к той функциональной группе программных продуктов для ЕС ЭВМ, к которой принадлежит система КАМА.

**Среда функционирования ШЕДОУ II.** Технические средства: центральное устройство — вычислительная машина ЕС ЭВМ-1 и ЕС ЭВМ-2;

минимальная реальная емкость памяти при использовании ШЕДОУ II — 512 Кбайт (собственно для пакета нужно 100 Кбайт);

внешняя память — сама система ШЕДОУ II умещается на одном дисковом пакете типа ЕС-5056; кроме того, поддерживает устройства типа ЕС-5052 и ЕС-5066;

конечные абонентские пункты — любое асинхронное или синхронное устройство мониторного типа или пишущая машинка.

Программные средства:

операционная система — ДОС, ДОС ВС, ДОС ВСЕ, ОС МФТ, ОС МВТ, ВС1, ВС2;

поддерживаемые программные языки — Кобол, ПЛ/1, Ассемблер, ЦПГ;

поддерживаемые системы обработки банка данных — ИДМС, ДЛ/1, БОМП, ДБОМП, ЦФМС, ИСОГЕН, ТОТАЛ, РАМИС.

**Особенности функционирования ШЕДОУ II.** ШЕДОУ II, функционируя в промежуточном слое между пользовательскими программами и операционной системой, расширяет управляющие функции и возможности операционной системы для работы банка данных в режиме on-line. При поступлении запроса от абонента ШЕДОУ II выбирает соответствующую пользовательскую программу и передает ей управление. Далее эта программа ведет диалог с абонентом и может пользоваться услугами ШЕДОУ II. Услугами операционной системы можно пользоваться также после выдачи соответствующих команд ШЕДОУ II.

*Обращение с программами:* претрансляторы осуществляют разбивку команд ШЕДОУ II пользовательских программ, написанных на языке Кобол, ПЛ/1 или Ассемблер. После этого трансляторы должны прогоняться обычным образом. Поскольку в среде on-line

целесообразно писать небольшие (приблизительно 5 Кбайт) пользовательские программы, в системе ШЕДОУ II имеются команды для связки программ. Тестированию программ помогают симулятор и динамическое сопровождение, активизируемое с абонентских пунктов (без изменения исходной программы). ШЕДОУ II препятствует тому, чтобы вследствие какой-либо ошибки в пользовательской программе остановился прогон всей секции (ввиду того, что пользовательские программы ШЕДОУ II и монитора прогоняются в общей секции).

*Работа с терминалом и каналом.* Ввод-вывод на абонентском пункте происходит на уровне GET/PUT, в связи с чем для программиста скрыта сложность обращения с сетью конечных абонентов. Исправление ошибок сети осуществляет монитор ШЕДОУ II. Возможность сопровождения передачи данных по каналу, активизируемая с операторской консоли или с абонентского пункта, во многих случаях делает излишним технический контроль ошибок. Экранные форматы пользовательских программ могут планироваться, изменяться и тестироваться в диалоге независимо от программы.

*Работа с памятью.* Предусмотрена динамическая загрузка и освождение как оперативной, так и внешней памяти. Большое внимание уделено разработке эффективных механизмов использования памяти с целью удовлетворения потребностей пользователей, работающих с небольшими объемами памяти.

*Работа с базой и массивом данных.* В любом режиме обращения к стандартным массивам данных ЕС ЭВМ в распоряжении программиста имеются команды ШЕДОУ II. Монитор автоматически защищает массив данных от обращения нескольких пользователей одновременно к одной и той же записи. Нет ограничений для того, чтобы в одной пользовательской программе монитора ШЕДОУ II были дополнения, вызывающие средства какой-либо системы управления базой данных, так как ШЕДОУ II обеспечивает связь с ранее перечисленными системами.

*Автоматическая регистрация и запуск после ошибки.* В зависимости от заданных параметров монитора ШЕДОУ II на внешней памяти регистрируется состояние каждой записи любого массива данных до проведения их модификации. Так, в случае ненормального окончания программы, обслуживающей массив данных, монитор автоматически и незамедлительно может восстановить то состояние массива данных, которое существовало до прогона программы, имеющей ошибку. Состояние записей после модификации также автоматически фиксируется. С помощью такой регистрации и при наличии ранее записанного массива можно восстановить актуальное состояние массива данных в случае, если изменилось последнее состояние массива из-за физической ошибки в данных, их потери или перезаписи. Новый запуск программ после исправления ошибки в простых случаях осуществляется автоматически, в других — программируется.

Установка системы может быть осуществлена за 4—5 ч. Пере- трансляция системы вследствие изменения ее конфигурации, на- стройки и т. д. требует 15—30 мин прогона.

При использовании монитора ШЕДОУ II на машинах ЕС ЭВМ-I с основной оперативной памятью 0,5—1 Мбайт система работает хорошо, если имеется сеть абонентских пунктов небольшого раз- мсра, большей частью состоящая из локальных экранных терми- налов. Из-за сложности работы с каналом связи при обслужива- нии удаленных экранных абонентских пунктов может повышаться нагрузка на центральное устройство, в результате чего замедляет- ся прохождение задания или увеличивается время ответа монитора ШЕДОУ II. Абонентские пункты с асинхронными пишущими ма- шинками гораздо меньше загружают центральный процессор (ЦП) из-за последовательного характера работы, но пользователи во многих случаях считают работу с ними неудобной. Распространяе- мый вместе с системой ШЕДОУ II ППП «ПРЕДИКАТОР» в кри- тических случаях пригоден для определения оптимальной нагрузки на ЦП.

Особенно широкое и полное применение система ШЕДОУ II может найти на виртуальных ЭВМ.

### **Генератор программ телеобработки ЦПГ**

На время создания систем on-line и эффективность их приме- нения в значительной мере влияют те средства программного обе- спечения, которые используются для разработки прикладных про- грамм. Из-за отсутствия в ЕС ЭВМ средств для подготовки тран- закционных программ, пригодных для работы в режиме on-line, в генераторе были использованы ориентированные на пакетную об- работку традиционные программные языки (Кобол, ПЛ/1 и т. п.).

У систем on-line, использующих монитор телеобработки ШЕ- ДОУ II, эту проблему решает коммуникационный генератор про- грамм ЦПГ — программный продукт, с помощью которого можно генерировать эффективные транзакционные программы, работаю- щие под управлением выбранного монитора. При этом наполовину сокращается время, необходимое для построения системы on-line (разработка, тестирование, реализация), и в то же время снижает- ся потребность созданной системы в ресурсах технических средств, в соответствии с чем значительно уменьшается время ее реакции.

ЦПГ функционально состоит из двух компонентов: центральной библиотеки вспомогательных программ ЦРЛ и трансляционной программы ЦПГ.

**Центральная библиотека вспомогательных программ** выполня- ет двойную роль: с одной стороны, она оптимизирует размеры и работу генерированных программ, с другой — обеспечивает неза- висимость их мониторов. Библиотека содержит обрабатывающие вспомогательные программы общего пользования, которые во время активизации монитора телеобработки непрерывно загружаются в раздел монитора, после чего сгенерированные программы ЦПГ, по

необходимости, вызывают и пользуются вспомогательными программами. Одновременно одну и ту же вспомогательную программу могут использовать несколько программ, что сокращает размеры транзакционных программ и потребность в оперативной памяти.

Одна специальная программа центральной библиотеки вспомогательных программ обеспечивает связь с актуальным монитором телеобработки. Все программы ЦПГ через эту программу подключаются к монитору, причем сами программы не содержат инструкций, находящихся в зависимости от монитора. Вследствие этого оригинальные программы ЦПГ без изменений могут быть перенесены из среды одного монитора в другую. Переход от одного монитора телеобработки к другому, в сущности, означает замену специальных программ ЦПГ.

**Язык программирования ЦПГ**, служащий для написания транзакционных программ, содержит восемь форм спецификации различного типа, одним из назначений которых является структурирование программ для облегчения их проектирования, тестирования и сопровождения. Пять форм спецификации необходимы для написания дополнительно задаваемых определений, а три, задаваемые в обязательном порядке, служат для управления процессом обработки, описанном в программе.

Одно из различий между языком ЦПГ и другими языками программирования вытекает из ориентированности генератора ЦПГ на абонентские пункты. Передача данных, выполненная с дисплейных абонентских пунктов, означает ввод-вывод определенных полей экрана, поэтому ЦПГ считается скорее языком обработки полей, чем записей. Для этого языка характерно:

- автоматическое выполнение функций обслуживания, запросов буферов, рабочих зон;

- поддержка использования всех арифметических и логических действий;

- возможность арифметической обработки массивов, таблиц решений;

- оказание помощи в использовании базы данных;

- использование языка без знания монитора телеобработки;

- простота синтаксиса, обеспечивающая легкость изучения языка и его применения.

Из исходных программ, написанных на языке ЦПГ, транслятор ЦПГ генерирует программы на языке Ассемблера, из которых дальнейшим транслированием и конструированием создаются рабочие программы. Сгенерированные программы автоматически удовлетворяют требованиям повторного запуска. Таким образом, одна и та же транзакционная программа может запускаться и параллельно прогоняться с нескольких конечных абонентских пунктов одновременно.

**Поддержка разработки систем.** Генератор ЦПГ обеспечивает много таких возможностей, которые значительно упрощают разработку транзакционных программ.

Созданию полных программ ЦПГ, планированию, подготовке и встраиванию в программу кадров экрана, управляемых программами, способствует возможность ввода программы on-line. Пользователи непосредственно с конечных абонентских пунктов могут разрабатывать транзакционные программы, могут запросить синтаксический контроль инструкций ЦПГ и непосредственное хранение готовых программ. Генератор ЦПГ обеспечивает диалоговое хранение программ.

Интегральной частью ЦПГ является средство проектирования кадра. Пользователь с помощью простых стандартных инструкций может запросить проектирование кадров, не зная особенностей технических средств дисплейных абонентских пунктов. Генератор программ телеобработки является открытой системой. Он обеспечивает доступ к массивам и базам данных, которые ЦПГ непосредственно не поддерживает, вводит пользовательские вспомогательные программы в центральную библиотеку.

Сравнение транзакционных программ, написанных на языке ЦПГ и на традиционных языках, во многих областях доказывает эффективность ЦПГ:

- размер программы ЦПГ приблизительно в 6 раз меньше, чем размер транзакционной программы, выполняющей ту же задачу;
- программы ЦПГ занимают в 5 раз меньше оперативной памяти, чем программы, выполняющие ту же задачу, но написанные на языке Кобол или ПЛ/1;

- сокращенная потребность в емкости оперативной памяти уменьшает и потребность в ресурсах центрального процессора; для программ ЦПГ значительно уменьшается и время ответа;

- независимые от монитора программы ЦПГ переносимы с одной ЭВМ на другую, в то время как перенос программ, написанных не на языке ЦПГ, требует значительной дополнительной работы программистов;

- при использовании программ ЦПГ значительно снижается трудоемкость программирования.

Генератор ЦПГ прогоняется на всех вычислительных машинах типа ЕС ЭВМ-2, на которых она может быть использована в любой системе Единой системы ОС СВС или ОС ВС1, ВС2 или ДОС ВС работает ШЕДОУ II. Поскольку язык ЦПГ ориентирован на абонентские пункты, то его можно использовать только через дисплейный абонентский пункт.

### **Диалоговая система разработки программ ГУТС**

Диалоговая система разработки программ ГУТС, созданная в Гётеборгском университете (Швеция), имеет собственный способ передачи данных, поэтому она может быть использована в любой системе ОС ЕС, в которой имеется подсистема КРОС. ГУТС предоставляет для машин Ряда-1 и Ряда-2 современное высокоэффективное диалоговое средство, которое можно применять на моделях небольшой производительности.

ГУТС около года использовалась на ЭВМ ЕС-1035 предприятия САМАЛК с польскими и венгерскими экранными терминалами. В системе предусмотрено собственное средство ведения массива данных, которое оптимально использует имеющуюся в его распоряжении область памяти. Эти преимущества создаются благодаря сжато-формату экрана и цепочечной организации данных. Кроме того, обеспечивается эффективная защита данных (пароль, защитные ключи массива данных, уровни компетенции пользователей).

Главная программа ГУТС МАСТЕР состоит из загружаемых модулей; она может обслуживать одновременно сколько угодно пользователей. Вся работа, за исключением диалоговой трансляции и прогона (например, составление текста, печатание списка массива данных ГУТС и т. п.), выполняет главная программа.

ГУТС в процессе функционирования самостоятельно составляет документы. Кроме того, она обладает обучающими возможностями. Эта функция, наряду с большим количеством специальных команд, реализуется следующим образом: каждая команда на параметр «?» выписывает на абонентском пункте синтаксис и функцию запрошенной команды с кратким пояснительным текстом.

Система располагает эффективными возможностями составления текста. В ней представлены два вида редакционных программ: нормальная редакционная программа на каждый абонентский пункт и редакционная программа на экранные абонентские пункты, которая использует экран в разделенном режиме. Максимальная длина строки конструируемого массива данных — 133 символа.

Система позволяет создавать процедуры, в которых часто используемые команды (например, команды редактирования) предварительно помещают в один массив данных и одной-единственной командой запрашивают выполнения этих команд. Выполнению процедур помогает развитый макроязык, который может применяться для этих целей. Макроязык разрешает:

- применение позиционных и символических параметров;
- выполнение условных команд и команд перехода;
- диалоговый режим выполнения команд;
- последовательные вызовы в многократных циклах;
- использование кнопок ПФ на экранных абонентских пунктах.

В поставляемых пользователю системах имеется около 400 полезных процедур, которые расширяют базовый набор команд.

ГУТС формирует задачи, составленные в диалоговом режиме, для выполнения в пакетном режиме. Последовательность задач, хранящаяся в массиве данных ГУТС, готова для прогона: может запрашиваться состояние задачи, ожидающей в ОС ЕС или прогоняемой, после чего задача стирается. Имеется список законченных задач, который можно вывести на АЦПУ или на абонентский пункт, можно разместить в массиве данных ГУТС. Подготовке прогона в пакетном режиме способствует синтаксический контроль (Фортран, Бейсик).

Для поддержки массивов данных ОС в системе имеется большое количество команд и процедур, часть которых выполняется

внутри программы МАСТЕР, в результате отпадает необходимость в отдельном разделе памяти.

Система обеспечивает диалоговую работу с массивом. Массивы данных, определенные в программах во время прогона, могут быть массивом ОС, массивом ГУТС, массивом абонентского пункта (для непосредственной связи) или массивом данных. Для разработки программ может быть использован транслятор Бейсик, имеется модуль стыковки с транслятором ОС. При построении программы могут использоваться модули массивов данных ОС и ГУТС, пользовательские макромодули, а также библиотечные модули, находящиеся в объектном или загружаемом виде. Система содержит большое число сервисных программ (независимых от ГУТС) и программу «Ватерлоо СКРИПТ», с помощью которой можно выполнять и сопровождать документацию.

Система ГУТС пригодна для всех конфигураций ЕС ЭВМ, которые работают с ОС и имеют минимальный комплект абонентских пунктов.

УДК 681.3.06

## БРАТИСЛАВСКАЯ ПРОГРАММНАЯ СИСТЕМА BPS

П. Фишер, инженер (ЧССР)

Система BPS, эксплуатируемая в Научно-исследовательском вычислительном центре в Братиславе (ЧССР), представляет собой систему программ, служащую для автоматизации разработки больших систем программ. В ее основу положены автоматическое применение современных методов программирования и интеграция средств программирования.

Выбранные методические принципы наглядно отражаются в языке программирования BPS/L, который используется в рамках системы BPS. Отдельные команды на языке BPS/L отвечают принципам структурированного программирования, а общая структура программы, наглядность и использование объектов, обращение с переменными и операциями — принципам модульного программирования.

BPS проектируется как интегрированная система программирования. Она состоит из взаимодействующих подсистем (рис. 1). Важной частью ее является библиотека системы и управляющая программа. В библиотеке BPS хранятся все программы, создаваемые при ее помощи. Кроме исходных текстов в библиотеке находится также информация о программах, необходимая для работы и взаимодействия отдельных подсистем. Пользователь имеет доступ лишь к исходным формам программ.

Со всеми другими видами информации в библиотеке работает лишь сама система. Управляющая программа принимает команды от пользователя и интерпретирует их. При этом она активизирует отдельные подсистемы и доставляет им нужные данные из библиотеки.

Языковые средства системы BPS состоят из компилятора системного языка BPS/L и ассемблера.

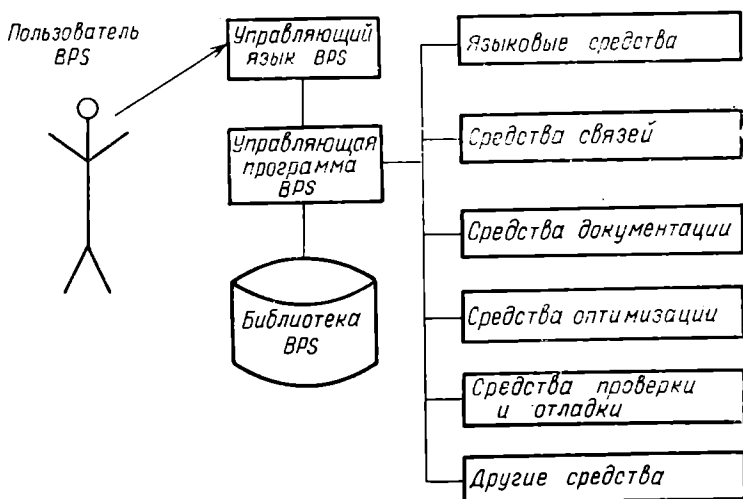


Рис. 1. Общая структура системы BPS

Проектирование компилятора BPS/L осуществлялось на базе языка, разработанного профессором Виртой. Одна из целей проектирования — распространить применение языка программирования не только на фазу кодирования, но и на проектирование структуры программы.

Основной программной единицей в компиляторе BPS/L является модуль. Программа, или же программная система, написанная при помощи языка BPS/L, состоит из нескольких модулей. Связи между модулями точно определены и должны составлять иерархическую структуру (ациклический граф). Правила выделения и применения отдельных программных объектов (переменные и процедуры) также исходят из иерархической структуры. Модули на более высоком уровне иерархии могут пользоваться объектами из модулей на более низких уровнях иерархии. Каждая связь этого типа должна быть явно определена в программе. Возьмем, например, структуру, показанную на рис. 2.

Стрелки в заданной структуре обозначают все возможные связи. В соответствии с ними модуль MAIN может пользоваться объектами из модулей M1, M2 и M5. С другой стороны, хотя модуль M2 находится на более высоком уровне, чем модуль M3, он не может



пользоваться его объектами, так как связь между ними не определена. Связь между модулями определяется в заголовке модуля, в USE-списке. Для модуля MAIN он выглядит следующим образом:

```
MODULE MAIN;
USE M1, M2, M5
```

Важно точно определить взаимную связь и интерфейс между модулями, а также установить, какие объекты модуля могут быть использованы вне заданного модуля. Все объекты, которые должны использоваться вне модуля, следует описать в декларационной секции, которая обозначена зарезервированным словом GLOBAL. После этого в модуле видны лишь эти объекты. Компилятор контролирует правильное применение отдельных объектов. В качестве примера описания глобальных объектов приведем модуль, составляющий таблицу, в которой можно хранить пары и значения модулей.

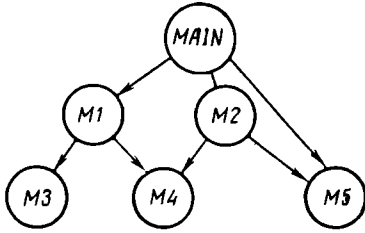


Рис. 2. Пример связи модулей

```
MODULE TABLE;
USE ... ;
GLOBAL
TYPE KEYS — ... ;
  VALTYPE — ... ;
VAR DOUBLE: BOOLEAN ;
OP INSERT: IN KEYS, IN VALTYPE;
  DELETE: IN KEYS;
  MEMBER: IN KEYS, OUT VALTYPE BOOLEAN;
END TABLE;
```

В секции TYPE описаны типы ключа и части значения из пары. Это дает возможность создать в модулях, применяющих модуль TABLE, переменные при помощи этих типов. Вне модуля можно пользоваться булевой переменной DOUBLE, которая после вызова процедуры INSERT сигнализирует о двойной записи пары с одинаковым ключом. Кроме упомянутой выше процедуры, в секции OP (в которой описываются процедуры и функции) описаны также процедура DELETE и булевая функция MEMBER. При описании операции приводятся имя операции, количество, тип и способ использования параметров.

Декларационная секция, обозначенная зарезервированным словом GLOBAL, иногда называется синтаксической спецификацией модуля. Необходимо также указывать семантику модуля. Однако пока нет формальных методов спецификации семантики, применимых для повседневной практики, в системе BPS следует специфицировать семантику неформально.

Предполагается, что конструкции, описанные на языке BPS/L (USE-список и спецификации модуля), позволяют использовать его в фазе проектирования структуры программы. Модуль TABLE, как он был описан выше, является действительной компиляционной единицей. После его перевода возможна компиляция модулей на более высоких уровнях иерархии, которые пользуются модулем TABLE. Система BPS контролирует соблюдение типовой дисциплины. В библиотеке отмечено, что запуск модуля TABLE в этой форме невозможен. При попытке создать программную единицу, элементом которой является модуль TABLE, система сигнализирует об ошибке.

Модуль TABLE подходит также для скрытия информации (information hiding). При использовании модуля проектировщика не интересуют детали установки модулей, а только назначение заданных в них операций и как ими следует пользоваться. Так, в случае применения модуля TABLE неважно, где расположена таблица (в памяти или на диске) и каковы механизмы доступа. Каким образом модуль выполняет свои задачи, проектировщику важно знать лишь при реализации модуля. В этой фазе программист пользуется командами языка программирования. Команды языка BPS/L заимствованы из языка MODULA, следовательно, у BPS/L структура данных и управляющая структура достаточно развиты. Структура полного модуля выглядит следующим образом:

|  |  |
|--|--|
| Имя модуля   | MODULE XYZ;  |
| Список модулей, объектами которых можно пользоваться | USE M1, M2, ..., MN;   |
| Описания объектов, используемых вне модуля           | GLOBAL<br>CONST ...<br>TYPE ...<br>VAR ...<br>OP ...                                 |
| Описания локальных объектов                          | LOCAL<br>CONST ...<br>TYPE ...<br>VAR ...<br>OP ...                                  |
| Инициализация локальных и глобальных переменных      | VALUE ...  |
| Тело модуля  | BEGIN ...<br>END XYZ;  |
| Тела глобальных и локальных операций                 | PROC P1, P, Q;<br>CONST<br>TYPE<br>VAR<br>BEGIN ... END P1;<br>PROC P2 ...<br>:<br>: |

Каждый модуль представляет собой отдельно компилируемую единицу. Основной проблемой компиляции в этом случае является

определение компиляционной единицы, модули определяются естественным выбором единицы для отдельной компиляции.

В модуле имеются только два уровня: глобальные и локальные данные (видимые во всем модуле) и операции с собственными локальными структурами данных. Удвоенные декларации недопустимы. С этой точки зрения следующая декларация также является запрещенной:

```
VAR I : INTEGER; X : RECORD I, J : CHAR END;
```

В пределах команды WITH

```
WITH X DO ... END;
```

знак I скрывал бы целочисленную переменную I.

Как упоминалось ранее, команды и формы выражений заимствованы из языка MODULA; единственным исключением являются явно заданные типовые конверторы. В языке MODULA не допускались записи (тип record) с вариантами (в отличие от языка MODULA2), однако почти все таблицы, используемые в системном программировании, необходимо описывать в двух вариантах, но внутри модуля следует отобразить структуры разного типа в одной и той же памяти. Для этого служат явно заданные типовые конверторы. В языке BPS/L можно преобразовывать любой краткий тип (integer, char, boolean, bits, enumeration, wordchar) в любой другой краткий тип, например:

```
VAR C : CHAR; I : INTEGER; ... I : —INTEGER/«A»/; C : —CHAR/TRUE/;
```

Явно заданные типовые конверторы могут быть, однако, использованы в модулях, обозначенных зарезервированным словом DEVICE, чем указывается их связанность с конкретной ЭВМ. Эти модули следует просмотреть в случае, если программы из одной ЭВМ переносятся на другую.

При программировании больших систем важным остается вопрос о том, вводить ли программы ассемблера в язык. В языке BPS/L эта проблема решена следующим образом: имеются модули, обозначенные зарезервированным словом ASSEMBLER, заголовков и декларации глобальных объектов которых описаны на языке BPS/L. Такой модуль имеет следующую форму:

```
ASSEMBLER MODULE ASS;
  GLOBAL OP ASSPROC;
END ASS;
ASS          CSECT
             ENTRY ASSPROC
ASSPROC     EQU
            .
            .
            .
            END
```

Составную часть системы BPS представляют средства связей. Их задача — связать отдельные модули в единицы или подготовить все данные для связи при помощи стандартных системных средств связи и контролировать при этом правильность отношений между

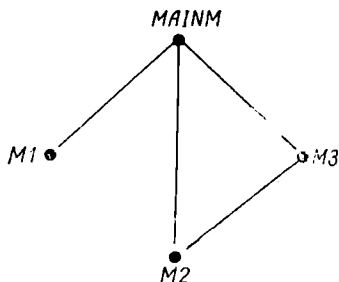
модулями. На основе информации, хранящейся в библиотеках (список использованных модулей), BPS обнаруживает все модули, принадлежащие заданной единице, их связи и взаимодействие. Взаимоотношения между модулями отображены в графе видимости, который создается системой и предоставляется пользователю. Приведем пример. Пусть модуль MAINM

```
MAIN MODULE MAINM;  
  USE M1, M2, ME;  
END MAINM;
```

использует (видит) модули M1, M2, M3. Модули M1 и M2 в дальнейшем никакие модули не применяют, а модуль M3 использует модуль M2.

```
MODULE M3;  
  USE M2;  
  .  
  .  
END M3
```

Граф видимости системы описанных модулей выглядит следующим образом:



Ацикличность созданного ориентированного графа автоматически проверяется связующей программой. В дальнейшем следует проверить правильность связей между модулями. Это значит, что модули, использующие другие модули, были транслированы после введения в библиотеку последней версии глобальной декларации использованных модулей. Если пользователь в своих программах внес много изменений и не разбирается в том, что следует перекомпилировать, он может попросить этой услуги у BPS. Система автоматически обнаружит, какие модули и в каком порядке следует перекомпилировать.

Задача средств документации — автоматическое генерирование документации программ, создаваемых при помощи BPS. Средства документации могут использоваться как самостоятельно, так и вместе с другими. При компиляции модуля можно получить список перекрестных ссылок объектов, которые встречаются в модуле. При объектах, определенных в компилируемом модуле, указываются их атрибут (глобальный, локальный), место определения и ссылки на

место, где они используются. При введенных в модуль объектах выписывается, кроме их атрибута, также имя модуля, из которого они введены. При связывании модулей документируется структура всей программы. Граф видимости при этом распечатывается в понятной пользователю форме.

Средства документирования могут быть использованы также самостоятельно. В этом случае документируется структура программы, за которой следует распечатка отдельных модулей вместе с перекрестными ссылками на объекты, использованные в заданном модуле. В конце распечатывается список глобальных объектов из всех модулей, вместе с назначением места их определения и использования. Составленная таким образом документация представляет собой хорошую базу для возможных модификаций созданной программной системы.

С точки зрения пользователя средства оптимизации системы не являются самостоятельной подсистемой. Одни из них относятся к интегральной части транслятора, другие — к языковым средствам; работу последних можно включить при помощи соответствующих команд управляющего языка.

Средства проверки и отладки обеспечивают интерактивную (в режиме on-line) и неинтерактивную (в режиме off-line) проверку и отладку программ, написанных на языке BPS/L. Пользователь имеет в своем распоряжении все основные функции таких систем: он может прекратить прогон программы в заранее определенных местах (break-point), изменить значения объектов, проверить их значения, повторить запуск программы и т. д. Стержнем системы является интерпретатор языка BPS/L.

Предполагается прибавить к системе BPS средства, которые собирали бы и представляли пользователю разные статистические данные о составлении программ (например, количество рекомпиляций отдельных модулей, количество тестовых прогонов и т. п.). Эти данные могли бы служить основой для разных плановых и управляющих работ.

Система BPS поддерживает все этапы создания программ, кроме фазы спецификации. Автоматизация касается прежде всего документирования программ, созданных в рамках BPS, учета и сопровождения программ в библиотеках системы, соединения отдельных модулей программ в единое целое, исправления межмодульных связей (в случае их несовместимости).

Преимущества системы BPS проявляются прежде всего тогда, когда она используется для создания широких программных систем (компиляторы, информационные системы, операционные системы) с большим количеством внутренних взаимозависимостей.

BPS следует применять лишь в том случае, когда имеются функциональные спецификации для создаваемой программы (или программной системы) и начинается проектирование архитектуры будущей программной системы.

Система BPS не автоматизирует работы по организации, планированию и управлению, но она может предоставлять данные для этих работ.

Документация по системе BPS состоит из трех справочников: справочника по использованию системы BPS (для каждого отдельного типа ЭВМ, на которую BPS внедряется: ЕС ЭВМ, СМ ЭВМ и т. д.);

справочника по программированию на языке BPS/L; инструкции по установке системы BPS (для каждого отдельного типа ЭВМ).

Система BPS требует 120 Кбайт оперативной памяти для ОС ЕС ЭВМ или 23 Кбайт СМ-4/20; для библиотеки системы BPS необходимо отвести один диск.

Систему BPS поставляет организация ПОЛЫТЕХНА, Прага, Панска ул., 9, ЧССР (POLYTECHNA, PRAHA, Panská 9, ČSSR).

УДК 681.3.06

#### ПРИКЛАДНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ СМ ЭВМ

В. С. Кузнецов, канд. техн. наук (СССР)  
В. П. Куприянов, канд. техн. наук (СССР)  
А. А. Азеев, канд. техн. наук (СССР)

СМ ЭВМ относятся к классу мини-ЭВМ, которые получили большое развитие в разных областях применения. Мини-ЭВМ создавались как средства автоматизации управления технологическими процессами и лабораторными экспериментами, поэтому их технические характеристики и программное обеспечение (ПО) были ориентированы на такое применение. Они обладали малыми объемами оперативной и внешней памяти, а операционные системы (ОС) и прикладное программное обеспечение (ППО) предназначались в основном для режима реального масштаба времени (РМВ) с целью получения данных с различного типа датчиков, выполнения вычислений и выдачи управляющих сигналов к объекту.

Улучшение технических параметров СМ ЭВМ позволило расширить сферу применения СМ ЭВМ и использовать их в системах управления промышленными предприятиями, в системах управления непромышленными объектами (банки, гостиницы и т. п.), в информационно-поисковых и справочных системах различного рода и в системах сбора и обработки данных производственного характера.

Многообразие областей применения СМ ЭВМ, их массовый выпуск, широкий набор периферийного оборудования предопределили в качестве главного требования к ПО проблемную ориентацию.

ПО СМ ЭВМ строится как совокупность различных по своим функциональным возможностям операционных систем и расширяющих их возможности пакетов прикладных программ.

Основными особенностями систем, построенных на базе СМ ЭВМ, являются работа в реальном масштабе времени, сочетание решения на одном вычислительном комплексе задач управления технологическими процессами и задач оперативного управления и планирования, применение для сбора и выдачи информации сети удаленных терминалов, подключаемых к вычислительному комплексу через линии связи, работа в условиях непредсказуемости момента обращения абонента к ресурсам системы.

Прикладное программное обеспечение АСУ можно разбить на пять основных групп:

- общего назначения;
- организации вычислительного процесса;
- технологии и автоматизации программирования;
- методо-ориентированных расчетов;
- функционального назначения.

Программные средства общего назначения позволяют пользователям проектировать системы ввода, контроля и преобразования информации, организовать и вести базы данных различной структуры, разрабатывать информационно-справочные и информационно-поисковые системы, обеспечивать хранение данных и организацию поколений данных и др.

Перечисленные функции реализуются пакетами прикладных программ (ППП) и системами программного обеспечения (СПО): ППМ «Банк», СПО, «База-СМ», СПО «База-ввод-СМ», СУБД «СЕТОР-СМ», СПО «БСП-СМ», СПО «Сорт», СПО «ДИАМС».

Пакет прикладных модулей для работы с базами данных (ППМ «Банк») предназначен для создания больших взаимосвязанных массивов информации и работы с элементами этих массивов на ЭВМ СМ-2. Программы пакета обеспечивают одновременную работу с восьмью базами данных, при этом доступ к ним могут иметь до 100 абонентов.

Система управления базами данных «СЕТОР-СМ» используется при создании баз данных сетевой и иерархической структуры и предназначена для широкого круга применений.

СПО «База-СМ» осуществляет формирование и ведение универсальной по структуре и составу внутримашинной информационной базы АСУ для СМ ЭВМ. Система выполняет функции хранения данных, управления хранимыми данными и обеспечения доступа к ним.

СПО «Дисковая диалоговая многопультная система» («ДИАМС») — комплекс программных средств, ориентированных на управление базами данных и решение информационно-логических задач на ЭВМ СМ-4.

СПО «База-ввод-СМ» предназначена для автоматизации программирования процедур ввода, контроля и преобразования вход-

ной информации, имеющей последовательную и иерархически-последовательную организацию.

СПО «Библиотека сервисных программ» (БСП-СМ) содержит программу управления данными и обслуживающие программы, которые обеспечивает создание, поддержку и обновление файла с последовательной, относительной и индексно-последовательной организацией на ЭВМ СМ-4.

СПО «Сорт» предназначена для сортировки файлов с последовательной, относительной и индексно-последовательной организацией. Работает под управлением ОС РВ версии 2 и расширяет функциональные возможности БПО.

Возможности программных систем общего назначения могут быть использованы с большой эффективностью при организации вычислительного процесса. В этом случае они предоставляют пользователю средства организации вычислительных центров коллективного пользования, средства связи между ЭВМ различного типа, средства организации сети ЭВМ. К их числу относятся СПО «КП-СМ» и ППП «ОС СМ».

СПО «КП-СМ» (система программного обеспечения обработки данных многомашиным вычислительным комплексом) предназначена для управления вычислительным процессом в отдельных АСУ. Она позволяет объединить несколько АСУ на одном вычислительном комплексе. Эта СПО используется в системах обработки информации, поступающей с удаленных видеотерминалов ЭВМ СМ-2 в реальном масштабе времени.

ППП «ОС СМ» (ППП обмена и обработки сообщений на линиях связи), реализующий функциональные возможности телеобработки на СМ-4, объединяет в себе две системы — систему обработки транзакций (СОТ) и систему передачи сообщений по линиям связи (СПСЛ). СОТ через СПСЛ обслуживает поступающие с удаленных терминалов и терминальных станций запросы на обработку данных и обеспечивает при этом быстрый, надежный и независимый доступ к архивам данных.

Прикладное программное обеспечение ни одной из ЭВМ не обходится без средств автоматизации программирования. ППП «Генератор табуляграмм СМ-2», ППП «Диалог», ППП «Фобрин-2», ППП «Кобол-СМ» дают возможность пользователю применять проблемно-ориентированные языки программирования при проектировании АСУ, автоматизировать процесс проектирования АСУ, подготовки и выдачи проектной документации.

ППП «Организация диалоговых процедур» (ППП «Диалог»), функционирующий на одно- или двухпроцессорных комплексах СМ-1 и СМ-2, предназначен для организации диалоговых процедур в АСУ.

Программная система компиляции и отладки программ на языке Кобол для СМ-4 (ППП «Кобол-СМ») — процедурно-ориентированный язык программирования экономических задач.

ППП «Генератор табуляграмм-СМ» предназначен для автоматизации вывода экономической информации в виде документов



сложной структуры на устройство вывода в удобном для чтения виде.

Программная система ввода, первичной обработки и вывода информации в диалоговом режиме на СМ-4 (ППП «Фобрин-2») обеспечивает ввод, контроль и обновление данных в файлах пользователя, а также их обработку и вывод результатов, требуемых пользователю.

Программные средства методо-ориентированных расчетов реализуют различные классы экономико-математических методов, которые применяются в АСУ для научно-технических расчетов и научных исследований. Широко используются программные средства в виде пакетов прикладных программ «Обработка данных методами математической статистики» («ПАСТ»), «Библиотека для научно-технических расчетов».

К программным средствам функционального назначения относятся наборы программных типовых модулей, служащие для автоматизации обработки данных. Причем обработка данных рассматривается не в общем виде, а выделяются алгоритмы, несущие функциональную нагрузку АСУ. С помощью ППП «Кадры-КИД-СМ» создается ПО подсистемы «Управление кадрами» и решается комплекс задач «Контроль исполнительской дисциплины». Пакет функционирует на ЭВМ СМ-4 как в пакетном, так и в диалоговом режимах.

Пакеты прикладных программ для СМ ЭВМ характеризуются универсальностью в пределах соответствующих классов задач, многообразием реализуемых алгоритмов, малой зависимостью от типов используемых технических средств и легкостью настройки на условия конкретного применения.

Высокая эффективность использования вычислительной техники достигается не только путем снижения стоимости технических средств, но и путем применения, развития и усовершенствования инструментальных средств программного обеспечения СМ ЭВМ. Это развитие и усовершенствование происходит по направлениям:

- разработки средств генерации форматов экрана;

- создания непроцедурных языков запросов для пользователей-непрограммистов;

- эмуляции интеллектуальных терминалов для удаленного ввода заданий и возможности обработки файлов (данных) в распределенной сети ЭВМ;

- создания программного окружения ПС общего назначения различного рода сервисными средствами, в том числе облегчающими проектирование структуры информационной базы, загрузки и ее актуализации;

- развития комплексных программных средств ведения линейных независимых файлов и систем управления базами данных различной структуры;

- создания средств, обеспечивающих работу многомашинных комплексов и сетей ЭВМ;

- разработки средств автоматизации программирования и совер-

шенствования технологии процесса производства ПО, включая создание кросс-систем;

создания программных частей проблемно-ориентированных комплексов, обеспечивающих управление объектами или процессами данного класса.

В связи с широким использованием малых ЭВМ наметилась тенденция интенсивной разработки централизованных и распределенных систем управления базами данных. Прогнозные оценки показывают, что к 1985 г. потребность в СУБД различного назначения составит до 50% общего потребления изделий прикладного программного обеспечения. Наиболее характерными для перспективного применения СМ ЭВМ являются распределенные системы управления базами данных (СУБД). Принципы организации СУБД «СЕТОР-СМ» — основной системы управления базами данных — и логическая структура данных совместимы соответственно с принципами организации и логической структурой данных СУБД «СЕТОР» для ЕС ЭВМ и микроЭВМ. Это создает предпосылки для разработки систем децентрализованной обработки данных, построенных с использованием ЕС, СМ и микроЭВМ.

Система СУБД «СЕТОР-СМ» используется в разработке СПО внутримашинной базы АСУ (СПО «ВИБ-СМ»). СПО «ВИБ-СМ» — комплексная система программного обеспечения банков данных, удовлетворяющая разнообразным потребностям широкого круга пользователей. Она состоит из ряда взаимосвязанных ППП, каждый из которых может функционировать автономно либо во взаимодействии с остальными ППП.

Ядром системы является СПО «База-СМ». Она предназначена для создания средств, обеспечивающих в рамках единой информационной базы интеграцию данных, накапливаемых в отдельных локальных файлах и базах данных, организованных средствами различных СУБД, а также интеграцию всех метаданных в системах обработки данных (СОД). При этом доступ к образуемой виртуальной информационной базе обеспечивается с помощью операторов языка манипулирования данными, не зависящего от конкретной СУБД. Основным преимуществом СПО «ВИБ-СМ» является высокая степень независимости программ не только от физической, но и от логической структуры данных, благодаря чему достигается переносимость прикладного и базового ПО с одной СУБД на другую, повышается гибкость СОД и создаются предпосылки к эволюционному развитию внутренней структуры информационной базы. На этих принципах разрабатывается ряд программных средств, реализующих интерфейсы различных категорий пользователей банка данных.

Основным инструментом администратора базы данных является словарь (справочник) данных. Его языковые и программные средства обеспечивают поддержку проектирования базы данных, начиная с этапа конструирования концептуальной схемы до физического проектирования базы данных. При этом осуществляется формальный контроль полноты, непротиворечивости и нормализо-

ванности схемы. В процессе эксплуатации базы данных словарь (справочник) является единственным активным источником всех метаданных, используемых в системе. На всех этапах словарь обеспечивает полное документирование БД.

Программисту предоставляется процессор доступа, обеспечивающий манипулирование данными в табличной (реляционной) форме, независимой не только от физической, но и от логической организации данных. Средства языка манипулирования виртуальными данными позволяют выполнять операции покортежного доступа к информационной базе. Конечному пользователю для взаимодействия с базой данных предоставляется диалоговый язык запросов.

СПО «ВИБ-СМ» рассчитана на работу в многопользовательском режиме, причем увеличение количества обслуживаемых ею терминалов не оказывает существенного влияния на используемые ресурсы. СПО «ВИБ-СМ» функционирует на ЭВМ СМ-4 в среде ОС РВ 2.0.

Аналогичные программные средства разрабатываются для ЭВМ СМ-2. В дальнейшем предусматривается включение в «ВИБ-СМ» программных средств, обеспечивающих эффективное функционирование системы на ЭВМ СМ-1600, а затем — на СМ-1420.

Внедрение СМ ЭВМ с применением системного и прикладного программного обеспечения привело к активному использованию систем обработки данных для объектов непромышленной сферы. Малые ЭВМ становятся основой большинства систем управления в этой сфере.

СПО «БСП-СМ», «Сорт» и «Фобрин» внедрены в АСУ портами Минречфлота. В состав объектов внедрения первой очереди входят 200 портов, которые являются самостоятельными хозяйственными транспортными предприятиями.

ППП используются в АСУ-Порт — составной части АСУ-Речфлот для функциональных подсистем оперативного управления работой порта, управления пассажирскими перевозками, управления грузовой и коммерческой работой, оперативного учета и анализа производственных показателей порта.

Эти же ППП внедрены в АСУ-Союзпищепром в подсистеме информационного обслуживания руководящих работников с отображением справочных данных на видеотерминалах, а также в подсистеме оперативного управления для контроля выполнения планов, материально-технического снабжения, капитального строительства, обеспечения транспортом и др.

Информационная база АСУ-Комбинат, предназначенная для управления элеваторной, комбикормовой и мукомольно-крупяной промышленностью, разработана с помощью СУБД «СЕТОР-СМ».

В АСУ-Гострудсберкасса, предназначенной для автоматизированного сбора и обработки информации центральной и подотчетных сберегательных касс по платежам и вкладам, используются ППП «БСП-СМ» и «ОС СМ».

ППП «ПАСТ» включен в состав программного обеспечения АСУ технологическим процессом свеклосахарного производства с

целью решения оптимизационно-прогностической задачи определения рационального режима ведения технологического процесса на сахарном заводе. Эффективно используется ППП «ПАСТ» в конструкторском бюро биофизической аппаратуры для решения задач обработки данных биофизических экспериментов, определения надежности аппаратуры и ее работоспособности, построения математических моделей разрабатываемых устройств, обработки результатов технических экспериментов, связанных с исследованием вибрационных характеристик ультразвуцифуг.

Приобретение программной системы АСУ в специализированной организации и постоянный контакт с такой организацией позволяют разработчикам АСУ снизить затраты на разработку, поддержание в работоспособном состоянии и повышение качества используемых ПС. Кроме того, сокращаются сроки разработки АСУ, так как при ее создании используются хорошо документированные ПС, прошедшие достаточную проверку на надежность функционирования. Одновременно появляется возможность использовать штат собственных программистов для решения непредвиденных проблем и уникальных задач.

УДК 681.3.06

#### **СИСТЕМА ПЕРЕВОДА И ВЫПОЛНЕНИЯ ПРОГРАММ СМ-2, РАЗРАБОТАННЫХ НА ПОДМНОЖЕСТВЕ ЯЗЫКА Кобол ЕС ЭВМ**

**Ю. Н. Первалов**, канд. техн. наук (СССР)  
**В. В. Островский**, инженер (СССР)  
**С. Н. Судаков**, инженер (СССР)

Интегрированная обработка данных с использованием сетей разнородных ЭВМ, замена ЭВМ ставят задачу переноса программ и данных с машины на машину. Использование машинно-независимых языков Фортран, Кобол, Бейсик, Алгол является наиболее доступным средством для создания переносимых программ. Однако компиляторы с этих языков не всегда имеются на малых ЭВМ, а различия реализации требуют перевода программ с одного диалекта языка на другой.

Задача компилятора — перевод с фиксированного машинно-независимого языка на любой заданный язык. В качестве такого языка часто выступает язык Ассемблера машины, на которую производится перенос, или язык виртуальной машины. Сам компилятор не обязательно должен быть переносимым, достаточно перестраивать генератор команд в соответствии с конкретной машиной. Поскольку процесс переноса основывается на промежуточном языке виртуальной машины или на языке Ассемблера, необходимо обес-

печатать возможность записи программ на магнитную ленту и считывание с нее.

На вычислительном центре Волжского автозавода разработана система перевода исходных программ на языке Кобол ЕС ЭВМ, написанных с ограничениями для ЭВМ СМ-2.

Процесс перевода разбивается на пять этапов:

трансляция исходной программы;

генерация команд расширенного мнемкода СМ-2;

перекодировка;

перевод с расширенного мнемкода в стандартный мнемкод;

ассемблирование.

Первые три этапа выполняются на ЕС ЭВМ, перевод и ассемблирование — на СМ-2.

Минимальная конфигурация и необходимое программное обеспечение модели ЕС ЭВМ: оперативная память емкостью 512 Кбайт; два диска емкостью по 29 Мбайт; один накопитель на магнитной ленте; операционная система ОС ЕС.

Для ЭВМ СМ-2: оперативная память емкостью 64 Кслов; один дисковод ИЗОТ 1370; один накопитель на магнитной ленте (ЕС ЭВМ); операционная система ДОС АСПО.

Разработанная система позволяет переносить программы, написанные на Коболе без синтаксических ошибок и отлаженные на ЕС ЭВМ, поддерживает работу с последовательными файлами и с файлами прямого доступа. Имеется возможность блокировки и разблокировки логических записей, обработки многотомных, файлов, формирования меток файлов и томов на магнитных лентах, выполнения нумерации блоков на магнитных лентах. Средства трассировки программ, распечатки дампа оперативной памяти облегчают отладку и проверку перенесенной программы.

Назначение логических номеров физическим устройствам или файлам на магнитных дисках выполняется перед запуском программы. Диспетчер пакетной обработки автоматизирует запуск заданий, заранее подготовленных на диске.

На этапе трансляции исходный текст программы переводится во внутренний язык системы. Транслятор, состоящий из четырех модулей, выполняет три просмотра. В результате первого просмотра формируются таблицы описания файлов, таблицы описания рабочих полей, таблица описания литералов, таблица символических имен данных. Во время второго просмотра создается таблица параграфов и секций, проверяются форматы языка на соответствие ограничениям, накладываемым реализацией языка на СМ-2. Третий просмотр формирует таблицу описания процедур и таблицу описания условий. Если в программе используются запрещенные конструкции, третий просмотр не выполняется, а выводятся диагностические сообщения.

Все таблицы во время трансляции хранятся в оперативной памяти, они записываются на диск после завершения всех просмотров.

Полученные таблицы литералов, рабочих полей, описания файлов, описания условий, описания процедур используются на этапе генерации.

Генерация служит для преобразования программы из внутреннего языка таблиц в расширенный мнемокод СМ-2. Генератор состоит из шести основных и нескольких вспомогательных подпрограмм, которые формируют заголовок программы на мнемокоде, включают команды описания файлов, обрабатывают иерархическую структуру раздела данных программы, написанной на Коболе, и строят последовательность макрокоманд, которая описывает рабочие поля и литералы. Эти макрокоманды являются расширением мнемокода СМ-2; они обрабатываются до ассемблирования программы на ЭВМ. Таблица процедур преобразуется в последовательность инструкций на мнемокоде. Для хранения адресов переходов и точек возврата операторов PERFORM строятся стеки.

После обработки каждого оператора из таблицы процедур сформированные строки текста программы записываются в файл на диске. Полученный файл используется на этапе перекодировки.

Программа перекодировки обрабатывает файлы с записями постоянной длины. Записи могут включать данные четырех типов: символьные, двоичные, десятичные, десятичные упакованные. Перекодировку можно выполнять из ДКОИ в код ASC-2 и наоборот. Эта программа перекодировки используется для переноса программ и файлов с машины на машину. Магнитная лента с перекодированной программой в дальнейшем обрабатывается на СМ-2.

Макрокоманды расширения мнемокода переводятся в инструкции на мнемокоде, записи из фиксированного формата преобразуются в переменный.

Ассемблирование выполняется стандартным транслятором мнемокода в ДОС АСПО.

Для получения работающей программы объектный модуль связывается стандартным компоновщиком ДОС АСПО с библиотекой подпрограммы, написанной на Коболе.

При выборе структуры объектного модуля учитывалось следующее:

- операционная система ДОС АСПО позволяет выделять разделы оперативной памяти, емкостью не более 32 Кслов;

- адресация оперативной памяти осуществляется по словам и нет адресации по байтам;

- отсутствует аппаратная реализация операций десятичной арифметики;

- для совместимости по данным с ЕС ЭВМ необходимо поддержание символьного, двоичного, десятичного, десятичного упакованного типов данных;

- арифметические операции выполняются с числами до 18 цифр.

Все операторы Кобола, за исключением GO TO, реализуются командами обращения к модулям из библиотеки стандартных программ языка Кобол на СМ-2, что позволило уменьшить размер объектного модуля.

Большие преобразования данных происходят при выполнении арифметических операций, операторов IF, MOVE. Выполнение этих операций с различными типами данных требует приведения операндов к одному типу и выравнивания относительно десятичной точки.

Для использования Кобола на СМ-2 было выбрано два типа арифметических операций: операции с двоичными данными и операции с десятичными упакованными данными.

Двоичная арифметика реализуется достаточно просто имеющимися на СМ-2 инструкциями. Для десятичной арифметики были разработаны программы десятичного сложения (вычитания), десятичного умножения и десятичного деления, работающие с десятичными упакованными данными.

Обращения к подпрограммам и параметры генерируются на ЕС ЭВМ. Использование зарезервированных рабочих полей и соглашение о порядке размещения операндов позволило уменьшить количество параметров вызова подпрограмм и таким образом значительно сократить объем раздела процедур в оперативной памяти.

Схема преобразований типов арифметических операндов приведена в табл. 1, а преобразований для операций сравнения — в табл. 2.

Для работы с магнитными лентами расширена стандартная библиотека управления данными. Добавлены модули контроля меток на магнитных лентах, модули блокировки (разблокировки) логических записей, модули формирования и контроля номеров блоков, модуль обработки многотомных файлов. Доступ к файлам осуществляется через расширенный блок управления данными.

Работа с индексированными данными выполняется через подпрограммы обработки индексов, которые используют описатели таблиц.

Приведенная на СМ-2 программа, написанная на Коболе, включает: управляющий оператор мнемокода (режим трансляции); оператор NAM; служебные рабочие поля; литералы; таблицы файлов; подпрограмму DLFIL; подпрограммы, реализующие кобельные операторы в порядке их следования в исходной программе; стеки операторов PERFORM; описатели индексированных рабочих полей; операторы EXT для подпрограмм библиотеки Кобола.

При компоновке программы можно подключать необходимый набор подпрограмм из библиотеки, что уменьшает общий объем программы.

Структура представления программы, переведенной на СМ-2, позволяет перенести кобельные программы без оверлейной структуры объемом 1500—2000 операторов раздела процедур. Эта цифра условная, так как увеличение в программе размера раздела рабочей памяти влечет за собой уменьшение количества операторов раздела процедур, однако объем выделяемого раздела не должен превышать 32 Кслов.

Подпрограммы стандартной библиотеки Кобола на СМ-2 проектировались с использованием метода наращиваемой макроструктуры. Этот подход позволил составить библиотеку стандартных программ языка Кобол на СМ-2 из программ, простых по логике, небольших по объему занимаемой памяти, достаточно оптимизированных по времени выполнения. Параметры обращения к подпрограммам определяются во время компиляции на ЕС ЭВМ. Библиотека включает 119 подпрограмм.

Таблица 1

| Тип второго и третьего операнда | Тип первого операнда |       |       | Тип второго и третьего операнда | Тип первого операнда |       |      |
|---------------------------------|----------------------|-------|-------|---------------------------------|----------------------|-------|------|
|                                 | D                    | C3    | C     |                                 | D                    | C3    | C    |
| D.D                             | D→C3                 | C3→C3 | C→C3  | DC                              | D→C3                 | C3→C3 | C→C  |
|                                 | D→C3                 | D→C3  | D→C3  |                                 | D→C3                 | D→C3  | C3→C |
|                                 | C3→D                 | C3→D  | C3→D  |                                 | C3→C                 | C3→C  | C→C  |
| C3.D                            | D→C3                 | C3→C3 | C→C3  | C3.C                            | D→C3                 | C3→C3 | C→C  |
|                                 | C3→C3                | C3→C3 | C3→C3 |                                 | C3→C3                | C3→C3 | C3→C |
|                                 | C3→D                 | C3→D  | C3→D  |                                 | C3→C                 | C3→C  | C→C  |
| C.D                             | D→C3                 | C3→C3 | C→C   | C.C                             | D→C                  | C3→C  | C→C  |
|                                 | C→C3                 | C→C3  | C→C   |                                 | C→C                  | C→C   | C→C  |
|                                 | C3→D                 | C3→D  | C→D   |                                 | C→C                  | C→C   | C→C  |
| Таблица 2                       |                      |       |       |                                 |                      |       |      |
| Тип второго и третьего операнда | Тип первого операнда |       |       | Тип второго и третьего операнда | Тип первого операнда |       |      |
|                                 | D                    | C3    | C     |                                 | D                    | C3    | C    |
| D.C3                            | D→C3                 | C3→C3 | C→C3  | D                               | C3                   | C3    | C3   |
|                                 | D→C3                 | D→C3  | D→C3  |                                 | C3                   | C3    | C3   |
|                                 | C3→C3                | C3→C3 | C3→C3 |                                 | C                    | C3    | C    |
| C3.C3                           | D→C3                 | C3→C3 | C→C3  | C3                              | C3                   | C3    | C3   |
|                                 | C3→C3                | C3→C3 | C3→C3 |                                 | C3                   | C3    | C3   |
|                                 | C3→C3                | C3→C3 | C3→C3 |                                 | C                    | C3    | C    |
| C.C3                            | D→C3                 | C3→C3 | C→C   | C                               | C3                   | C3    | C3   |
|                                 | C→C3                 | C→C3  | C→C   |                                 | C3                   | C3    | C3   |
|                                 | C3→C3                | C3→C3 | C3→C3 |                                 | C                    | C3    | C    |

Примечание. D — десятичное упакованное данное; C3 — десятичное упакованное данное; C — двоичное данное.

Программы трансляции и генерации написаны на Коболе ЕС, программа перекодировки — на языке Ассемблера ОС ЕС, а модули библиотеки стандартных программ языка Кобол СМ-2 — на мнемокоде ДОС АСПО.

Время выполнения программ, переведенных на СМ-2, возрастает в сравнении с выполнением их на ЕС-1033 в 2—2,5 раза. Время выполнения программ, написанных на Коболе на СМ-4, откомпилированных компилятором Кобола, по отношению к ЕС-1033 возрастает в среднем в 2 раза. Различие во времени обработки программ на СМ-4 и СМ-2 в основном объясняется различными характеристиками дисководов.



Компиляция программы, состоящей из 300 рабочих полей и 500 операторов раздела процедур, выполняется за 14 мин на ЕС-1055, а перевод программы, состоящей из 800 рабочих полей и 1200 операторов раздела процедур, — за 30 мин.

Описанная система перевода использовалась для перевода на СМ-2 программ оперативного бухгалтерского учета складов. Возможность перевода программ обеспечила применение типового программного обеспечения на вычислительных центрах, оснащенных разнородными ЭВМ: ЕС, СМ-2, СМ-4.

Разработка программ на ЕС ЭВМ дает возможность использовать все сервисные средства этих машин. Программирование на усеченном Коболе обеспечивает переносимость программ между ЭВМ ЕС, СМ-4, СМ-2.

Использование Кобола сокращает сроки отладки, экономит машинное время. Для разработки программы на Коболе, состоящей из 2000 строк, на ЕС ЭВМ потребуется 1—2 человеко-месяца и 2—3 ч машинного времени. Из-за отсутствия на СМ-2 языка высокого уровня и операций десятичной арифметики разработка аналогичной программы на мнемокоде потребует 3—5 человеко-месяца и 100 ч машинного времени. Перенос же кобольной программы из 2000 строк с использованием разработанной системы перевода осуществляется не более чем за 1 ч машинного времени на ЭВМ ЕС и СМ-2.

Практика показала, что система перевода программ повышает производительность труда программиста в 4—5 раз, а затраты машинного времени сокращаются в среднем в 10 раз.

Перечислим основные ограничения Кобола для СМ-2:

переводимая программа не должна содержать ошибок любого уровня;

строку продолжения может иметь только буквенно-цифровой литерал;

не допускаются строки продолжения в разделе процедур;

в качестве символов пунктуации можно использовать: пробел, точку, апостроф, левую и правую круглые скобки;

файлы могут иметь только фиксированный формат записей;

запрещены уровни данных 66, 77, 88;

максимальный размер литералов: буквенно-цифрового — 80 символов, цифрового — 16 цифр;

допустимые фигуральные константы: ZERO, SPASE, ALL «литерал», где литерал — один или два одинаковых символа;

для работы с файлами прямого доступа размер поля ACTUAL KEY должен быть не более 10 байт и перед вводом-выводом необходимо заносить в последние 2 байта этого поля двоичный порядковый номер записи в файле;

все поля в секции ввода-вывода описываются под уровнем 01;

записи файлов на магнитных лентах должны содержать четное количество байт;

разрешается только одномерная индексация, индекс может быть представлен именем двоичного данного;

максимальный размер символических имен — 18 символов, имя файла — 8 символов (используется 6 первых);  
 двоичное данное занимает 2 байта;  
 двоичное данное описывается только как целое;  
 если один из операндов в операции описан как двоичное данное, все остальные операнды обрабатываются как целые числа;  
 в операции IF первый операнд не может быть литералом;  
 вложенные оператором PERFORM не должны иметь одну точку выхода;  
 в разделе процедур нельзя использовать отрицательные литералы;  
 сравнение с фигуральной константой не может быть со связкой OR или AND.

Ниже приведены форматы языка Кобол с ограничениями для SM-2.

Раздел идентификации.

Нет ограничений.

Раздел оборудования.

ENVIRONMENT DIVISION.  
CONFIGURATION SECTION.  
SOURCE-COMPUTER.                    имя машины.  
OBJECT-COMPUTER.                    имя машины  
INPUT-OUTPUT SECTION.  
FILE-CONTROL.  
       SELECT имя файла ASSIGN TO системное имя  
       [ACCESS MODE IS    { SEQUENTIAL } ]  
                                   { RANDOM } ]  
       [ACTUAL KEY IS    имя данного].

Раздел данных.

DATA DIVISION.

FILE SECTION.

FD    имя файла

      [BLOCK CONTAINS            целое { CHARACTERS } ]  
   { RECORDS } ]  
       [RECORD CONTAINS            целое CHARACTERS]

LABEL RECORD IS    { OMITTED }  
                                   { STANDARD }

[DATA RECORD IS            имя данного].

      номер уровня {    имя данного }  
                           { FILLER }

[REDE FINES            имя данного-2]

[OCCURS            целое]

      [ { PICTURE }  
           { PIC }            строка литер]

[USAGE IS

{ DISPLAY  
COMP  
COMP-3 }

[VALUE IS                      литерал].

строка-лидер может состоять из S, V, X, Z, 9, ..., +, -, .

WORKING-STORAGE SECTION.

Запрещены номера уровней 66, 77, 88.

Раздел процедур.

PROCEDURE DIVISION.

OPEN INPUT            имя файла-1, ...

OUTPUT                имя файла-N, ...

CLOSE                 имя файла-1, ...

IF имя данного-1 условие-1 имя данного-2

{ OR  
AND }                условие-2    имя данного-3]

MOVE { имя данного-1 } TO имя данного-2, [имя данного-3]...  
          { литерал

GO TO                    имя процедуры.

ADD { имя данного-1 } TO    имя данного-2  
      { литерал

ADD { имя данного-1 } { имя данного-2 } GIVING имя данного-3  
      { литерал-1                      { литерал-2

SUBTRACT { имя данного-1 } FROM    имя данного-2  
              { литерал                      GIVING    имя данного-3

MULTIPLY                { имя данного-1 } BY    имя данного-2  
                              { литерал                      [GIVING    имя данного-3]

DIVIDE { имя данного-1 } INTO    имя данного-2  
          { литерал                      [GIVING    имя данного-3]

PERFORM                имя процедуры-1 [THRU    имя процедуры-2]

READ                    имя файла            [INTO    имя данного]

{ AT END  
INVALID KEY } GO TO    имя процедуры.

WRITE                    имя записи [FROM    имя данного]

[INVALID KEY GO TO    имя процедуры]

EXIT.                    имя данного-1                      [имя данного-2]...

DISPLAY

[UPON CONSOLE]

ACCEPT                имя данного                      FROM CONSOLE

STOP RUN

## ОДИН ПОДХОД К ПРОБЛЕМЕ ПОВЫШЕНИЯ НАДЕЖНОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

М. В. Либерман, инженер (СССР)

Современные тенденции развития программного и аппаратного обеспечения выдвигают требование повышения качества программных продуктов, сдаваемых в эксплуатацию. Одной из важнейших характеристик качества программ является их надежность. При этом следует учитывать, что даже в условиях использования самых современных методов программирования программные продукты, прошедшие этап разработки, содержат значительное количество ошибок («оставленных» программистом), в связи с чем для сдачи программ в промышленную эксплуатацию необходим этап отладки, нередко более продолжительный и трудоемкий, чем сама разработка. Однако и после отладки, как правило, программы не лишены ошибок, которые могут выявиться во время функционирования программ. В связи с этим чрезвычайно важной представляется задача повышения надежности самого инструмента работы программиста, т. е. языковых средств, которыми он пользуется при написании программ. Необходим подход, который позволил бы (помимо субъективных факторов, принимаемых во внимание в сегодняшней практике, — опыт работы персонала с тем или иным языком, качество компилятора) объективно выбирать наиболее подходящий для конкретного набора задач язык программирования на основе количественных оценок.

В рамках работ по созданию программного обеспечения СМ ЭВМ, в частности проектирования подсистемы обеспечения надежности для автоматизированной информационной системы управления перевозками на железнодорожном транспорте, был предложен такой подход, основывающийся на оценках надежности функционирования и метрических свойствах разрабатываемых программ.

В качестве базовой была введена характеристика «надежность языка программирования», а также предложена система показателей данной характеристики. Надежность языка программирования выражает свойство приспособленности языка для написания программ, не подверженных отказам на этапе функционирования.

Целесообразность и перспективность введения данной характеристики и использования оценки надежности в качестве критерия в тех случаях, когда существует ряд «конкурирующих» языков, видимо, будет возрастать по мере повышения качества и удешевления аппаратного обеспечения, с одной стороны, и увеличения удельного веса затрат на разработку программного обеспечения — с другой.

При разработке системы показателей надежности языка программирования использовались положения метрической теории программ Холстеда [1] и модели надежности программного обес-

печения, в частности модели Нельсона, Шумана [2]. В качестве примера показателя надежности языка программирования можно привести удельный коэффициент надежности языка программирования. Пусть задано:

множество программистов

$$Z = (z_{ij} : i = 1, 2, \dots, I), \\ j = m_1, m_2, \dots, m_J$$

где  $i$  — порядковый номер программиста;  $j$  — дискретное значение оценки профессионального мастерства программиста, причем  $m_1, m_2, \dots, m_J$  — натуральные числа и  $m_1 < m_2 < \dots < m_J$ ;

множество языков программирования

$$\Psi = (\Psi_d : d = 1, 2, \dots, D),$$

где  $d$  — порядковый номер языка;

множество алгоритмов

$$A = (A_l : l = 1, 2, \dots, L),$$

где  $l$  — порядковый номер алгоритма;

множество интервалов времени, отведенных на реализацию алгоритма на языке программирования,

$$\tau = (\tau_s : s = 1, 2, \dots, S),$$

где  $s$  — порядковый номер интервала.

Введем следующие обозначения:

$P_{\Psi_d, A_l, i, j, \tau_s}$  — программная реализация алгоритма  $A$  на языке  $\Psi_d$ , выполненная программистом  $i$  с уровнем мастерства  $j$  за время  $\tau_s$ ;

$V(P_{\Psi_d, A_l, i, j, \tau_s})$  — объем программной реализации алгоритма [1];

$V^*(P_{A_l})$  — потенциальный объем программной реализации алгоритма [1];

$R^k(P_{\Psi_d, A_l, i, j, \tau_s})$  — оценка надежности функционирования программной реализации на протяжении  $k$  прогонов;

$R^*(P_{A_l})$  — потенциальная надежность программной реализации алгоритма, т. е. оценка для абсолютно надежной программы, реализующей заданный алгоритм.

Для любой пары  $[(P_{\Psi_d, A_l, i, j, \tau_s}); k]$  частный удельный коэффициент изменения надежности программной реализации алгоритма устанавливается следующим образом:

$$\rho^k(P_{\Psi_d, A_l, i, j, \tau_s}) = \left| \frac{\overline{R^k(P_{\Psi_d, A_l, i, j, \tau_s})} - R^*P_{A_l}}{V(P_{\Psi_d, A_l, i, j, \tau_s}) - V^*(P_{A_l})} \right|.$$

Например, в терминах моделей Холстеда [1] и Нельсона [2]

$$\rho^k(P_{\Psi_d, A_l, i, j, \tau_s})$$

можно определить следующим образом:

$$\rho^k(P_{\Psi_d, A_l, i, j, \tau_s}) = \left| \frac{\exp\left[-\int_0^T h(t)dt\right] - 1}{N \log_2 \eta - [(2 + \tau_2^*) \log_2 (2 + \tau_2^*)]} \right|,$$

где  $T$  — суммарное время выполнения  $k$  первых прогонов программы  $P_{\Psi_d, A_l, i, j, \tau_s}$ ;

$$h(t_k) = -\frac{\ln(1 - Q_k)}{\Delta t_k},$$

где  $\Delta t_k$  — время выполнения  $k$ -го прогона;

$t_k$  — суммарное время выполнения  $k$  прогонов;

$Q$  — вероятность отказа программы на  $k$ -м прогоне;

$N$  — суммарное число всех операндов и операторов, имеющих в программной реализации алгоритма [1, с. 17];

$\eta_2^*$  — минимально возможное число отдельных операндов, которое может иметься в программной реализации алгоритма;

$\eta$  — число отдельных операторов и операндов, имеющих в программной реализации алгоритма.

Групповой удельный коэффициент надежности языка программирования  $\rho_{\Psi_d}(L, I, J, S)$  рассчитывается по формуле

$$\rho_{\Psi_d}(L, I, J, S) = \frac{\sum_{k=1}^K \sum_{l=1}^L \sum_{i=1}^I \sum_{j=1}^J \sum_{s=1}^S \rho^k(P_{\Psi_d, A_l, i, j, \tau_s})}{K \cdot L \cdot I \cdot J \cdot S}.$$

Обобщенный удельный коэффициент надежности языка программирования  $\rho_{\Psi_d}$  определяется следующим образом:

$$\rho_{\Psi_d} = \lim_{l \rightarrow \infty} \rho_{\Psi_d}(L, I, J, S).$$

Физический смысл удельного коэффициента надежности языка программирования заключается в том, что он показывает, как изменяется надежность программы на данном языке с изменением ее объема. Для практических целей может использоваться групповой коэффициент, который рассчитывается для конкретного множества алгоритмов при соответствующих ограничениях значений остальных входных параметров модели.

Соотношение показателей данной характеристики, полученных для различных языков (в частном случае, для выделенной группы алгоритмов, представляющих определенный класс) будет характеризовать степень пригодности того или иного языка для написания на нем надежных программ по сравнению с другими языками. В данном случае исследуется свойство самого языка с «зафиксированными» состояниями уровня мастерства программиста, продолжительности его работы, сложности реализуемого алгоритма, т. е. всего того, что (помимо самого языка программирования) влияет на степень подверженности программы отказам.

Описанный выше подход к оценке надежности языка программирования может использоваться при разработке новых языков и

при исследовании и оценке существующих. При этом должна ставиться задача не только оценки и выбора конкретного языка программирования, но и повышения надежности программ, написанных на данном языке. Для решения данной задачи необходимы новые инструментальные средства, отвечающие современному уровню развития аппаратного обеспечения и технологии программирования, облегчающие и удешевляющие разработку надежных программ. В качестве такого средства предлагаются корректирующие текстовые редакторы (КТР), которые совмещают в себе функции по созданию, контролю, диагностированию и коррекции текстов программ на конкретных языках программирования.

В процессе интерактивного ввода текста КТР анализирует программу в соответствии с рекомендациями по использованию различных конструкций языка (в зависимости от их пригодности для написания надежных программ), а также в соответствии с рекомендациями относительно рационального стиля и эффективных приемов программирования. КТР выдает диагностические, предупреждающие сообщения, рекомендации по альтернативной реализации отдельных фрагментов программы. Следует подчеркнуть, что функционально КТР не дублирует компилятор в отношении диагностирования. Например, он может не выполнять синтаксический контроль (это зависит от конкретных целей и условий применения в определенной операционной среде), но должен предупреждать разработчика о наличии чреватых ошибками на этапе функционирования фрагментов программы, которые формально могут быть абсолютно верными с точки зрения правил записи на данном языке.

Применение КТР позволяет, с одной стороны, использовать автоматизированные процедуры контроля и повышения надежности программы на стадиях разработки, предшествующих этапу компиляции (целесообразность чего обоснована [3]), а с другой — избежать дополнительного этапа разработки, так как качественно новые средства и процедуры вводятся на традиционных этапах разработки. Кроме того, предлагаемое инструментальное средство является не только корректирующим, но и фактически обучающим, т. е. не только улучшает качество программ, но и повышает профессиональное мастерство программистов.

В настоящее время разрабатывается система, в которой использован описанный в настоящей статье подход к оценке надежности языка программирования. Результаты применения системы на практике будут опубликованы позже.

#### ЛИТЕРАТУРА

1. Холстед М. Х. **Начала науки о программах.** — М.: Финансы и статистика, 1981.
2. Тейер Т., Липов М., Нельсон Э. **Надежность программного обеспечения.** — М.: Мир, 1981.
3. Майерс Г. **Надежность программного обеспечения.** — М.: Мир, 1980.

# IV

## Применение средств вычислительной техники

---

УДК 65.015.12:681.3

### РАБОЧИЕ МЕСТА ДЛЯ КОНСТРУКТОРОВ И ТЕХНОЛОГОВ НА ОСНОВЕ МИКРОЭВМ СМ-1630

Б. Луль, канд. техн. наук (ГДР)  
В. Реллер, инженер (ГДР)

Возрастающая степень автоматизации производства требует усиленного использования ЭВМ на рабочих местах для эффективной и рациональной технической подготовки производства. В ГДР в зависимости от места в процессе технической подготовки производства типовые рабочие места, построенные на базе СМ-1630, различаются по виду ввода-вывода: алфавитно-цифровой, пассивно графический, интерактивно графический. Типовые рабочие места содержат все приведенные в табл. 1 компоненты и имеют единую модульную структуру.

Для программного обеспечения типовых рабочих мест в унифицированном виде определены приведенные в табл. 1 интерфейсы (в соответствии с международной тенденцией к стандартизации). Графическое базовое программное обеспечение уровней 2, 3 и 4 создает предпосылку для аппаратно независимой разработки и применения пользовательских программ. Уровень 2 служит для выравнивания «интеллекта» между разными графическими устройствами, уровень 3 поддерживает графический ввод-вывод, уровень 4 — обработку графических структур и специфическое для проблемы ведение диалога.

Для пользователя очень важно то, что его программы основываются на независимых от устройств уровнях графики (уровни 3 и 4 в табл. 1). Изменение или обновление конфигурации не отражается на пользовательских программах.

В табл. 2 приведены аппаратные компоненты основных видов конфигурации рабочих мест на базе микроЭВМ СМ-1630.

Основная конфигурация 1 — диалоговое рабочее место без интерактивной графики для специальных случаев применения (проектирования печатных плат, создания документов и носителей данных).

Основная конфигурация 2 — диалоговое рабочее место без интерактивной графики. Диалоговая работа возможна лишь в алфа-



витно-цифровом режиме. На растровом дисплее предусмотрено пассивное графическое изображение.

Основная конфигурация 3 — диалоговое рабочее место для конструкторов и технологов, имеющее более низкую производительность по графическому изображению и манипуляции.

Таблица 1

**Компоненты рабочих мест на основе ВМ**

| Компоненты  |   | Уровни ПО с определенными интерфейсами |
|---|---|--|
| Применение пользовательских программ                              |   |  |
| Микроэлектроника  | Машиностроение  |  |
| Общее количество пользовательских программ                        |   |  |
|   | Геометрические модули (взаимодействие), поддерживающее программное обеспечение, конструкция и технология                                  | 5                                      |
| Структурное и диалоговое ядра                                     | графическое программное обеспечение, независимое от устройств   | 4                                      |
| Графическое ядро  |   | 3                                      |
| Специфический для устройств графический уровень                   |   | 2                                      |
| Операционная система MOOS 1600                                    | Проблемно-ориентированное программное обеспечение общего назначения (ПО банка данных, компиляторы, интерпретаторы, математические методы) | 1                                      |
| Программы обслуживания общих и графических периферийных устройств |   | 0                                      |
| Пассивная и интерактивная графическая периферия                   | Технические средства СМ ЭВМ или ЕС ЭВМ  |  |

Основная конфигурация 4 — диалоговое рабочее место, удовлетворяющее максимальным требованиям за счет подключения к микроЭВМ СМ-1630 интерактивного графического терминала с цветным изображением на большом векторном дисплее.

Из табл. 2 и из краткого описания следует, что основные конфигурации совместимы снизу вверх. Их можно использовать в качестве терминалов в рамках ЕС ЭВМ. Выявляется возможность предоставления разных проблемно-ориентированных комплексов путем применения аппаратной техники с гибкой конфигурацией и

оснащением базовым и пользовательским программным обеспечением [2].

Комбинат «Роботрон» поставляет «Рабочее место для конструирования печатных плат robotron A 6452», имеющее основную конфигурацию 1, и «Рабочее место для конструкторов и технологов robotron A 6454», имеющее основную конфигурацию 3.

Таблица 2

Устройства основных конфигураций на основе СМ-1630

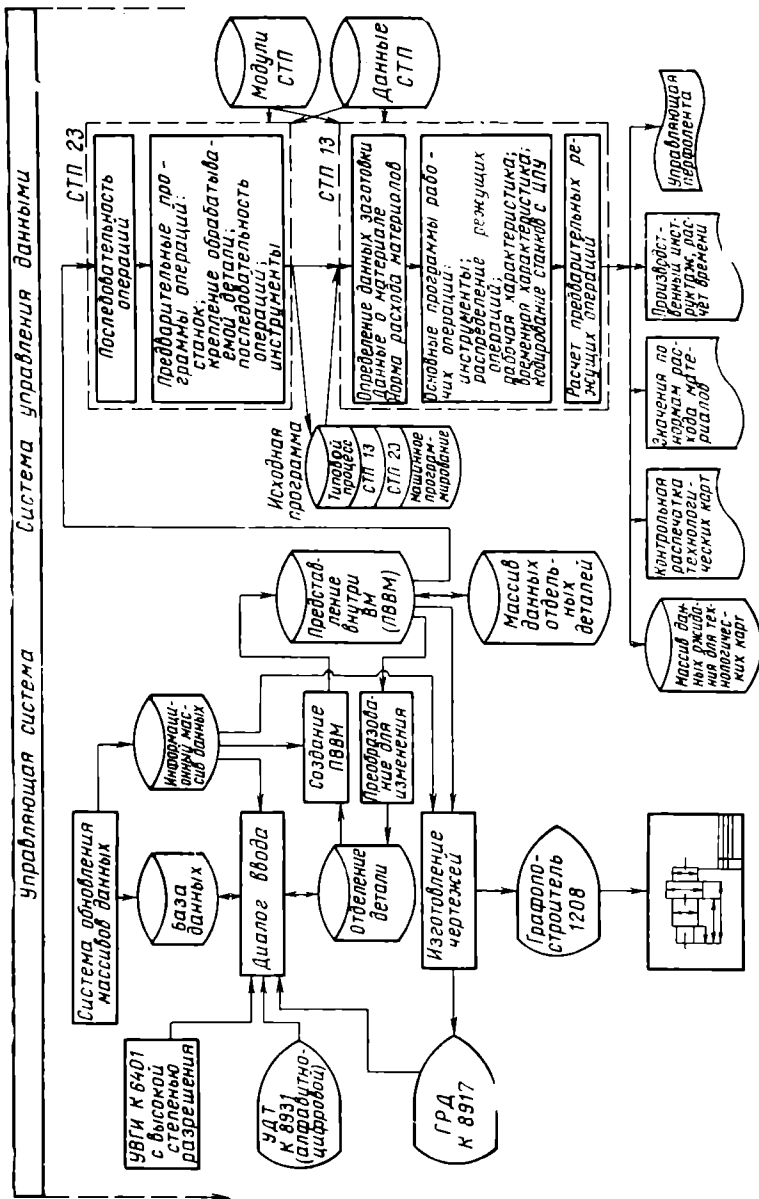
| Устройства  | Основные конфигурации |   |   |   |
|---|-----------------------|---|---|---|
|   | 1                     | 2 | 3 | 4 |
| МикроЭВМ СМ-1630  | ×                     | × | × | × |
| Общая периферия СМ-1630   | ×                     | × | × | × |
| пульт оператора   | ×                     | × | × | × |
| ВЗУ (накопители на магнитных дисках, магнитных лентах, на гибких дисках)                                | 0                     | 0 | 0 | 0 |
| печатающее устройство (параллельного и последовательного действий)                                      | 0                     | 0 | 0 | 0 |
| перфоленточное устройство   | 0                     | 0 | 0 | 0 |
| дисплейный терминал   | 0                     | 0 | 0 | 0 |
| далее см. [1]   |                       |   |   |   |
| Периферия для выполнения графической работы   |                       |   |   |   |
| графический растровый дисплей   | ×                     |   |   |   |
| графопостроитель (планшетный графопостроитель, чертежное устройство с цифровым программным управлением) | 0                     |   | 0 | 0 |
| устройство для изготовления твердых копий   |                       |   |   | × |
| устройство ввода графической информации   |                       |   | 0 | 0 |
| дисплейный, алфавитно-цифровой терминал   |                       | 0 | 0 | 0 |
| растровый дисплей   |                       | × | × | × |
| интерактивный графический терминал  |                       |   |   | × |

× — устройство является составной частью основной конфигурации;

0 — устройство является составной частью основной конфигурации, тип и количество зависит от проблемно-ориентированного комплекса.

В [3] информируется о рабочем месте для конструирования печатных плат А 6452 и о входящей в его состав системе проектирования печатных плат.

**Рабочее место для конструкторов и технологов** (см. рисунок) представляет собой систему устройств CAD/CAM (CAD — Computer Aided Design, CAM — Computer Aided Manufacturing), позволяющую значительно повысить производительность в процессах конструкторской и технологической подготовки производства (в отдельных процессах — в 6 раз, в среднем — в 3 раза, по исследованиям в машиностроении). А 6454 представляет собой модульную



Вариант применения САD/CAM «POTA-TEILE»:

СТП — система технологических программ; УВГИ К 6401 — устройство ввода графической информации К 6401; УДТ — удаленный диалоговый терминал; ГРД К 8917 — графический дисплей К 8917

систему и содержит все приведенные в табл. 1 компоненты, кроме программного обеспечения уровня 4. Ядром А 6454 является микроЭВМ СМ-1630 с оперативной памятью 128 или 256 Кбайт. К ЭВМ в зависимости от требований пользователей могут подключаться обычные периферийные устройства.

В качестве графических устройств могут использоваться графический растровый дисплей К 8917, устройство ввода графической информации К 6401 с высокой разрешающей способностью и планшетный графопостроитель «Дигиграф» или графопостроитель размером 90×120.

Вид и количество периферийных устройств, подключаемых в конкретном случае применения, зависят от требований пользователя. При емкости оперативной памяти 256 Кбайт можно работать максимально с четырьмя станциями (дисплейными рабочими местами или местами ввода графической информации). Сопряжения А 6454 с другими ЭВМ (например ЕС-1055 М как вычислитель на заднем фоне) могут быть реализованы через предусмотренные для системы микроЭВМ СМ-1630 интерфейсы.

Основными областями использования А 6454 являются: получение конструктивных вариантов; расчет деталей; детализовка изделий с помощью ЭВМ; создание и вывод с помощью ЭВМ чертежей; получение с помощью ЭВМ технологических карт и других технологических документов; программирование с помощью ЭВМ станков с цифровым программным управлением; проектирование с помощью ЭВМ машин и установок. Кроме того, А 6454 пригодна для решения задач проектирования в области строительства. А 6454 обеспечивает графический ввод и вывод, графическое изображение и графические манипуляции для разных случаев применения при сравнительно низких требованиях к графическому диалогу.

В основной комплект А 6454 входит существующее для системы микроЭВМ-1630 системное и стандартное программное обеспечение, описанное подробно в приведенной литературе: модульная операционная система MOOS 1600 [4]; проблемно-ориентированное программное обеспечение общего назначения [5, 6]; проблемно-ориентированное программное обеспечение для разработки программ [7].

Для А 6454 предоставляется следующее базовое программное обеспечение: графическая базовая система GKS 1600; базовое программное обеспечение для преобразования графической информации в цифровую DIG 1600; геометрическое базовое программное обеспечение (геометрический модуль) GBS 1600.

Прикладное программное обеспечение должно быть разработано пользователем с учетом специфических для отрасли и предприятия условий. В ГДР для первого использования разрабатываются и обобщаются математические модели и алгоритмы конструирования и технологии объектов и процессов [2].

Табл. 3 показывает модель уровней программного обеспечения для А 6454 и его связь с операционной системой.

**GKS 1600** — система программ для обработки графического вывода на графопостроитель и графические дисплеи и для графиче-

ческого ввода с графических дисплеев. Объем функций GKS 1600 соответствует приведенному в проекте стандарта ИСО версии 7.0 объему [8]. Функции графической базовой системы GKS 1600 предоставлены в распоряжение программиста-пользователя в виде подпрограммы на Фортране [9, 10]. GKS 1600 дает программисту-пользователю возможность с помощью одних и тех же инструкций обращаться к устройствам разных типов (графопостроители, графические дисплеи) и разных моделей. В результате экономится

Таблица 3

Модель слоев программного обеспечения robotron A 6454

|   |                         |
|---|-------------------------|
| Прикладное программное обеспечение  |                         |
| GKS 1600  | независимо от устройств |
| в зависимости от устройств (специфический по устройствам графический уровень) |                         |
| модуль для устройства 1 . . . . .   | модуль для устройства n |
| Операционная система MOOS 1600  |                         |
| Программы обслуживания периферийных устройств                                 |                         |
| графических   | общих                   |
| устройство 1 . . . . .  | устройство n            |

труд программистов-пользователей (в особенности при изменении технических средств графики).

**DIG 1600** — пакет программ, позволяющий пользователю работать на устройстве ввода графической информации К-6401 без дополнительного программирования. Данные вводятся в последовательный массив данных в структуре, позволяющей дальнейшую обработку под управлением графической базовой системы GKS 1600 (например, вывод на графопостроитель) [9, 10].

**GBS 1600** — интерактивная графическая система, реализующая обработку геометрических двухкоординатных изображений объектов и тем самым непосредственно поддерживающая работу конструкторов и технологов. Одна из важных конструкторских работ состоит в проектировании изделий, узлов и деталей с помощью повторного применения конструктивных элементов. Процесс конструирования с помощью ЭВМ на А 6454 при наличии GBS 1600 реализуется в диалоговом режиме через алфавитно-цифровой и

графический дисплей. Графический дисплей выводит мгновенное состояние обработки. Поэтому можно наглядно выполнить и проверить дополнения и изменения. Размеры указываются автоматически и дополняются или проверяются в интерактивном режиме на графическом дисплее. Конструктор освобождается от рутинных процессов и может сосредоточиться на творческой деятельности. Он получает быстрее желаемый вариант или может создать несколько вариантов, из которых выбирается самый лучший. После завершения разработки возможен автоматический вывод чертежей на графопостроитель [9, 10]. GBS 1600 может быть использована вместе с GKS 1600 и DIG 1600. Функции и условия применения GKS 1600, DIG 1600 и GBS 1600 подробно описаны в [10].

В случае использования А 6454 исключительно в виде рабочего места для технологов применяется уменьшенный в соответствии с основной конфигурацией вариант 2. При этом ориентация на работу технологов только задается соответствующими пользовательскими программами, например программами для разработки технологических карт и производственных инструкций, для определения технологических мощностей, для проведения информационного поиска и для создания управляющих программ для станков с числовым программным управлением.

#### ЛИТЕРАТУРА

1. Сикор М. Новые технические средства и системное программное обеспечение СМ ЭВМ из ГДР. — В кн.: Вычислительная техника социалистических стран. Вып. 11. — М.: Финансы и статистика, 1982.
2. Internationaler Kongress Metallbearbeitung (IKM 82). Leipzig, Tagungsmaterial, 1982.
3. Роевский К., Фогель Л. Рабочее место конструктора печатных плат АКТ 30.1. — В кн.: Вычислительная техника социалистических стран. Вып. 14. — М.: Финансы и статистика, 1983.
4. Фойгт К.-У. Модульная операционная система МООС 1600 для СМ-1630. — В кн.: Вычислительная техника социалистических стран. Вып. 14. — М.: Финансы и статистика, 1983.
5. Hientzsch E. u. a. Problemorientierte Systemunterlagen für das Mikrorechnergerätesystem robotron K 1600. — Übersicht und DATO 1600. Neue Technik im Büro, Berlin, 26 (1982), 5.
6. Виггерт К.-Х. Программное обеспечение управления данными для микроЭВМ СМ-1630. — В кн.: Вычислительная техника социалистических стран. Вып. 15. — М.: Финансы и статистика, 1984.
7. Хорн Э., Тач Б. Технологическая система для разработки программного обеспечения для микроЭВМ СМ-1630 (robotron K 1600). — В кн.: Вычислительная техника социалистических стран. Вып. 14. — М.: Финансы и статистика, 1983.
8. Draft International Standard ISO/DIS Information Processing Graphical Kernel System (GKS) Funktional. Description ISO TC 97/SC5/WG2 Nr. 117, Januar 1982.
9. Baier, Mikut, Reller. Grafische Grundsystemunterlagen des АКТ А 6454. Edv-aspekte. Berlin, 1/82.
10. Autorenkollektiv VEB Robotron ZFT. Arbeitsplatz für Konstruktion und Technologie АКТ А 6454. Berlin, Anwenderinformation, 1983.

**ПРОБЛЕМНО-ОРИЕНТИРОВАННЫЙ КОМПЛЕКС  
АВТОМАТИЗИРОВАННЫХ РАБОЧИХ МЕСТ  
ДЛЯ ПРОГРАММИРОВАНИЯ И ОТЛАДКИ МИКРОПРОЦЕССОРНЫХ  
УСТРОЙСТВ АРМ2-05**

**С. С. Забара**, д-р техн. наук (СССР)  
**А. Д. Мильнер**, канд. техн. наук (СССР)  
**А. В. Богачев**, инженер (СССР)

Комплекс АРМ2-05 служит для подготовки программ и микропрограмм для микропроцессорных устройств (МПУ), встраиваемых в различное промышленное оборудование (станки, периферийные устройства ЭВМ, терминалы, средства связи, технологические контроллеры) и в специализированные системы управления, работающие в реальном масштабе времени.

Применение МПУ как составной части промышленного изделия в качестве элемента, работающего в контуре управления процессами, характеризуется рядом особенностей, к которым относятся следующие:

к МПУ в целом и его программному (микропрограммному) обеспечению в частности предъявляются повышенные требования по эффективности и надежности;

требование эффективности МПУ при ограниченной эффективности базовых микропроцессорных элементов делает целесообразным совместное проектирование программ (микропрограмм) и определенной части аппаратуры, комбинированное использование программного и микропрограммного уровней, а также построение мультимикропроцессорных структур;

из требования эффективности вытекают ограничения на конструктивные и функциональные ресурсы для построения полноценных резидентных средств автоматизации программирования и микропрограммирования;

существенное место в программном (микропрограммном) обеспечении занимают процедуры логического (невыхислительного) характера, включая внутренние пересылки данных, операции над битами, анализ и обработку различных условий и т. п.;

значительная часть программ и микропрограмм хранится в постоянных запоминающих устройствах (ПЗУ);

проектирование программ и микропрограмм для МПУ осуществляется, как правило, непрофессиональными программистами; разнообразие применений МПУ и специфичность каждого из объектов управления делают затруднительными использование в широких масштабах ранее разработанных программ (даже если они были разработаны для аналогичных изделий);

промышленный характер применения МПУ заставляет подходить к его программному (микропрограммному) обеспечению как

к изделию, что ведет к более строгой регламентации различных сторон жизни такого изделия.

Указанные особенности оказывают существенное влияние на технологию проектирования МПУ и на архитектуру систем автоматизации его проектирования.

К настоящему времени известно большое количество отечественных и зарубежных систем и отдельных средств автоматизации разработки программного обеспечения МПУ [1—4].

Все многообразие этих средств можно условно разбить на три класса: отладочные комплексы, специализированные системы проектирования, универсальные аппаратно-программные комплексы проектирования. Основным признаком предлагаемого разбиения является состав технических средств и программного обеспечения.

**Отладочные комплексы.** Основой отладочного комплекса является микроЭВМ, реализованная на микропроцессорном наборе того же типа, что применяется в проектируемом МПУ. В качестве устройств ввода-вывода может применяться простая клавишная панель с индикацией или телетайп. Емкость оперативной памяти для отлаживаемых программ и данных может достигать 64 Кбайт.

Программирование при использовании отладочных комплексов ведется либо в машинных командах с вводом с клавиатуры, либо на ассемблере с загрузкой с перфоленты (кассетной магнитной ленты).

Отладочные комплексы предназначены в первую очередь для оценки возможностей микропроцессорного набора и отладки программ объемом до 4—8 Кбайт.

**Специализированные системы проектирования.** Такая система является развитием отладочного комплекса и представляет собой двухуровневую систему: первый уровень включает микроЭВМ для управления средствами проектирования общего назначения, ориентированными на микропроцессоры, а второй — представляет собой специализированный модуль — внутрисхемный эмулятор, который обеспечивает эмуляцию и диагностику конкретного микропроцессора.

При комплексной отладке аппаратного и программного обеспечения проектируемого МПУ система проектирования подключается к МПУ посредством кабеля модуля эмуляции. Эмуляция функций управления микропроцессора осуществляется инструментальным микропроцессором, размещенным в модуле эмуляции и являющимся полным аналогом микропроцессора проектируемого МПУ. Модуль эмуляции выполняет все отладочные операции (сравнение адресов, условий, подсчет времени и т. д.) аппаратно, что не требует дополнительного времени работы монитора системы, и отлаживаемая программа выполняется в реальном масштабе времени.

Характерной особенностью систем проектирования, кроме внутрисхемной эмуляции, является возможность программирования не только на ассемблере, но и на языках высокого уровня, например ПЛ/М. Указанные возможности обеспечиваются широким набором



внешних устройств (дисплеи, накопители на гибких магнитных дисках, печатающие устройства) и наличием развитой операционной системы.

**Универсальные комплексы проектирования.** В последнее время широкое применение находят универсальные аппаратно-программные комплексы, предназначенные для проектирования МПУ на основе восьми- и шестнадцатиразрядных микропроцессорных наборов.

Они предоставляют пользователю все возможности ранее рассмотренных систем, но в отличие от последних обеспечивают работу с различными типами микропроцессоров. Такая универсальность достигается наличием мощной центральной части с возможностью смены модулей эмуляции и некоторой части прикладного программного обеспечения.

Характерной особенностью таких комплексов является наличие в их составе нескольких терминалов, подключенных к центральной части и использующих ее ресурсы. Терминалы имеют, как правило, собственный процессор, память и средства диалогового взаимодействия пользователя с системой проектирования. В состав терминала включается модуль эмуляции, определяющий тип микропроцессора проектируемого МПУ.

Программное обеспечение терминала достаточно развитое и позволяет пользователю на своем рабочем месте отлаживать программы.

В состав центральной части комплекса входит мощный процессор (как правило, мини-ЭВМ) с оперативной памятью емкостью до 256 Кбайт. В качестве внешней памяти используются накопители на сменных магнитных дисках. Комплекс имеет быстродействующие печатающие устройства.

Универсальным аппаратно-программным комплексом проектирования МПУ является и рассматриваемый нами комплекс АРМ2-05.

Универсальность комплекса АРМ2-05 поддерживается модульным принципом построения в функциональном и конструктивном плане. Инструментальные средства базового комплекса обеспечивают его настройку на различные микропроцессорные наборы. Под настройкой понимаются разработка и подключение прикладных аппаратных и программных модулей, определяющих целевые условия применения (тип микропроцессора, структуру МПУ, языки программирования, выходные документы). Трудоемкость настройки составляет от 5 до 20% трудоемкости создания специализированной системы проектирования.

Отличительной особенностью комплекса является возможность его применения для целевого проектирования программного и микропрограммного обеспечения МПУ.

Вторая особенность комплекса — свойство открытости — позволяет, пользуясь базовыми инструментальными средствами, создавать пакеты прикладных программ, расширяющие функциональные возможности системы проектирования.

Третьей отличительной чертой комплекса является многопультная работа, что достигнуто благодаря значительному повышению «интеллекта» рабочих мест.

Комплекс предусматривает выход через адаптер межмашинной связи на ЭВМ более высокого уровня (ЕС ЭВМ), что обеспечивает подключение комплекса АРМ2-05 к более мощным иерархическим САПР.

Кроме этих основных особенностей можно отметить следующие функциональные характеристики комплекса.

**Многоязыковость.** Реализован базовый язык Микросленг; используются языки разного уровня (Ассемблер, Бейсик, ПЛ/М, Микрод); выполняется предварительный контроль; применяется русская символика.

**Интерактивность.** Заложена инженерная ориентация диалога; используются общие изобразительные средства; реализовано назначение функциональных клавиш; осуществляется выдача «меню»; выполняется расшифровка сообщений и «твердая» копия; выполняется полноэкранный редакция; используются макродирективы отладки.

**Отладочный сервис.** Используется символьная отладка и локальная коррекция; реализована внутрисхемная эмуляция и программное моделирование; обеспечена совместная отладка программ и аппаратуры в реальном времени; имеется шесть отладочных режимов и большое количество точек прерывания; трассировка обеспечивает отслеживание состояния микропроцессора и шин адреса и данных.

**Документирование.** Возможность записи отлаженных программ на любом системном носителе; наличие непосредственного программирования БИС ППЗУ (электрическое прожигание); учет стандартов ЕСКД и ЕСПД; синхронная коррекция во всех массивах; сопровождение документами.

Конфигурация комплекса АРМ2-05 представлена на рис. 1.

Комплекс представляет собой двухуровневую систему, на верхнем уровне которой работает базовый комплекс, реализованный на основе УВК СМ-4.

На нижнем уровне находятся рабочие места — терминалы, служащие для интерактивного взаимодействия пользователя с системой проектирования и проектируемым МПУ.

Базовый комплекс включает в себя оперативную память емкостью 256 Кбайт и внешние запоминающие устройства: ВЗУ на магнитных дисках СМ-5402.09; ВЗУ на сменных магнитных дисках СМ-5407.01 и ВЗУ на магнитной ленте СМ-5301.10.

В состав базового комплекса входят системные устройства ввода-вывода: перфоленточное СМ-6202.01, параллельное печатающее СМ-6305, печатающее СМ-6300.01, видеотерминал ВТА 2000-2, «интеллектуальный» видеотерминал ВТА 2000-13. Для связи с ЭВМ серии ЕС в состав базового комплекса включено устройство связи вычислительных машин А711-16.

Терминал подготовки программ (рис. 2) выполнен на основе микроЭВМ СМ-1803.05, в которой подключается специальное устройство эмуляции и связи с макетом (УЭСМ). В состав модели СМ-1803.05 входят базовая микроЭВМ (модуль центрального процессора и модуль системного контроля), модуль оперативной памяти 64 Кбайт, видеотерминал ВТА 2000-30, ВЗУ на гибких магнитных дисках (емкостью 512 Кбайт), пульт-программатор для зане-

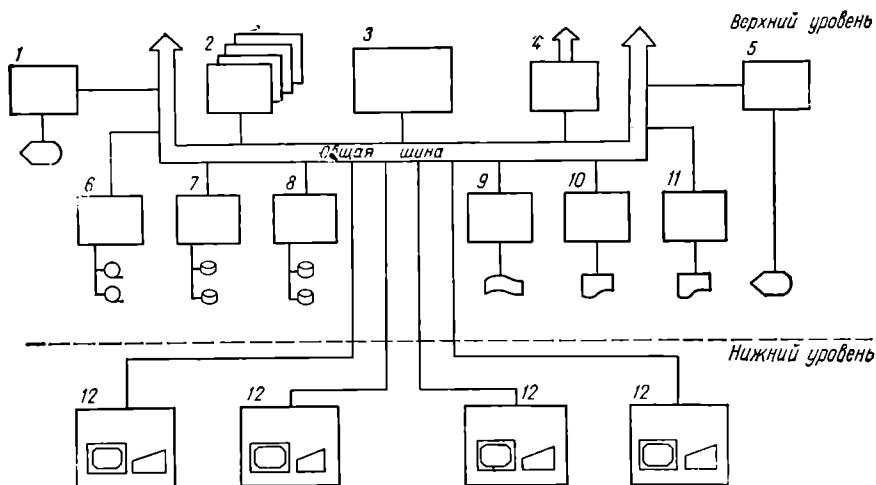


Рис. 1. Структурная схема комплекса АРМ2-05:

1 — ВТА 2000-2; 2 — ОЗУ; 3 — процессор СМ-2401; 4 — УСВМ А711-16; 5 — ВТА 2000-13;  
6 — УВП МЛ СМ-5301.10; 7 — УВП МД СМ-5402.09; 8 — УВП СМД СМ-5407.01; 9 — УВВ  
ПЛ СМ-6202.01; 10 — АЦПУ СМ-6305; 11 — АЦПУ СМ-6300.01; 12 — терминал подготовки  
программ

сения информации в микросхему К 556 РТ4, устройство последовательной печати, модуль связи с УВК СМ-4.

Устройство эмуляции и связи с макетом предназначено для подключения проектируемого МПУ к микроЭВМ СМ-1803.05 и эмуляции микропроцессора и управляющей памяти.

Комплекс может работать в двух режимах: в режиме 1 — разработка прикладных программ, настраивающих комплекс на проектирование конкретных (целевых) МПУ, и в режиме 2 — программирование и отладка целевых МПУ. Основная группа пользователей применяет комплекс в режиме 2. Для этой группы он является инструментом проектирования, и именно с этой точки зрения рассмотрим применение потребительского комплекса САПР на основе АРМ2-05.

**Технология целевого проектирования.** Комплекс представляет собой совокупность нескольких индивидуальных рабочих мест с общими системными ресурсами и общей базой данных. Каждое рабочее место оснащено видеотерминалом, накопителем на гибком магнитном диске, программатором и устройством эмуляции и связи с макетом, настроенным на конкретный микропроцессорный на-

бор. Рабочее место не закрепляется за конкретным пользователем, и перед началом работы пользователь должен идентифицировать себя и проектируемое МПУ (либо программу).

На первом этапе проектирования осуществляется ввод модуля с исходным текстом программы либо с клавиатуры дисплея, либо с персональной дискеты, предварительный контроль модуля и запись его в библиотеку системы.

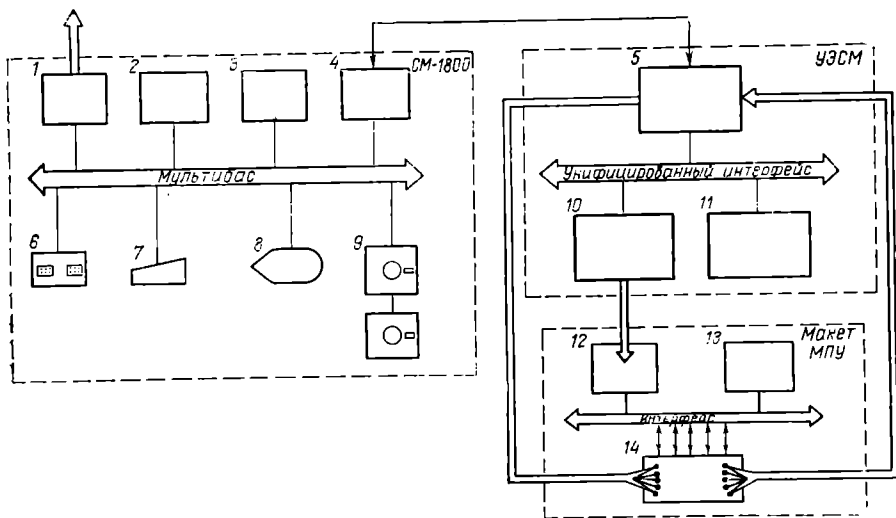


Рис. 2. Структурная схема терминала подготовки программ:

1 — модуль связи с ОШ; 2 — ОЗУ; 3 — процессор SM-1800; 4 — адаптер УЗСМ; 5 — модуль связи с МПУ; 6 — программатор; 7 — клавиатура; 8 — дисплей; 9 — накопитель на гибких магнитных дисках; 10 — модуль эмуляции микропроцессора; 11 — модуль эмуляции управляющей памяти; 12 — целевой микропроцессор; 13 — целевая управляющая память; 14 — логические схемы МПУ

После трансляции, контроля и редактирования выполняется (при необходимости) второй этап — отладка. Комплекс обеспечивает два уровня отладки:

автономную (статическую) отладку модуля (или группы модулей) программным моделированием проектируемого МПУ. Управление отладкой осуществляется в диалоговом режиме или с помощью сформированных тестов. Доступ к отлаживаемой программе осуществляется в терминах исходного языка программирования;

комплексную (динамическую) отладку при подключении к системе макетного (опытного) образца отлаживаемого МПУ. Подключение осуществляется через устройство эмуляции и связи с макетом посредством установки кабельного разъема на место микропроцессора или блока микропрограммного управления в МПУ.

Как при автономной, так и при комплексной отладке пользователь может запускать программу с различных точек, останавли-

вать ее выполнение по заданным условиям, определять и изменять значения переменных, моделировать внешние воздействия, корректировать программу, отслеживать ход (трассу) ее выполнения.

Третьим этапом проектирования является выпуск технологических документов или машинных носителей, которые могут быть переданы в другие системы проектирования (например, САПР для БИС ПЗУ).

Если управляющая память проектируемого МПУ реализуется на ППЗУ серии К 556, то с помощью программатора можно непосредственно на рабочем месте осуществить занесение программных кодов в микросхемы памяти.

Последним этапом проектирования в системе является выпуск комплекта эксплуатационной документации, оформленной в соответствии с требованиями ЕСКД, ЕСПД и ЕСТД. Кроме того, обеспечивается сопровождение программного изделия путем коррекции целевых программ, находящихся в библиотеке системы, с выпуском извещений на изменения в документации или перевыпуском комплекта программной и технологической документации.

#### Технические данные

|   |   |
|---|---|
| Базовая ЭВМ   | УВК СМ-1407.01 (СМ-4)   |
| Количество рабочих мест                               | до 4  |
| Тип рабочего места                                    | терминал подготовки программ (ТПП)  |
| Связь ТПП с СМ-4                                      | ИРПС  |
| Удаление ТПП, м                                       | до 500  |
| Скорость обмена данными между ТПП и СМ-4, бод         | до 9600   |
| Метод подключения отлаживаемого МПУ к ТПП             | внутрисхемная эмуляция; прямое управление   |
| Количество дополнительных точек подключения МПУ к ТПП | до 48   |
| Режим отладки   | пошаговое выполнение программы; непрерывное; до заданной контрольной точки; циклическое; программная трассировка; доступ к переменным регистрам |
| Базовые операционные системы                          | ОС РВ СМ-4; СПО СМ-1800   |
| Проблемно-ориентированное программное обеспечение     | БПО АРМ2-05   |
| Занимаемая площадь, м <sup>2</sup>                    | 80  |

**Развитие.** Заложенный при разработке принцип универсальности требовал настройки комплекса на многие типы микропроцессоров. Модульная структура в функциональном и конструктивном плане позволила выделить сменную часть (модуль), учитывающую целевые условия применения конкретного микропроцессорного набора. Этот модуль представляет собой аппаратные блоки, содержащие внутрисхемный эмулятор микропроцессора и управляющую память конкретного МПУ, и программное обеспечение, ориентированное на этот микропроцессор. Сменный модуль оформлен в виде отдельного изделия, и включение его в комплекс осуществляется установкой аппаратных блоков на предусмотренные свободные ме-

ста в устройстве эмуляции и связи с макетом и загрузкой прикладной программы с дискеты.

В первую очередь комплекса АРМ2-05 включены сменные модули, настроенные на микропроцессорные серии К 580 и К 589. В дальнейшем предусмотрены модули, настроенные на серии К 586 и К 1804.

Для комплекса АРМ2-05 можно выделить несколько основных составляющих, обеспечивающих его экономическую эффективность.

Во-первых, распределенная обработка информации с использованием четырех рабочих мест позволяет сократить капитальные и эксплуатационные расходы на одно рабочее место на 40—60% по сравнению с аналогичными показателями других систем проектирования.

Во-вторых, наличие проблемно-ориентированных аппаратных и программных средств комплекса АРМ2-05 позволяет сократить затраты, связанные с разработкой потребительских комплексов, настроенных на один конкретный микропроцессорный набор, в 1,5—2 раза.

В-третьих, использование комплекса АРМ2-05 в проектной организации позволяет расширять номенклатуру применяемых микропроцессорных наборов без приобретения новых систем проектирования, а лишь путем допоставки и установки в комплекс сменного модуля, стоимость которого составляет менее 5% общей стоимости системы проектирования.

#### ЛИТЕРАТУРА

1. Рауд Р. К., Тамм Б. Г. Состояние в области программирования для микроЭВМ. — Программирование, 1982, № 5.
2. Липаев В. В. Состояние и проблемы производства программного обеспечения для систем управления и обработки информации. — Управляющие системы и машины, 1980, № 1.
3. Santoni A. Microcomputer Development Systems. — EDN, 1980, Sept, N 5.
4. Krieger M. Microcomputer-Entwicklungssystem Mit Echtzeit-Emulation-Elektronik-Report, 1980, vol. 5, N 3.

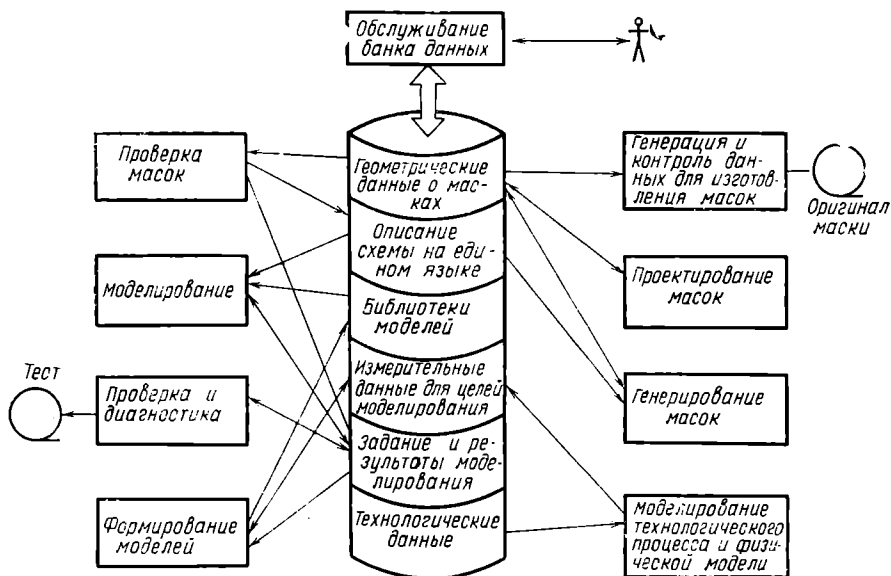
УДК 621.3.049.77:02.57

#### АВТОМАТИЗИРОВАННОЕ ПРОЕКТИРОВАНИЕ БИС И СБИС В ЧССР

**И. Пилуха**, инженер (ЧССР)  
**П. Глоушек**, канд. техн. наук (ЧССР)  
**Л. Нойманн**, инженер (ЧССР)  
**Я. Пейчох**, канд. техн. наук (ЧССР)

Чехословацкая программа развития микроэлектроники в 7-й пятилетке (1981—1985 гг.) предусматривает выпуск нового ряда интегральных схем (ИС), новых типов

гибридных схем и освоение производства интегральных схем с очень высокой степенью интеграции. Проектирование и производство таких схем немислимо без комплексного использования вычислительной техники и автоматизации отдельных этапов проектирования ИС. Хотя в ЧССР имеется ряд эффективных программ, которые можно использовать на некоторых этапах проектирования ИС, между этими программами нет никакой взаимосвязи. Это значит,



Структура программной системы для проектирования СБИС

что при переходе с разных уровней моделирования, анализа и контроля огромное количество данных преобразуется вручную и высока вероятность ошибок. Практический опыт показывает, что собственно процесс проектирования ИС в результате часто обесценивается. Для создания сверхбольших интегральных схем (СБИС) имеющиеся программные средства недостаточны и в значительной степени ограничивают скорость проектирования и сложность проектируемых схем.

Для обеспечения намеченных целей в электротехнической промышленности ЧССР создано в 1981 г. при НИИ техники связи им. А. С. Попова в Праге головное подразделение для проектирования ИС. Основные его задачи:

создать концепцию развития программных средств;

обеспечивать создание программных средств, применимых на доступных ЭВМ, и выбранных графических систем для проектирования СБИС с интеграцией от нескольких сотен тысяч до миллиона элементов их на чипе до 1985 г. по всем типам технологии (биполярной и МОП);

координировать разработку программного обеспечения и методов проведения работ в области проектирования ИС в рамках министерства электротехнической промышленности;

создать центральную библиотеку программ и управлять использованием программных средств для автоматизированного проектирования ИС;

проработать модель унифицированного обеспечения рабочих мест автоматизированного проектирования средствами вычислительной техники.

**Программная система для проектирования СБИС.** Концепция создаваемой программной системы строится на том, что данная система должна быть комплексной, емкой, достаточно определенной и интегрированной, чтобы исключить неблагоприятное влияние человеческого фактора при подготовке данных, являющегося причиной ошибок и неточностей. Структура создаваемой системы показана на рисунке.

Система выполнена как открытая с модульной структурой, в которой отдельные модули, представляемые самостоятельными программами или их комплектами, можно постепенно дополнять или обновлять и приспосабливать ее таким образом к меняющимся требованиям. Модульность достигается благодаря концентрации всех данных, касающихся проектирования, в жестко установленном основном массиве данных. Посредством этого массива осуществляются все связи между программами. Через него поступают также основные первичные данные, задаваемые человеком. Это позволяет создать систему контроля данных в процессе проектирования и свести к минимуму неблагоприятное влияние человеческого фактора. Сосредоточение данных в основном массиве, где определены связи между отдельными компонентами, дает возможность разработчику схем получать наглядную информацию.

Система создается постепенно. В ней использованы ранее созданные программы, которые в случае потребности перерабатываются для включения в систему, и создаются новые программы. В настоящее время существует уже ряд программ — частей системы. Это, например, программа МОНЕТ, включенная в подсистему, обозначенную на рисунке как «Проверка проектирования масок». Программа МОНЕТ позволяет осуществлять контроль предложенной морфологии схемы перед ее производством таким образом, что из графических данных, описывающих маски схемы, создается описание электрической схемы с учетом и паразитных емкостей корпуса. Программа предназначена для проектирования схем в технологии КМОП с металлическим затвором или же *n*-МОП с металлическим затвором и позволяет анализировать схемы даже с 1000 транзисторами. В настоящее время завершается версия МОНЕТ Х2, которая не зависит от технологии и имеет более высокое быстродействие.

В подсистему «Проверка проектирования масок» включена также программа ЛОМАХ для логического моделирования функций ИС МОП на основании данных, полученных из анализа масок. В современной версии программа ЛОМАХ предназначена для схем,



имеющих до 5000 транзисторов. Программа базируется на представлении того, что логические состояния определяются диффузионными областями, заданными рисунком масок. Входными данными для программы ЛОМАХ является описание схемы, сгенерированное программой МОНЕТ. Объединением обеих этих программ можно проверить, отвечают ли принципиальные функции схемы, реализуемой данными масками, ожидаемым. Вновь создается подсистема для проверки масок путем сравнения с первоначально спроектированной схемой. Для данной подсистемы готова программа ЭЛКО, которая позволяет сравнивать логические элементы в КМОП-схеме, сгенерированной программой МОНЕТ, с элементами в массиве данных.

Для подсистемы «Моделирование» имеются имитаторы электрической и логической функций схем. Для моделирования частей интегральных схем на уровне электрической схемы предназначена программа SIC. С помощью этой программы можно моделировать поведение схемы в установившемся режиме, при возбуждении малым гармоническим сигналом и при любом временном прохождении возбуждающего сигнала большой амплитуды. Кроме основного анализа в рамках отдельных методов моделирования, она позволяет также проводить анализ чувствительности, допусков, вычисление худшего случая, спектральные и температурные расчеты. На основе автоматически сгенерированных линеаризованных моделей нелинейных схем можно вычислить характеристики передачи и входные или выходные импедансы этих схем для малых сигналов. При временном анализе можно применить гибридную имитацию, при которой часть схемы моделируется на уровне электрической схемы и часть — на уровне логической схемы. Программа имеет для данной цели логический имитатор (который работает во взаимодействии с программой электрического анализа) и модели преобразовательных элементов интерфейса, которые обеспечивают преобразование электрических и логических величин.

При временном анализе можно затребовать распечатку по-разному определяемых задержек и длительностей фронтов импульса, распечатку дискретизированных значений схемных величин и т. п.

За один прогон программы можно проанализировать разные варианты одной или нескольких схем. Вычисления можно условно закончить в зависимости от хода решения. Промежуточные результаты в ходе вычислений можно запоминать для позднейших вычислений или для дополнительной обработки выходных данных без проведения нового анализа.

Комбинирование разных видов анализа и возможность обобщения результатов в обзорной графической форме облегчают пользователю ориентацию в полученных данных. Программа SIC оснащена рядом встроенных моделей полупроводниковых приборов, в частности моделями, пригодными для моделирования ИС МОП.

Существует несколько версий программы SIC с разными требованиями к емкости оперативной памяти. Например, версия для

360 Кбайт позволяет анализировать схемы с 80 транзисторами МОП и 200 логическими элементами максимально с 800 входами.

Для моделирования функций логической схемы существует программа ЛОСИМО, которая позволяет моделировать функцию логической схемы, составленной из логических элементов, заложенных в основном массиве данных. В качестве логического элемента может выступать любая частная схема, которую можно составить из элементарных схем AND, NAND, OR, NOR или схем с тремя состояниями. Пользователь может определить собственные элементарные схемы, для которых, однако, он должен написать таблицу истинности в форме подпрограммы на языке Фортран.

Был создан более современный логический имитатор ЛС 1, предназначенный для логического моделирования асинхронных схем (включая контроль ложных выходных сигналов логической схемы). Входной язык программы имеет свободный формат и по синтаксису и структуре близок мышлению проектировщика схемы. Программа ЛС 1 имеет встроенные модели основных комбинационных элементов AND, OR, NAND, NOR, XOR, NXOR и модуль универсального комбинированного элемента (логическая функция определена пользователем). ЛС 1 содержит далее модели различных типов памяти, считывающего устройства и регистра сдвига. Описание схемы — иерархическое, максимально с 10 уровнями. Описание возбуждения позволяет описывать с помощью входного языка отдельные входы и целые группы одновременно. Оно позволяет применять, кроме классического управления, и условные команды (входная последовательность зависит от результатов моделирования), макрокоманды (входная последовательность определена в общем виде и исползуется затем с определенными параметрами). При объеме оперативной памяти 250 Кбайт программа ЛС 1 может моделировать схему с 2000 вентилями.

Для проектирования масок предназначен программный комплект ЗАНИО. Этот комплект на основе логической схемы автоматически создаст программы для управления генератором изображений для изготовления масок. Входными данными комплекта являются: закодированная логическая схема и данные, специфицирующие размеры чипов и изменения проектных правил, которые встроены в систему. Входные данные представляют команды для управления генератором изображений чертежных автоматов, записанные на диске или ленте. Комплект ЗАНИО содержит следующие блоки программ: для размещения логических элементов и контактов чипа; для соединения логических элементов и контактов чипа; для генерирования команд, управляющих чертежным автоматом для изготовления маски; для генерирования команд, управляющих генератором изображений; для записи и выбора данных из основного массива.

Комплект ЗАНИО дополнен программой ГЕВЕРС, предназначенной для автоматизации проектирования структур ПЗУ. Программа позволяет генерировать графические данные, представляющие информацию, которую надо записать в ПЗУ на основе задания этой информации в форме шестнадцатеричных данных или в форме

программы, записанной в виде последовательности инструкций в системе команд проектируемой системы. С помощью ГЕВЕРС можно на основе графических данных расшифровать записанную информацию.

Для подсистемы «Формирование моделей» имеется универсальная программа РЕГРЕС. Эта программа была разработана как универсальное средство для решения частных или комплексных задач идентификации параметров нелинейных моделей элементов ИС. Задачу идентификации параметров модели, которая должна быть задана в явной форме, она решает с помощью минимизации суммы квадратов. Выбрав один из двух методов минимизации, можно выбирать между надежностью и скоростью поиска минимума. Ни один из методов не требует знания явного выражения градиента критерия минимизации.

Для решения конкретной задачи надо написать на языке Фортран IV подпрограмму МОДЕЛЬ. Полученные данные для ввода в программу необходимо записать в стандартном виде, выходные данные представляются в форме таблиц и графов. Программа проектирована как открытая. Пользователь может ее дополнить собственными подпрограммами МЕРЕНИ и РЕСУЛТ, с помощью которых можно будет считать входные данные и в другой, отличной от стандартной, форме и дополнить вывод программы своими специальными требованиями.

Ходом всего процесса идентификации можно управлять с помощью управляющих параметров. В наиболее простых случаях достаточно задать только первую оценку искомых параметров. Можно, однако, создать и более сложные варианты, когда в циклах проводится ряд минимизаций с разными входными и управляющими данными, причем результаты одного цикла могут быть увязаны со следующим. Все программы создаваемой системы были отлажены на языке Фортран и введены на ЕС ЭВМ.

В 1983—1985 гг. будет завершена реализация оставшихся подсистем. Прежде всего будут созданы средства для моделирования технологического процесса и физические модели и средства для испытания и диагностики. С созданием комплексной системы проектирования будет доработана и программная подсистема «Диалоговое проектирование масок».

**Графические системы для проектирования ИС.** Для проектирования морфологии, а во многих случаях и для проектирования схем и составления документации, применяются интерактивные графические системы, т. е. системы с программным обеспечением для подготовки и обработки графических данных и с достаточным объемом дополнительных специализированных программных средств для проектирования ИС. Для проектирования схем СБИС, однако, свойства этих графических систем недостаточны. Поэтому строится комплексная система проектирования, которая позволит сосредоточить данные проекта и связать все применяемые в нем программные средства. Она позволит разработать морфологию схем в субмикронной технологии, что в имеющихся системах, применяющих

шестнадцатибитовое слово для координат, сделать затруднительно. Система проектирования имеет иерархическую структуру: диалоговая графическая станция — диалоговая система проектирования — сеть ЭВМ. Ядром системы проектирования является управляющая мини-ЭВМ с длиной слова 32 бита. Графическая станция имеет в своем составе управляющую микроЭВМ, цветной растровый дисплей, алфавитно-цифровой дисплей, обеспечивает ввод координат с планшета или цифрового преобразователя, оснащена средствами индикации.

Все данные проекта будут удобно структурированы и записаны в основном массиве. База данных позволяет иметь доступ к интерактивным программам для подготовки и обработки данных и к дополнительным программам для конкретного применения (например, к моделирующим программам). Графическое представление объектов будет только одним из их представлений и будет служить, в частности, для связи проектировщика с системой, так как эта форма для человека является наиболее удобной. Программное обеспечение предназначено для подготовки и обработки всех данных, создаваемых проектировщиком, и использования всего, что нужно для проектирования, например составление иерархических схем в графическом представлении, формирование морфологии (включая высшие уровни), подготовку документации и т. п.

**Формы координации и управления.** Для управления и координации работ по заданиям головного предприятия по проектированию ИС создан в качестве консультативного органа Координационный совет. Его членами являются представители научно-исследовательских институтов, вузов, институтов Академии наук и специалистов заводов-изготовителей. Главной задачей Координационного совета является объединение программного обеспечения и технических средств. Основой объединения программного обеспечения является введение единых структур данных для ввода-вывода. Единые структуры данных являются основной предпосылкой для простой и быстрой передачи программных средств между рабочими местами, несложного и быстрого создания собственных проблемно-ориентированных программных систем из программ любого происхождения, гибкого обновления программного обеспечения (замены устаревших программ новыми).

В 1982 г. рабочие группы Координационного совета подготовили проект логической структуры банка данных, который должен служить в качестве стандарта для передачи информации между системами проектирования и подсистемами (рабочее название MIF — Microelectronic Intermediate Form). Структура данных имеет стабильно определенную часть, связанную с информацией о собственной структуре схемы, определен способ дополнения информации о схеме, например данными о графической форме схемы, о форме масок, моделях и т. д. Проект логической структуры был доработан до уровня логических фраз, т. е. были определены типы фраз, их содержание, формат позиций и синтаксис всего комплекта. Физическую форму фраз необходимо определить отдельно для каждой

систем, между которыми будет происходить обмен. Для описания графических изображений масок и схем подготовлен проект языка СИМ.

УДК 681.325.5.181.5:62-82

## ИСПОЛЬЗОВАНИЕ МИКРОПРОЦЕССОРОВ В СИСТЕМАХ УПРАВЛЕНИЯ ЭЛЕКТРОГИДРАВЛИЧЕСКИМИ ИСПЫТАТЕЛЬНЫМИ МАШИНАМИ

В. Кацегл, инженер (ЧССР)

По своему использованию, сложности и требованиям к обслуживанию электрогидравлические испытательные машины относятся к оборудованию, называемому научными приборами и аппаратурой. Такие устройства собираются из сопрягаемых модулей, которые характеризуются входными и выходными параметрами. В отличие от испытательных машин, управляемых электронной аппаратурой без ЭВМ, испытательные машины, управляемые при помощи ЭВМ, обеспечивают качественно более высокий уровень программирования алгоритмов испытаний и расширяют информацию об испытываемом объекте.

Электрогидравлические испытательные машины в большинстве случаев включены по схеме электрогидравлического сервопривода, работающего в режиме, удовлетворяющем высоким требованиям к постоянству статической характеристики замкнутого контура и ширине частотного диапазона динамической характеристики. На рис. 1 приведена принципиальная схема такой машины, предназначенной для испытаний материалов. По схеме можно проследить принцип действия машины.

Сигнал 1 подается на вход электрогидравлического преобразователя, который управляет впуском напорной жидкости в прямолинейный гидродвигатель. К поршневому штоку гидродвигателя подсоединен испытываемый объект 2, воздействующий на датчик силы 3. Силовое воздействие поршневого штока, управляемое сигналом 1, преобразуется датчиком силы 3, экстензометром 4 или датчиком смещения 5 в электрический сигнал обратной связи. В регуляторе этот сигнал сравнивается с сигналом требуемого значения 6, и их разность после преобразования подается на сервоклапан, который регулирует нагрузочную силу таким образом, чтобы эта разность минимизировалась.

Электрогидравлические испытательные машины используются для возбуждения программируемого и изменяющегося во времени воздействия величины (силы, деформации, смещения) на исследуемый объект. Кроме того, эти машины приспособлены для измерения и обработки данных, характеризующих реакцию исследуемого объекта. Иными словами, речь идет о метрологических задачах,

сводящихся к определению или измерению усталостных характеристик или характеристик долговечности образцов материалов и деталей, а также динамических характеристик долговечности различных конструкций и их узлов.

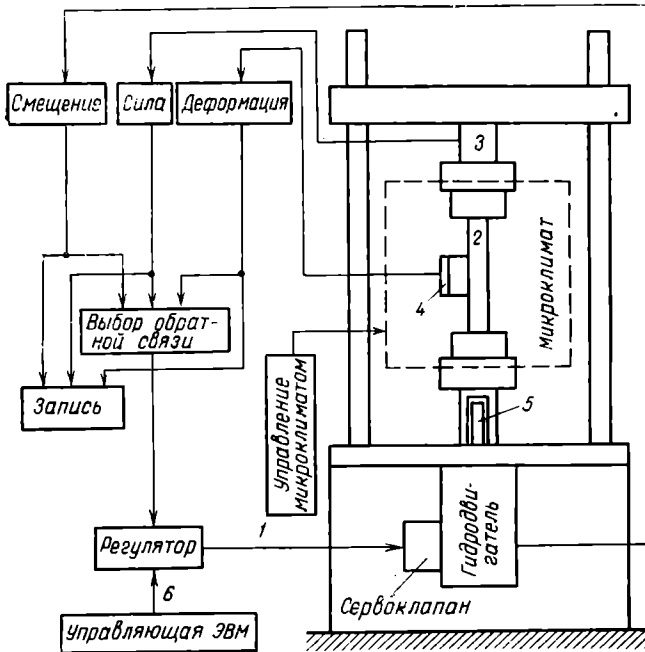


Рис. 1. Принципиальная схема электрогидравлической испытательной машины, предназначенной для испытания материалов:

1 — сигнал; 2 — испытуемый объект; 3 — датчик силы; 4 — экстензометр; 5 — датчик смещения; 6 — сигнал требуемого значения

Для оценки технического уровня электрогидравлической испытательной машины решающим фактором является уровень автоматизации эксперимента. Человек должен только подготовить эксперимент, а затем вся программа эксперимента осуществляется автоматически. Человек должен вмешиваться в ход эксперимента только в заранее предусмотренных случаях или в случае непредвиденных обстоятельств.

Электрогидравлическая испытательная машина является метрологическим устройством. Она выполняет следующие функции:

создает нагрузочное воздействие требуемой величины и формы на испытуемый объект в требуемом месте, в нужном направлении и с необходимой точностью;

измеряет с требуемой точностью заданную реакцию исследуемого объекта под нагрузкой и использует одну из реакций в качестве сигнала обратной связи;

отличается длительной бесперебойной работой при соблюдении требуемой точности нагрузки и измерения, а также наличием защиты, предотвращающей повреждение машины и разрушение испытуемого объекта в случае неисправности сервомеханического привода.

Потребительская стоимость электрогидравлической испытательной машины возрастает по мере расширения возможностей программирования нагрузочной величины (возможность выбора любой формы, повышения точности, расширения частотного диапазона и т. д.) и по мере повышения объема, качества, точности и скорости получения информации.

Использование средств вычислительной техники в управляющем устройстве испытательной машины радикально влияет на ее технические параметры и многообразие выполняемых функций. Существенно повышается возможность программирования процесса нагрузки, возрастает скорость анализа реакции нагружаемого объекта и свобода выбора метода анализа.

Несмотря на большую скорость осуществления указанных операций вычислительной машиной, ее процессор представляет собой критическое звено системы, поскольку обработка информации в нем производится последовательно.

Использование вычислительной техники в управлении машинами можно рассматривать как случай цифрового управления быстропротекающими технологическими процессами. Эффективность использования ЭВМ зависит не только от параметров собственно процессора, но и от структуры системы управления и свойств интерфейсных устройств, обеспечивающих связь ЭВМ с экспериментом. Многое зависит от выбора уровня, на котором происходит разделение функций программных и технических средств. Блок-схема устройства, пригодного для управления быстрыми процессами, показана на рис. 2. Система содержит управляющую мини-ЭВМ со стандартными периферийными устройствами, к которой через самостоятельные входные и выходные каналы подключены интерфейсные устройства для связи с управляемым процессом. Для быстрой синхронизации работы ЭВМ с процессом система имеет расширенный канал сигналов прерывания.

Преимуществом такой структуры является возможность использования единственной ЭВМ, которая составляет существенную часть стоимости управляющего устройства. Недостатком является высокая степень централизации функций управления. Это ведет к необходимости использовать для подключения специальные периферийные устройства, параллельные каналы передачи данных с высокой пропускной способностью и вспомогательные технические средства, ускоряющие передачу через канал прямого доступа в память. При проведении эксперимента машинное время должно динамически распределяться между большим количеством разных задач, связанных во времени с отдельными событиями в управляемом процессе. При этом известно, что переклечение процесса на решение разных задач приводит к потерям времени, в особенности если оно организуется средствами программного обеспечения (опе-

рационной системой). Неблагоприятное последствие частого переключения управления при указанной структуре в значительной мере компенсируется мощным каналом сигналов прерывания. Эта структура обеспечивает синхронизацию вызова подпрограмм прерывания работы процессора. События, вызывающие это прерывание, выявляются из протекающего процесса при помощи технических средств.

Структура, представленная на рис. 2, используется в управляющем устройстве ЭДИЗ 5, разработанном в 1975—1980 гг. на предприятиях ИНОВА. Устройство предназначено для управления слож-

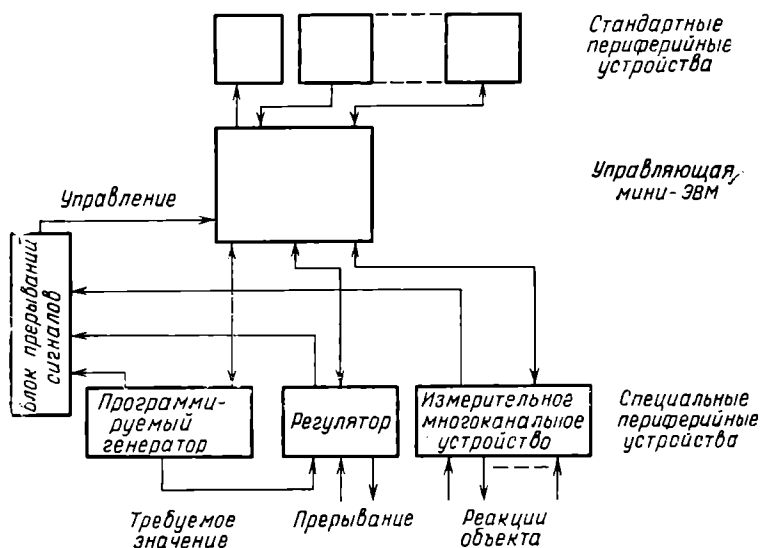


Рис. 2. Блок-схема устройства для управления быстрыми процессами

ными испытаниями материалов и узлов конструкций и оснащено шестнадцатизарядной мини-ЭВМ типа ADT 4500. Специальные периферийные устройства построены таким образом, что они автономно реализуют ряд часто встречающихся функций, например генерирование промежуточных значений входного сигнала либо сжатие большого числа измеренных данных и их представление характеристическими значениями (расчет экстремумов). Такое решение существенно экономит машинное время и создает возможность использования сравнительно медленного, но более удобного для пользователя языка программирования Бейсик.

Опыт, накопленный при разных применениях управляющего устройства ЭДИЗ 5, однако, показывает, что введение высокопроизводительных специальных периферийных устройств и каналов передачи данных, полностью удовлетворяющих требованиям эксперимента, влечет за собой чрезмерную сложность работы и произ-



водства аппаратуры, а также ее пониженную надежность. При этом исправная работа всех приборов управляющего устройства обусловлена правильной работой ЭВМ.

Новые возможности создает применение микропроцессорной техники, которая позволяет использовать при построении системы весьма выгодные структуры с распределенным «интеллектом», существенную часть которого можно реализовать на наинизшем уровне. Это позволяет разделить единственное слабое звено вычислительной системы — процессор — на несколько разнесенных частей в соответствии с отдельными функциями. При распределении задач между несколькими процессорами можно использовать следующие структурные варианты системы:

мультипроцессорная система, имеющая несколько процессоров, которые обрабатывают самостоятельные задачи, а для взаимной передачи данных используют общую часть оперативной памяти. Такое решение позволяет осуществить весьма эффективную работу отдельных процессоров без потерь времени, вызванных частой передачей управления между отдельными процессорами. Одновременно возможна весьма быстрая и эффективная передача данных между отдельными процессорами и значительное уменьшение организационной роли операционной системы;

мультипроцессорная система с распределенной обработкой, которая, кроме того, управляется операционной системой более высокого уровня, следящей за работой отдельных процессоров и динамически использующей их свободное время для решения задач с большой скоростью вычисления. Недостатком обеих указанных мультипроцессорных систем является необходимость иметь тесную физическую связь между процессорами;

децентрализованная система, позволяющая производить обработку отдельных задач в самостоятельных процессорах без непосредственной связи между программами. Процессоры могут быть включены в систему с иерархической структурой. Преимуществом являются относительно простые и легко определяемые взаимосвязи между отдельными процессорами. Недостаток такой структуры состоит в сложности передачи данных между процессорами, проводимой в форме согласованных протоколов посредством внешних каналов. При использовании подходящего типа каналов передачи данных отдельные части системы могут быть взаимно разнесены.

При выборе подходящей структуры необходимо учесть возможность разделения отдельных функций управляющего устройства в зависимости от объема и характера данных, которыми должны обмениваться отдельные процессоры. Одновременно необходимо учитывать гибкость структуры и необходимость создания высшего уровня управления, централизующего основные функции управления.

Как уже было сказано, управляющее устройство электрогидравлических испытательных машин обеспечивает выполнение трех основных функций, у которых программируемость является основным требованием: генерирование заданного воздействия; регулирование его приложения; измерение и обработку данных, характеризующих

реакцию нагружаемого объекта. Такое разделение важнейших функций удобно использовать при определении физической структуры управляющего устройства. Взаимосвязи этих функций в большинстве случаев относительно слабы и просто определяются. Исходной структурой, отвечающей этим требованиям, является децентрализованная система с самостоятельными процессорами, предназначенными для выполнения отдельных основных функций.

Эксперименты с простой программой нагружения и простой обработкой данных могут реализовываться системой с указанной

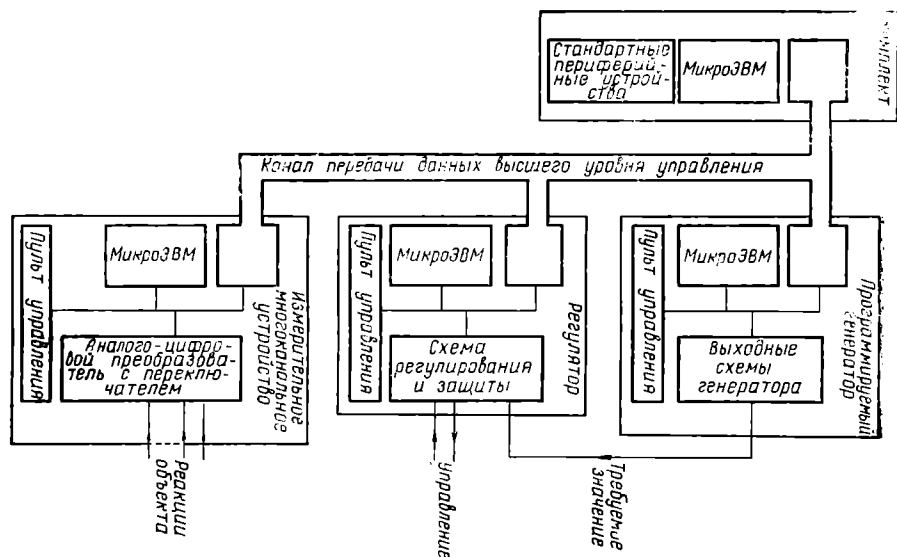


Рис. 3. Блок-схема управляющего устройства электрогидравлической испытательной машины с децентрализованной структурой

базовой структурой. Для управления экспериментами с высокими требованиями к координации отдельных функций необходимо основную структуру расширить добавлением центрального управления. Блок-схема управляющего устройства электрогидравлической испытательной машины с децентрализованной структурой показана на рис. 3. Управляющее устройство состоит из трех приборов с собственными микроЭВМ: программируемого генератора требуемого значения воздействия, электронного регулятора воздействия с программируемым законом управления и измерительного многоканального устройства. Система может быть дополнена микроЭВМ со стандартными периферийными устройствами для осуществления центрального управления.

Каждый прибор содержит собственную микроЭВМ, к шине которой подключены простой пульт ручного управления и дополнительные схемы, специфические для функции каждого прибора. К шине далее присоединены интерфейсные схемы для связи с высшим уровнем. Приборы отличаются указанными дополнительными

схемами, а также управляющей программой, заложенной в постоянной части оперативной памяти микроЭВМ. Остальные части приборов могут быть идентичными.

Построенные таким образом приборы могут выполнять свои функции совершенно независимо. Перед началом выполнения требуемой функции микроЭВМ запрашивает необходимые параметры в диалоге, проводимом с оператором при помощи пульта управления. Параметры запоминаются в резервируемой части оперативной памяти.

Преимущество микроЭВМ, расположенной непосредственно в приборе, наиболее отчетливо проявляется при управлении соответствующей функцией в реальном времени. Узкий спектр решаемых задач и возможность создать специфические схемы прибора соответствующими свойствами процессора приводят к эффективному использованию машинного времени микроЭВМ. Таким образом можно, например, построить генератор требуемого значения воздействия, технические параметры которого сравнимы с параметрами соответствующего генератора в системе с центральной ЭВМ, однако его сложность и потребляемая мощность примерно в два раза ниже. Для его работы не требуется ни дополнительного устройства, ни внешнего канала передачи данных. Надежность простого компактного прибора поэтому в несколько раз возрастает. Обслуживание по сложности не отличается от обслуживания простых лабораторных приборов и не требует знаний программиста.

Опыт создания такого генератора показывает, что микропроцессор типа 8080 может одновременно обеспечить генерирование четырех независимых функций времени с программируемыми амплитудой, постоянной составляющей и длительностью каждого полупериода при частотах до 50 Гц. В этом режиме у процессора остается приблизительно 30% времени для решения дополнительных задач, которые могут быть вызваны событиями, возникающими, например, при вмешательстве оператора.

К подобным результатам можно прийти при сравнении технических параметров и надежности регуляторов и измерительных устройств. Возможность программной сортировки и обработки измеренных данных в ходе эксперимента непосредственно в измерительном устройстве существенно снижает требования к оснащению лаборатории дополнительным оборудованием. С весьма простой управляющей аппаратурой, состоящей лишь из трех приборов, можно проводить большинство часто встречающихся испытаний образцов материалов и конструкций.

Для сложных испытаний, требующих программирования отдельных функций в большом многообразии и их взаимного координирования, необходимо подключить все приборы через канал связи к центральной ЭВМ. Требования к производительности этой ЭВМ и канала связи существенно ниже, чем у системы с центральным процессором, за счет значительной автономности работы отдельных приборов на низшем уровне.

Присоединение центральной ЭВМ расширяет возможности управляющего устройства в следующем отношении:

создается централизованное средство управления работой всех приборов в реальном времени;

возможно осуществление режимов управления, требующих большого многообразия взаимосвязи между приборами (например, переключение управляемого воздействия без возникновения толчков);

возможно осуществление функций, требующих наличия стандартных периферийных устройств (печать протокола, графический выход);

возможно осуществление таких функций в приборе, для задания которых не хватило бы емкости их памяти.

Центральную ЭВМ можно использовать самостоятельно (не во время эксперимента) для подготовки испытаний на языке высокого уровня и их записи во внешнюю память и для программной обработки измеренных данных с использованием сложного программного обеспечения, доступного лишь при наличии внешней памяти.

Взаимосвязь между уровнями системы используется для введения задаваемых параметров в память приборов до начала эксперимента, для синхронного пуска и останова работы приборов, для передачи блоков сжатых данных в центральную ЭВМ для принятия решений в момент достижения заданного критерия. Быстрые последовательности команд могут реализоваться простым запуском приборов на низшем уровне в соответствии с одним из резервных заданий, некоторое количество которых может быть предварительно заложено в память микроЭВМ.

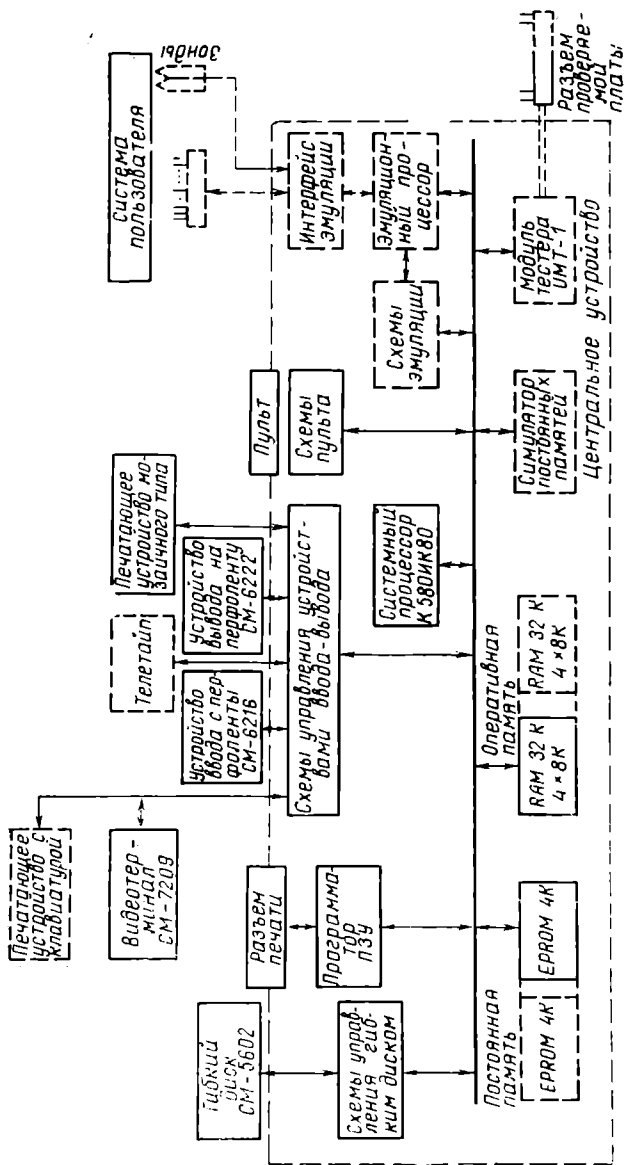
При использовании большего числа микропроцессоров, специализированных на решении узких задач при возможности взаимного обмена данными на нескольких уровнях, появляется возможность составления из одинаковых сравнительно простых независимых функциональных блоков управляющей аппаратуры, производительность которой может изменяться в очень широких пределах. Таким способом можно эффективно удовлетворять самые различные требования эксперимента.

УДК 681.325.5-181.5

## **МИКРОПРОЦЕССОРНАЯ СИСТЕМА ВСПОМОГАТЕЛЬНЫХ СРЕДСТВ ПРОЕКТИРОВАНИЯ**

**Т. Синкевич, д-р техн. наук (ПНР)**

Микропроцессорная система вспомогательных средств проектирования MSWP, созданная в Институте математических машин в Варшаве, предназначена для разработки, наладки и испытаний аппаратной и программной части микропроцес-



Структура микропроцессорной системы вспомогательных средств проектирования

сорных устройств. Конфигуратор системы отличается широким набором вспомогательных технических и программных средств проектирования и наладки микропроцессорных устройств, реализованных на схемах серии К 580 и К 589. Модульная структура системы позволяет применять ее и для других микропроцессорных серий.

Основные функциональные блоки системы (см. рисунок) размещены в центральном устройстве, к которому подсоединены системные устройства ввода-вывода, накопитель на гибких магнитных дисках и отлаживаемая микропроцессорная система пользователя. На пульте центрального устройства находятся основные элементы управления и сигнализации, кнопки управления прерываниями и разъем программатора устройств постоянной памяти и логических матриц.

Пользователь системы в зависимости от нужд и имеющихся средств может менять состав системы MSWP. Это возможно благодаря модульной структуре центрального устройства, системе магистрали типа MULTIBUS и схемам ввода-вывода, обеспечивающим возможность применения различных внешних устройств.

В центральном устройстве MSWP с минимальным составом оборудования могут использоваться следующие функциональные модули: системный процессор, схемы управления устройствами ввода-вывода, память с произвольным доступом емкостью 16 Кбайт, постоянная память емкостью 2 Кбайт, схема управления пульта и программатор программируемых ПЗУ типа INTEL 1702A. В качестве универсального устройства ввода-вывода можно применять теле-тайп. Таким образом получается простая система подготовки и отладки программы для микропроцессорной серии типа INTEL 8080 и ее аналогов.

Система MSWP в полном составе поставляется с ОЗУ емкостью 60 Кбайт, ПЗУ емкостью 4 Кбайт, накопителем на гибких магнитных дисках емкостью около 1 Мбайта, симулятором блоков постоянной памяти (512 слов длиной до 64 бит), сменными блоками программаторов ППЗУ типа INTEL 1702A, 2704, 2708, 2716, 2732 и аналогов, например типа К 573РФ1 советского производства, схемным эмулятором микропроцессорной серии К 580 (аналог серии INTEL MCS 8080).

В качестве стандартного операторского пульта в системе MSWP применяется алфавитно-цифровой видеотерминал СМ-7209 с клавиатурой типа СМ-7103 или любой другой алфавитно-цифровой видеотерминал или печатающее устройство с клавиатурой, работающие с последовательным интерфейсом V24. Для ввода-вывода перфолент применяются перфоленточное считывающее устройство СМ-6216 и перфоратор типа СМ-6222. Дополнительным внешним устройством является печатающее устройство типа DZM 180 (с параллельным интерфейсом), пригодное для вывода листингов.

Учитывая запросы пользователей разработаны следующие дополнительные модули системы MSWP:

- универсальный модуль тестера UMT-1;
- программатор ППЗУ типа ТМ 621/624 фирмы Tungstap;

программатор логических матриц типа 82S 100;  
алфавитно-цифровой видеотерминал с клавиатурой, построенный на основе типового телевизионного приемника;  
лампа, стирающая запись в ППЗУ с помощью ультрафиолетового излучения.

Универсальный модуль тестера UMT-1 проверяет правильность изготовления печатных плат и тестирует платы с интегральными схемами, имеющими входные и выходные сигналы уровня. Тестер может обслуживать до 320 выводов (входов и выходов), а стандарты на проверку печатных плат и комплектных ТЭЗ могут определяться заказчиком.

Видеотерминал на базе телевизионного приемника с успехом заменяет алфавитно-цифровой видеотерминал с клавиатурой, но стоит дешевле, хотя выполняет все требуемые функции операторской консоли, а также графические функции.

Имеющиеся системные программы дают пользователю возможность подготавливать программы для микропроцессорных устройств:

- редактировать исходные программы;
- транслировать исходные программы на машинный язык;
- отлаживать и проверять программы на машинном языке;
- исправлять и симулировать микропрограммы;

создавать проблемно-ориентированные языки на базе языка Ассемблера с использованием аппарата макроинструкций;  
эмулировать команды микропроцессоров, отличающихся от К 580ИК80 макроинструкциями;  
создавать и модифицировать собственное системное программное обеспечение.

Программное обеспечение системы MSWP в зависимости от используемого носителя можно разделить на три группы:

- программы, хранимые в ПЗУ системы и входящие в состав фирменного программного обеспечения;
- программы на перфолентах;
- программы на гибких магнитных дисках.

К программам первой группы принадлежат следующие программы:

MONITOR — основная операционная система, управляющая начальной установкой системы, содержащая программы, управляющие устройствами ввода-вывода, и одиннадцать директив для наладки программ пользователя в двоичном коде;

PROGRAMATOR 1702 — программа, управляющая программатором ПЗУ типа EPROM INTEL 1702;

DYSK — программа, управляющая работой внешней памяти на гибких магнитных дисках.

Ко второй группе относятся программы:

EDYTOR 1 (PEEDYT) — редактор текстов с буфером текста, ограниченного размером оперативной памяти (RAM);

EDYTOR 2—редактор текстов, записанных на перфоленте, использующий буфер текста, равный максимальному количеству знаков исправляемой строки текста;

МАКРОASSEMBLER REM — транслятор программ, написанных на языке макроассемблера типа INTEL 8080;

МАКРОASSEMBLER-EDITOR REMAK — эта программа является соединением программ EDYTOR 1 и программы REM с общим буфером, содержащим исходную программу;

MINI BASIC — интерпретирующая программа языка BASIC, действующая в диалоговом режиме с шаговым выполнением последовательно вводимых инструкций, оперирующая числами с фиксированной запятой;

BASIC — интерпретирующая программа языка BASIC, действующая в диалоговом режиме. Она использует буфер емкостью 6 Кбайт для хранения команд и данных программы; оперирует числами с плавающей запятой;

EMULATOR 8080 — программное обеспечение модуля эмулятора для систем с микропроцессором типа INTEL 8080;

SYMROM — программное обеспечение симулятора памяти ROM емкостью  $512 \times 64$  бита для систем с микропроцессором типа К 589ИК01 (аналог INTEL 3000);

TEST RAM — набор тест-программ для проверки оперативной памяти RAM;

TEST CT/DT — набор тест-программ для проверки перфоленточных устройств;

MONITOR TESTERA — управляющая программа тестера UMT-1 для проверки и наладки печатных плат;

JEZYK TESTERA — транслятор языка описания тестирующих последовательностей тестера.

Третья группа содержит две дисковые операционные системы и программы на гибких магнитных дисках, действующие под управлением этих систем.

Дисковая операционная система OS-I (аналог системы CP/M) состоит из редактора текстов, транслятора языка Ассемблера типа INTEL 8080, программы передачи данных, отладочной программы, программы изменения формата файла, программы ведения статистики о состоянии файлов, программы выдачи дисковых файлов, программы интегрирования системных макродиректив.

К дополнительным программам системы OS-I относятся: транслятор языка макроассемблера INTEL 8080, символическая отладочная программа и программа форматирования текстов.

BASIC 2.1 — быстрая интерпретирующая программа языка BASIC. BASIC 2.2 — полукомпилятор языка BASIC для научных и технических расчетов, позволяющий обрабатывать значительно больше программ, чем с помощью интерпретирующей программы. BASIC 2.3 — полукомпилятор языка BASIC, предназначенный для административных применений и содержащий десятичную арифметику и богатый состав функций для обработки больших программ.



TEST FD — тест-программа памяти на гибких магнитных дисках (может использоваться тоже для проверки носителя).

ODZYSK — программа реконструкции испорченных файлов с точностью до 1 Кбайт.

RETRI — программа реконструкции испорченных файлов с точностью до одной зоны. UZDA — программа, приспособляющая гибкие диски с физическими повреждениями для работы в системе.

SPRAWDZ — программа проверки перфоленточной копии дисковых файлов.

В состав дисковой операционной системы OS-II входят: программа форматирования гибких дисков и копирования файлов, интерпретирующая программа системных макродиректив, выдачи содержания каталога, копирования файлов, ликвидации файлов, программа, изменяющая атрибуты и форматы файлов, программа управления библиотеками, программа, преобразующая перемещаемые объектные файлы в абсолютные, программа объединения нескольких результирующих модулей в один.

К дополнительным программам OS-II относятся: редактор текстов, транслятор перемещаемого макроассемблера и компилятор языка PL/M аналог PL (M-80).

В конечной стадии разработки находятся: интерпретирующая программа языка высшего уровня FORTH, работающая под управлением дисковой операционной системы OS-I, и компилятор языка Фортран, работающий под управлением дисковой операционной системы OS-II.

Проектировщики микропроцессорных систем успешно используют систему MSWP как при независимой разработке оборудования и программного обеспечения, так и при наладке оборудования совместно с программным обеспечением.

При разработке пользовательских программ проектировщик может применять вспомогательные средства системы MSWP, начиная с этапа кодирования исходных программ.

Нужная системная программа вводится в ОЗУ с перфоленты или с гибкого магнитного диска. Ввод, корректировка и вывод программ в исходном виде выполняются с помощью наиболее выгодной из редактирующих программ. Двоичные коды пользовательских программ получаются в результате применения транслирующих программ, причем пользователю системы MSWP предлагается несколько трансляторов.

Наладка результирующих программ осуществляется с помощью директив программы MONITOR, если система оснащена запоминающим устройством на гибких магнитных дисках — с помощью более эффективной отладочной программы или ее расширенной версии — символической отладочной программы. Обе программы работают под управлением дисковой операционной системы OS-I.

Наладка оборудования, содержащего микропроцессоры с фиксированным перечнем команд, совместно с программным обеспечением наиболее эффективно производится при использовании модуля схемного эмулятора, управляемого программой EMULATOR, дирек-

тивы которой обеспечивают отладку и тестирование пользовательских программ с помощью эмуляционного процессора (аналогичного процессору пользователя) и эмуляционных схем сначала без подключенных схем пользователя. В зависимости от типа применяемого микропроцессора используется соответствующая версия программы EMULATOR совместно с модулем эмулятора. Пользовательские программы после окончания их наладки можно записать в ПЗУ системы пользователя с помощью модулей программатора ПЗУ, а затем исследовать совместно с оборудованием пользователя.

При наладке систем с микропроцессорами сегментного типа, например серии К 589, с помощью симулятора проверяется корректность разработанных микропрограмм. Управление симулятором ведется с помощью программы обслуживания SYMROM, разрешающей чтение и запись микропрограмм согласно стандарту языка CROMIS.

В результате подключения к системе MSWP модуля тестера УМТ-1 создается программируемый тестер печатных плат и комплектных цифровых ТЭЗ, пригодный для мелкосерийного производства цифрового оборудования.

Состав системы MSWP может меняться в зависимости от ее применения. Система может использоваться, например, для обработки информации в контрольно-измерительных системах, сбора и переработки данных в торговых системах, библиотеках и др., для создания диалоговых систем и дидактических лабораторий для проектирования аппаратных и программных средств микропроцессорных устройств, для программирования на языках высшего уровня и т. п.

УДК 658.5:681.3

### **СИСТЕМА УПРАВЛЕНИЯ ПРОИЗВОДСТВОМ НА БАЗЕ МИКРОЭВМ ДЛЯ МЕЛКИХ И СРЕДНИХ ПРОМЫШЛЕННЫХ КООПЕРАТИВОВ**

**Т. Раммахер, инженер (ВНР)  
Г. Шомодвари, инженер (ВНР)**

Особенностью управления мелкими и средними промышленными кооперативами Венгрии является относительно небольшой период управления, потребность в частичных решениях, высокая чувствительность к затратам.

В связи с малым периодом управления для принятия решений требуется интенсивная информационная подготовка, при этом постоянно меняется как структура выпускаемой продукции, так и данные, характеризующие состав клиентов. Характерные особенности управления промышленными кооперативами определяют и

те требования, которые предъявляются системой управления производством к информационной системе: она должна обеспечивать достоверной информацией частичные решения, своевременно предоставлять информацию и функционировать при малых затратах.

Система управления производством должна разрабатываться с учетом следующих факторов:

- нужны единые принципы разработки;
- желательно разрабатывать типовые системы;
- внедрение системы не должно представлять особых трудностей;
- структура разрабатываемой системы должна быть модульной, чтобы можно было постепенно внедрять создаваемые программы;
- стоимость ЭВМ, выбранной для обработки данных, не должна превосходить критического предела финансовых возможностей кооператива.

С учетом этих факторов была выбрана микроЭВМ ВТ-20 (производства ВНР), встроенная в видеотерминал СМ-7401. В ее конфигурацию входят: процессор с оперативной памятью емкостью в 64 Кбайт, АЦПУ типа СМ-6302, два накопителя на магнитных дисках СМ-5400 емкостью 5 Мбайт, два накопителя на гибких дисках СМ-5074-1 емкостью 256 Кбайт, четыре терминальных дисплея СМ-7219.

Рабочие программы для делопроизводства написаны на языке Паскаль. Система способна управлять прямыми, последовательными и сцепленными массивами. Логическая связь между записями осуществляется с помощью физических указателей. Отдельные рабочие процессы в системе могут выполняться параллельно. Система состоит из следующих модулей обработки данных: управления поступлением материалов и товарооборотом, управления запасами, решения финансово-отчетных задач, обработки технологической документации, расчета потребности в материалах, управления производственными процессами.

На этапе разработки системы тщательно определялся максимальный размер массивов, которым система может надежно управлять. При этом исходили из емкости имеющихся у пользователей накопителей на магнитных дисках.

**Массивы.** Система включает в себя 13 массивов.

Функции вспомогательного массива заключаются в том, чтобы облегчить доступ к массиву основных данных и к находящимся с ним в непосредственной логической связи массиву последовательных номеров и к массиву номеров заказов (в нем содержатся номенклатурный номер и указатели).

В основном массиве содержатся данные о материалах, полуфабрикатах, готовой продукции, а также различные указатели для облегчения доступа к записям.

Массив мест накопления стоимостей содержит данные о местах, где имеются запасы любого вида, в том числе расходуемых материалов, которые были переданы в данный пункт с целью полного использования.

В массиве запасов содержатся количественные данные и информация о местах хранения по всем видам номенклатур. Внутри данного массива номенклатуры одного типа связаны указателями и хранятся в одном месте.

В массиве статистических данных каждой записи массива основных данных соответствует запись в массиве статистических данных. В его записях содержатся необходимые для статистической системы ВНР первичные данные: запасы на 1 января, общее поступление материалов, общий расход материалов, сумма реализации, остаток запасов на 31 декабря.

В массиве структурного построения продуктов хранятся нормативные данные тех первичных материалов, которые входят в запчасти, узлы и готовую продукцию. Данный массив всегда нужно задавать в форме одноступенчатой структуры.

Массив мощностей содержит основные данные о календарном и рабочем фондах времени (оба в разрезах групп станков одинакового назначения и необходимых для их обслуживания специальностей).

Массив оборудования содержит данные об отдельных станках и оборудовании.

Массив рабочей силы включает данные о каждом рабочем, специальность которого описана в массиве мощностей.

Массив серийных номеров содержит данные о запущенных в производство сериях.

В массиве учета серий накапливаются данные о количестве израсходованной рабочей силы и материалов, связанных с изготовлением данной серии.

Массив заказчиков и подрядчиков хранит основные данные о заказчиках и подрядчиках.

В массиве заказов хранятся данные о пассивных и активных заказах и данные об их выполнении.

Система построена для работы в диалоговом режиме, поэтому контроль первичных данных выполняется на этапе их ввода в машину, что позволяет избежать ввода в систему запрещенных, ошибочных и логически неверных данных. Вызов рабочих программ осуществляется при помощи так называемого «меню». Вначале на экран выводятся наименования модулей «меню», а после выбора необходимого для работы модуля на экране высвечиваются наименования программ данного модуля. При вводе данных на экране вначале появляется их формат (реквизит). В случае обнаружения ошибки программа приостанавливает ввод данных и выдает сигнал о необходимости исправления ошибки. Программа ввода не считает верной ту запись транзакции, для которой нет соответствующей записи в основных массивах. Программа не удаляет запись, в поле указателей которой имеется также ссылка на записи в других массивах. Программа выводит на печать все транзакции.

Задача модуля движения материалов и товарного оборота заключается в предоставлении пользователю информации о состоянии запасов и о составе активных и пассивных заказов.

Для того чтобы данный модуль мог функционировать, необходимо, чтобы в рабочем состоянии находились вспомогательный и основной массивы, массив мест возникновения стоимостей, массив заказчика и подрядчиков и массив серийных номеров. Записи основного массива могут запрашиваться в печатной форме. Отдельные записи или группы записей можно запрашивать на экран дисплея. По завершении построения основного массива вводятся данные массива запасов, причем их нужно вводить по складам.

В системе существует возможность получения двух типов информации о складских запасах:

величины отдельных видов складских запасов (на печать выводятся данные как в натуральном, так и в денежном выражении); сводной суммы запасов (в этом случае производится вывод на печать суммарной величины запасов каждого пункта номенклатуры по всем складам вместе. Распечатка осуществляется в натуральном и денежном выражении).

Оба вида распечаток можно выполнить как на АЦПУ, так и на экране дисплея.

После ввода данных о запасах нужно создать массив заказов. Программы проверки признают верными только те записи, для которых имеются соответствующие записи в массивах заказчиков и подрядчиков и в основном массиве. Заказы могут запрашиваться по номенклатурному номеру заказанной продукции либо по номеру клиентов (отдельно по заказчикам и отдельно по подрядчикам). При этом запросы могут выводиться на АЦПУ или на экран дисплея. После того как сформированы массивы запасов и клиентов, вводятся данные о движении материалов. В результате обработки данных можно оперативно получить информацию о существующих складских запасах, о состоянии выполнения заказов и о расходе запасов по отдельным номерам заказов.

После обработки данных о движении материалов программа автоматически составляет сводку о движении материалов по складам: запасы на 1 января; распечатка данных о движении материалов по номерам квитанций; остаток запасов на 31 декабря.

Сводка о недвижимых материалах выполняется в двух вариантах: в первом печатаются сведения о материалах, которые не расходовались в течение последних трех или шести лет, во втором варианте — материалы, расход которых не был отмечен ни на одном из складов кооператива. В системе предусмотрена возможность анализа приоритета материалов, в результате которого в записи массива запасов вводятся коды, определяющие приоритет материала.

Статистические сводки составляются программами, которые определяют обработку данных с учетом статистической отчетности об изменении запасов.

При функционировании модуля расчета потребностей можно получить информацию в виде сводки о свободных запасах, в которой содержатся сведения о величине запасов в натуральном и денежном выражении и о материалах, не требующихся по плану вы-

пуска готовой продукции, и в виде сводки об имеющихся запасах (т. е. о суммарной величине свободных и активных запасов).

**Расчет расхода материалов.** В результате этого расчета получают итоговые данные о ежемесячном и суммарном использовании материалов и полуфабрикатов по всем пунктам номенклатуры изделий.

Дальнейшее развитие модуля зависит от внедрения модуля расчета потребностей и управления производством, который рассчитывает потребности в материалах и узлах и потребности в мощностях. Имея данные о фактическом использовании материалов, можно определить величину отклонения от норм.

В рамках модуля управления производством подготовка документации выполняется автоматически, т. е. в ходе расчета потребностей получают рабочие наряды и наряды на расход материалов. Обработка нарядов является отдельной задачей системы управления.

**Модуль решения финансово-отчетных задач.** Расчеты, выполняемые в этом модуле, поддерживают финансово-отчетную деятельность пользователя. Как упоминалось ранее, содержание сводки о движении материалов на складах отвечает требованиям аналитического учета материалов. Определяются и выводятся следующие данные:

данные о счетах Главной книги; внутри второго класса счетов, согласно предписаниям пользователя, дается величина запасов на начало отчетного периода, величина всех поступлений и расходов и величина запасов на конец отчетного периода;

итоговые данные о росте и уменьшении запасов в денежном выражении;

расход, т. е. разница между ценой поставки и действительной ценой отдельных номенклатур; расчеты могут быть сгруппированы по номерам заказчиков, а внутри них — по номерам номенклатуры, что дает возможность выявить самого выгодного поставщика.

**Модуль обработки технологической информации.** В этом модуле определяются нормы расхода материалов и нормы трудоемкости. Нормы расхода материалов хранятся в массиве структурного построения продуктов, а норма трудоемкости — в массиве технологических операций.

Программы данного модуля подготавливают:

одноступенчатый перечень компонентов сборки, из которых изготавливается один узел или один конечный продукт;

одноступенчатый перечень встроений, который определяет все компоненты и конечные изделия, содержащие данный вид материала;

построчную распечатку всех компонентов продукции, входящих в данную продукцию, причем они перечислены в порядке их сборки (дается и норма потребности в них);

итоговую потребность в различных материалах и компонентах, содержащихся в некоторых конечных продуктах;

итоговую распечатку отдельных компонентов, встроенных в различные конечные изделия;

распечатку каталога технологических операций, в котором содержится перечень операций, нормативные данные потребностей в рабочей силе для отдельных операций;

итоговую трудоемкость для отдельных видов конечной продукции;

калькуляцию стоимости с учетом материальных и трудовых затрат для каждого вида конечной продукции.

**Модуль расчета потребности в материалах** (содержит две программы). Он определяет потребность в материалах на плановый период, рассчитывает потребность по каждому виду номенклатуры материалов в порядке возрастания номенклатурных номеров. Определяется также потребность в материалах по отдельным номерам (на экран выводится **потребность в материалах по запрошенным номерам заказов**) и трудоемкость для отдельных номеров изделий (на экран выводится трудоемкость для запрошенных номеров). Модуль изготавливает отчет о загрузке производственных участков, в котором содержится фонд полезного времени за квартал на отдельных участках и **сумма загружающих эти мощности нормативных материалов по различным номерам заказов**. Рассчитывается также чистая потребность в материалах. Программа I просматривает и резервирует необходимые для выполнения квартального плана выпуска объемы материалов и выдает информацию как о свободных запасах, так и о недостающих материалах, которые следует заказать у подрядчиков. Программа II выполняет те же функции, что и предыдущая, с той разницей, что она принимает во внимание не только имеющиеся на складе материалы, но и те, на которые приняты заказы.

**Модуль управления производственными процессами.** В результате работы этого модуля создаются учетные документы, определяется порядок производственных процессов и ведется контроль за выполнением сроков отдельных производственных этапов.

Программа выдает сигнал об отклонениях, но сама не влияет на изменения производственного процесса. Программа выдает следующую информацию:

распечатку материалов, в которой содержатся сведения о количестве материалов, выписываемых для некоторого номера, дата документа и планируемый срок выдачи материала со склада;

документы, сопровождающие изготавливаемую продукцию в процессе производства, в которых для каждой серии задается порядок выполнения операций, планируемые сроки начала и окончания операций и номер документа-наряда;

рабочий наряд;

распечатку отклонений в выполнении сроков производственных операций.

Описанная система управления производством разработана для **мелких и средних промышленных кооперативов**, поэтому в ней предусмотрены только те модули информационной системы, которые в

большой степени освобождают пользователей от выполнения ручных работ. Данная система успешно эксплуатируется в десяти промышленных кооперативах Венгрии.

УДК 681.324

## **МАШИННАЯ СИСТЕМА ДЛЯ НАХОЖДЕНИЯ ПАРТНЕРОВ ПРИ ОБМЕНЕ КВАРТИРАМИ**

**Н. Вашш, инженер (ВНР)**

**В** Венгрии в соответствии с принятой в 1968 г. Центральной программой по развитию вычислительной техники в последние годы все больше внимания уделяется применению ЭВМ непосредственно в интересах населения. Одним из примеров является использование ЭВМ в деле улучшения жилищных условий населения.

За прошедшие 25 лет республикой были достигнуты хорошие результаты в строительстве квартир. Однако за этот период произошли значительные изменения в степени износа квартир, в демографическом составе населения, в уровне потребностей населения. Перечисленные факторы в конечном счете повышают реальную потребность в квартирах. Для более качественного и своевременного удовлетворения этих нужд Центральный Комитет Венгерского Коммунистического Союза Молодежи и Главный отдел по делам квартир городского Совета г. Будапешта объявил конкурс на разработку автоматизированной информационной системы, облегчающей процесс обмена квартирами.

Система разработана (она начала функционировать с 1 октября 1981 г. в Будапеште и в нескольких больших городах Венгрии) предприятием ОКИС ССВ (предприятие при Совете промышленных кооперативов по организации и применению ЭВМ).

ОКИС ССВ занимается продажей анкет, которые заполняются клиентами, на предприятии осуществляются подготовка и обработка данных, а также бесплатное обслуживание клиентов. В ходе обслуживания ведется консультация по заполнению анкет, оказывается помощь в исправлении допущенных при заполнении анкет ошибок, выполняется бесплатная корректировка требования.

Система в режиме пакетной обработки выдает ответ в течение нескольких минут. Таким образом, при наличии соответствующих программных и технических средств можно будет обеспечить функционирование системы в диалоговом режиме.

Система имеет модульную структуру, так как она работает в разных по величине и характеристикам городах.

Учитывая потребности рынка, система разработана для следующих форм обмена: одну квартиру — на другую; одну квартиру — на две квартиры; две квартиры — на одну квартиру.



Единицей услуг системы является одно требование на обмен. Для обработки данных на ЭВМ входные данные формируются в группы:

общие данные (имя и фамилия клиента, его адрес, дата обращения к системе, номер телефона, номер паспорта заказчика, денежные условия);

данные о предлагаемой(ых) для обмена квартире(рах) (адрес, форма собственности, степень комфортабельности и расположение квартиры в жилом квартале и в зеленом массиве, уровень шума, расстояние от линий городского транспорта, наличие лифта и телефона, характерное расположение комнат и их количество, номер этажа, состояние здания, наличие горячей воды, вид отопления и покрытия пола, абонентская сумма за квартиру и прочие денежные вопросы, связанные с квартирой);

данные о желаемой(ых) квартире(ах) (требуемый район, основная площадь, форма собственности, состояние здания, степень комфортабельности квартиры, характерное расположение комнат, их количество, расположение в квартале и зеленом массиве, этаж, наличие телефона).

Количество первичных данных, описывающих отдельные случаи обмена, составляет 700 символов, что требует тщательного контроля при вводе данных в ЭВМ. На записи, принятые системой, выводится контрольный лист, при помощи которого клиент может убедиться в правильности введенных в систему данных. Значение контроля достоверности данных весьма велико, если учесть, что допущенные одним клиентом ошибки при заполнении анкеты могут неправильно ориентировать других клиентов.

Уровень услуг системы характеризуется качеством обслуживания, в том числе временем реакции системы. Это обстоятельство в значительной степени затрудняло разработку алгоритма сравнения. Кроме того, при создании системы надо было принять во внимание и ограничения, связанные с техническими возможностями и с имеющейся системой программного обеспечения. (Система функционирует на ЭВМ типа ЕС-1022 с ОЗУ объемом в 512 Кбайт, массивы расположены — в случае 20 тыс. обменов — на двух магнитных дисках емкостью в 29 Мбайт.)

Легко подсчитать, что при 20 тыс. обменах, если воспользоваться последовательной организацией массивов, потребуется 220 млн. операций ввода и 1 млрд. операций сравнения. При таких условиях обработка всего массива в среднем продолжалась бы 9—10 ч, и это время нельзя было бы сократить, применив известные методы организации массивов. Такие сроки обработки сделали бы систему нерентабельной и не был бы достигнут критерий мгновенной реакции.

Существенный интерес представляет свойство системы изменять форму своих услуг при сокращении числа сравниваемых параметров. С данной точки зрения различают две формы услуг:

двустороннее (полное) сравнение. При сравнении принимаются во внимание все параметры как предлагаемой, так и

требуемой квартир. При этом количество полностью сравниваемых адресов относительно невелико, т. е. во многих случаях эти адреса не обеспечивают широкого подбора соответствующей квартиры;

одно стороннее сравнение. Эта форма услуг была введена с 1 апреля 1982 г. с учетом опыта работы системы, функционирующей с 1 октября 1981 г. Согласно информации, полученной от клиентов, во многих случаях обмена осуществляются путем компромиссов, т. е. путем изменения первоначально заданных требований. Поэтому в первых из 12 параметров сравнения 7—8 были замаскированы, и операция сравнения была выполнена в сокращенном виде. Однако вскоре выяснилось, что такое решение неприемлемо, поскольку для каждого случая нельзя заранее предугадать маскирование. На основе анализа рекламаций было введено одно стороннее сравнение, суть его заключается в том, что для каждого клиента выдается перечень характеристик тех квартир, владельцы которых ищут квартиру данного клиента. После этого по желанию, за небольшую сумму, клиенту выдаются описания тех квартир, основные характеристики которых ему понравились. Таким образом удалось значительно расширить возможность выбора для клиента.

На уровень услуг, выполняемых системой, положительно влияет обеспечение бюро обслуживания достоверной информацией о составе фонда обмениваемых квартир, о выдаче адресов.

Повышение эффективности системы достигается за счет ведения двух вспомогательных таблиц:

таблицы анализа соотношения спроса и предложения на рынке, с помощью которой осуществляется распределение предложенных для обмена квартир по параметрам сравнения состава квартир, по параметрам сравнения состава квартир, которые обмениваются, и по параметрам сравнения запросов на обмен;

таблицы, информирующей бюро обслуживания. Она представляет собой упорядоченный по различным параметрам список предложенных для обмена квартир, статистику о выдаче адресов, алфавитный каталог заказчиков-клиентов.

Из трех логических модулей программы системы первый и третий модули написаны на языке ПЛ/ОПТ, второй модуль — на Ассемблере. Программы имеют структуру с перекрытием, а общее число входящих в систему программных фаз превышает 60.

Для уменьшения времени выполнения программ, написанных на языке ПЛ/ОПТ, использованы возможности оптимизации в данной системе программирования. Сокращение времени выполнения операции ввода-вывода достигнуто применением специальных методов организации массивов.

Система пользуется популярностью среди населения. Стоимость ее эксплуатации, даже при довольно низком уровне цен одной услуги благодаря умелой разработки системы, невысока, т. е. задача улучшения жилищных условий населения решается на предприятии ОКИС ССВ рентабельно.

Некоторые статистические данные о массивах и характеристиках системы приведены ниже.

### Сведения о составе массива

|  |       |
|--|-------|
| 1. Величина массива  | 5 131 |
| в том числе:   |       |
| желающих менять одну квартиру на две                             | 411   |
| желающих менять две квартиры на одну                             | 730   |
| 2. Характеристика предложенных квартир:                          |       |
| количество квартир   | 6 683 |
| из них:  |       |
| государственной собственности                                    | 4 223 |
| с полным комфортом   | 5 997 |
| расположены в новом районе                                       | 2 259 |
| расположены в зеленой зоне                                       | 4 097 |
| имеется телефон  | 3 554 |
| имеется больше трех комнат                                       | 914   |
| собственный жилой дом  | 350   |
| собственная квартира в двухквартирном доме                       | 194   |
| собственная квартира в 12-квартирном доме                        | 1 626 |
| собственная квартира в большом доме (свыше 12 квартир)           | 4 863 |
| 3. Характеристика желаемых квартир:                              |       |
| Количество квартир   | 7 002 |
| из них:  |       |
| государственной собственности                                    | 3 697 |
| с полным комфортом   | 6 259 |
| расположенных в новом районе                                     | 355   |
| расположенных в зеленой зоне                                     | 2 709 |
| имеется телефон  | 2 269 |
| собственный дом либо собственная квартира в двух-квартирном доме | 149   |
| в 12-квартирном доме   | 718   |
| в большом доме (свыше 12 квартир)                                | 6 135 |

### Характеристики предложения и спроса

|  |       |
|--|-------|
| Желающие менять квартиру на квартиру с более высокой степенью комфорта | 3 057 |
| Желающие менять на квартиру с более низкой степенью комфорта           | 1 417 |
| Желающие покинуть новый квартал  | 276   |
| Желающие переселиться в новый квартал                                  | 71    |
| Желающие менять государственную квартиру на частную                    | 42    |
| Желающие менять частную квартиру на государственную                    | 493   |
| Желающие менять на квартиру с большим числом комнат                    | 1 420 |
| Желающие иметь квартиру с меньшим числом комнат                        | 1 212 |
| Желающие иметь квартиру с большей площадью                             | 1 858 |
| Желающие иметь квартиру  | 1 590 |

### Сведения о выдаче адресов

|                          |       |
|--------------------------|-------|
| Всего клиентов в массиве | 6 272 |
| Из массива вычеркнуто    | 219   |
| Получили адреса:         |       |
| по полному соответствию  | 940   |
| с маскированием          | 112   |
| без маскирования         | 5 208 |
| Не получили адреса       | 12    |

## АВТОМАТИЗАЦИЯ ТАБЕЛЬНОГО УЧЕТА С ПОМОЩЬЮ МИКРОПРОЦЕССОРНОЙ ТЕРМИНАЛЬНОЙ СТАНЦИИ СМ-1621

Э. Кашимов, инженер (НРБ)  
Ю. Господинова, инженер (НРБ)

Табельный учет — актуальная проблема во всех отраслях промышленности и непроизводственной сфере. Наиболее эффективно эта проблема решается при введении скользящего графика рабочего времени, принцип которого объясняет рис. 1.

Зоны *AC*, *DE* и *FH* представляют собой зоны рабочего времени, в течение которого рабочий или служащий не находится на своем рабочем месте, причем зона *DE* включает в себя и перерыв на обед. В этих зонах организуется и так называемый баланс скользящего рабочего времени, которое не входит в отработанное время и обес-

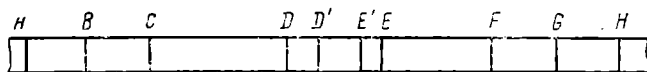


Рис. 1. Скользящий график рабочего времени:

*A* — начало утреннего скользящего рабочего времени; *B* — обычное начало рабочего дня; *C* — конец утреннего скользящего рабочего времени; *D* — начало скользящего рабочего времени обеда; *E* — конец скользящего рабочего времени обеда; *F* — начало вечернего скользящего рабочего времени; *G* — обычный конец рабочего дня; *H* — конец вечернего скользящего рабочего времени

печивает гибкость движения рабочих или служащих на предприятиях или в учреждениях. В зависимости от моментов утреннего прихода и вечернего ухода по отношению к точкам *B* и *C* баланс скользящего рабочего времени положителен (зоны *BC* и *FG*). Скользящее время может быть введено и в зону перерыва на обед *DE*. Продолжительность времени обеда (*D'E'*) гибко изменяется в зоне *DE*. Если это время будет превышено, к балансу скользящего рабочего времени в виде отрицательной величины добавляется время, на которое превышен перерыв на обед. Таким образом, в зоне *DE* баланс всегда отрицателен.

*CD* и *EF* представляют собой зоны так называемого фиксированного рабочего времени. В это время рабочий или служащий обязан находиться на своем рабочем месте. В этих зонах осуществляются координационные функции и устанавливаются связи между руководителями и исполнителями, оперативно решаются не предусмотренные планом или внезапно возникшие производственные проблемы и т. д., в связи с чем необходимо присутствие всего наличного состава предприятия, учреждения.

Скользящее рабочее время имеет ряд важных социальных и психологических преимуществ, наиболее важными из которых являются:

преодоление проблемы «нерегулярный транспорт»;

возможность решения личных проблем, требующих отсутствия на рабочем месте;

улучшение социального и психологического климата на предприятиях и в учреждениях;

сокращение потерь времени от кратковременных отлучек, связанных с уходом по личным делам;

увеличение производительности труда благодаря возможности выбора начала рабочего дня и его продолжительности каждым рабочим или служащим.

Скользящее рабочее время может учитываться автоматизированным и неавтоматизированным способами. В последнем случае необходимо участие контрольного пропускного органа, соответствующий учет моментов утреннего прихода и вечернего ухода. Однако при этом охрана занимается непривычными для нее функциями, а сам учет не отличается достаточной гибкостью.

Учет скользящего рабочего времени при помощи современных контрольно-пропускных систем, созданных на базе вычислительной техники, представляет собой совершенную форму организации табельного учета на предприятиях и в учреждениях. Широкие возможности вычислительной техники, а также большое разнообразие вычислительных средств и элементной базы способствовали созданию ряда контрольно-пропускных систем. К таким системам относятся автономные устройства для табельного учета на малых предприятиях, в бюро или конторах с персоналом до 250 человек и микропроцессорные пропускные системы для регистрации прохождения и присутствия наличного состава на предприятиях и в учреждениях с персоналом от 200 до 2000 человек. Наиболее мощные контрольно-пропускные системы управляются мини-ЭВМ или большими ЭВМ с широкими возможностями. Подобные системы обслуживают предприятия и комбинаты со штатом рабочих и служащих более 2000 человек. Данные об отработанном времени и отсутствии каждого работника регистрируются в буфере и обрабатываются, а в конце учетного периода (неделя или месяц) выводятся на магнитный носитель или по линии связи передаются к ЭВМ более высокого уровня, где они используются в качестве входных данных программ расчета заработной платы за отработанный период.

Общим признаком любой из перечисленных выше контрольно-пропускных систем является то, что с их помощью можно ввести график скользящего рабочего времени с параметрами утреннего прихода, перерыва на обед и вечернего ухода, заданными по желанию работника.

Терминальная станция СМ-1621 представляет собой типичную контрольно-пропускную систему учета скользящего рабочего времени. С ее помощью могут обслуживаться 14 категорий работников, т. е. могут быть заданы 14 видов рабочих графиков, скользящих или не скользящих, по их желанию. Терминальная станция СМ-1621 предназначена для учета наличного состава на предприятиях и в учреждениях. Она производит первоначальную обработку накопленной за день или за учетный период (месяц) информации

и автоматически определяет конец рабочего дня и учетного периода. Станция также выводит информацию на магнитный носитель (дискету) в стандартном формате. В конце учетного периода эта информация может быть выдана в виде входных данных на ЭВМ более высокого уровня, где посредством дальнейшей обработки могут быть выпущены платежные ведомости.

Терминальная станция представляет собой специализированный микропроцессорный вычислительный комплекс со следующими функциональными возможностями:

- ввода скользящего графика рабочего времени с заданными параметрами;

- обслуживания 14 категорий рабочих и служащих, работающих в одно- или двусменном режиме;

- обслуживания предприятий и учреждений с персоналом до 2250 человек;

- контроля прихода и ухода на предприятии или в учреждении посредством считывания информации с магнитной карты;

- специализированного диалога между абонентами и СМ-1621 посредством терминалов в связи с различными видами прохода;

  - ввода оператором данных для любого абонента станции;

  - индикации и распечатки данных оператором на любого абонента;

- автоматического завершения суток и учетного периода (месяца) со статистическими справками на каждого абонента;

  - вывода информации на АЦПУ;

- вывода информации на дискету: в пакетированном виде — для восстановления работы СМ-1621 после сбоя и в форматированном виде — для ввода в ЭВМ более высокого уровня;

  - форматирования дискет;

  - копирования системных дискет;

- автоматической индикации ошибок в работе периферийных устройств, а также оператора;

  - записи и контрольного считывания с магнитной карты;

  - обработки аварийных ситуаций.

Терминальная станция для контроля прохода состоит из центрального вычислительного устройства, периферийных устройств и терминалов ввода-вывода  $T_1—T_n$  для связи абонентов станции с ЦВУ.

Структурная схема СМ-1621 показана на рис. 2.

**Центральное вычислительное устройство (ЦВУ)** представляет собой специализированную микроЭВМ, имеющую процессор, постоянную и оперативную память. Базовая конфигурация специализированной микроЭВМ выполнена на базовых модулях болгарской микропроцессорной ЭВМ СМ-1627. Модуль «микропроцессор» реализован на одном ТЭЗе. Постоянная память на 16 Кбайт использует базовый модуль «постоянная память». Емкость оперативной памяти достигает 48 Кбайт. Оперативная память может иметь не более шести модулей «оперативная память», емкость каждого из них — 8 Кбайт. Терминальная станция СМ-1621 может обслужи-

вать 340 человек при 16 Кбайт оперативной памяти, 700 человек — при 24 Кбайт, 1100 человек — при 32 Кбайт, 1500 человек — при 40 Кбайт, 2250 человек — при 48 Кбайт.

ЦВУ осуществляет первоначальное накопление и обработку данных, регистрирует входные и выходные транзакции абонентов при специализированном диалоге с терминалами ввода-вывода, вычисляет баланс скользящего рабочего времени, автоматически завершает сутки и учетный период, по желанию потребителя выводит на печать или на дискету статистические данные в конце дня или

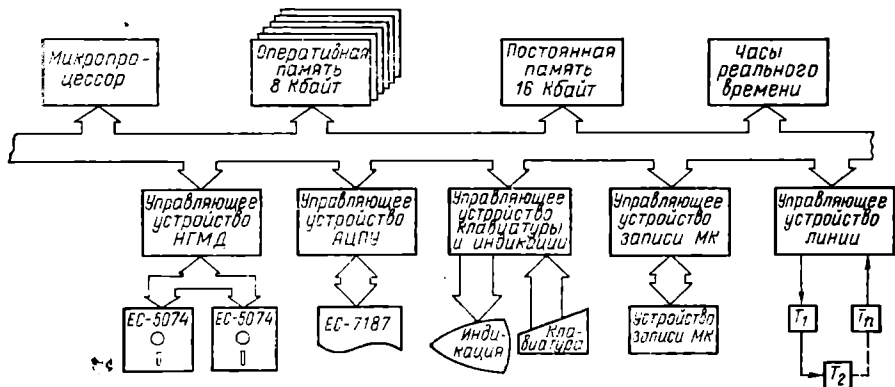


Рис. 2. Структура технических средств системы

учетного периода, причем операции получения двух носителей (листинг или дискета) не дублируются. На базе полученной информации может быть произведена автоматическая обработка на более высоком уровне, если используется носитель-дискета, или ручная, если применяется листинг.

**Периферийные устройства** к ЦВУ дополняют конфигурацию специализированного микропроцессора вычислительного комплекса: печатающее устройство ЕС-7187 для печати всех транзакций, статистических данных, вычисленных во время автоматического завершения рабочего дня и учетного периода, а также для печати операторских вмешательств и коррекций. На печать выводятся также некоторые диагностические сообщения о неправильном функционировании периферийных устройств. Скорость печати ЕС-7187 — 30 зн./с;

два НГМД ЕС-5074. Ежечасо автоматически или произвольно по желанию оператора на системную дискету НГМДО передается пакетированная информация, накопленная в оперативной памяти. В конце суток по желанию оператора на рабочую дискету в НГМДО может быть выведено содержание оперативной памяти в стандартном формате для обработки на ЭВМ более высокого уровня. Посредством НГМД1 форматируются новые дискеты и копируются системные. Емкость односторонней дискеты с единичной плотностью записи — 256 Кбайт;

устройство записи на магнитные карты — для записи и контрольного считывания специализированных пропусков с магнитной записью. Плотность записи на магнитную карту — 3 бит/мм; цифровая и функциональная клавиатуры для проведения операторского диалога;

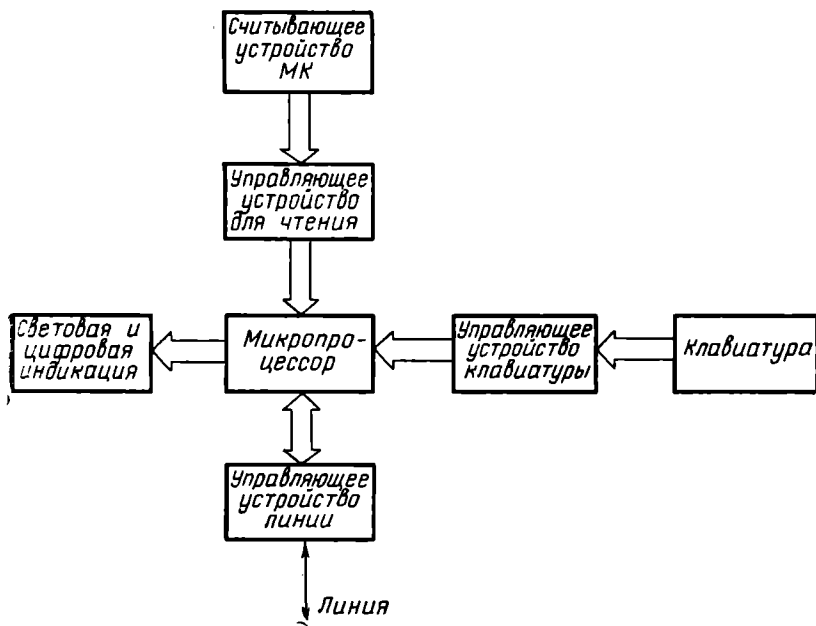


Рис. 3. Структура терминала системы

цифровая и световая индикация для контроля и проверки специализированных функций, а также для индикации режимов работы, неправильного функционирования периферийных устройств и при вмешательствах оператора.

Контроллеры для связи микропроцессора ЦВУ с периферийными устройствами построены с использованием параллельного интерфейсного адаптера, асинхронного интерфейсного серийного адаптера и синхронного интерфейсного серийного адаптера.

Терминалы ввода-вывода терминальной станции контроля доступа СМ-1621 представляют собой интеллектуальные микропроцессорные терминалы. Они являются закрытыми микропроцессорными системами с проблемно-ориентированным прикладным микропрограммным обеспечением для обслуживания диалога между терминалом и ЦВУ. Структурная схема терминала ввода-вывода представлена на рис. 3, а общий вид клавиатуры — на рис. 4. Цифровая индикация терминала показывает текущее астрономическое время.

Диалог между терминалом ввода-вывода и ЦВУ осуществляется посредством магнитной карты, имеющейся у каждого работни-



ка: она вставляется в устройство считывания с магнитных карт. В результате правильно считанной микропроцессором терминала информации осуществляется обмен с ЦВУ. ЦВУ отвечает абоненту, выводя на цифровую индикацию терминала его баланс скользящего рабочего времени. Абонент продолжает диалог посредством нажатия соответствующей клавиши. ЦВУ завершает диалог с терминалом, подтверждая световой индикацией правильность нажатой клавиши и индицируя данные об абоненте, осуществившем обмен.

При наличии ошибки со стороны абонента в диалоге, т. е. при неверно нажатой клавише, ЦВУ отвечает терминалу, а терминал выдает на цифровую индикацию предупредительный сигнал. После окончания диалога и выемки магнитной карты из устройства считывания терминала на цифровой индикации восстанавливается текущее астрономическое время.

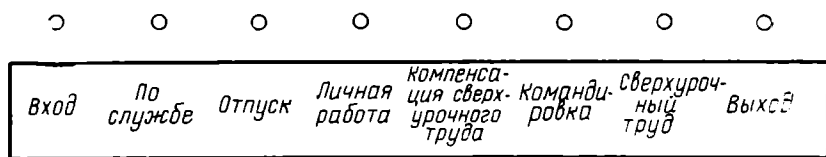


Рис. 4. Клавиатура терминала

Терминалы ввода-вывода подключены к ЦВУ через последовательный интерфейс с расстоянием между терминалами и ЦВУ до 2000 м. Скорость обмена по линии — 9600 бод, что дает возможность подсоединять и обслуживать большое количество терминалов на терминальной станции.

При времени обслуживания абонента, равном 3 с, теоретически может работать 180 терминалов ввода-вывода. Имея в виду, что один терминал в часы «пик» может обслуживать 100—250 человек, к терминальной станции максимально может быть подключено 32 терминала.

Интерфейсной связью между терминалами ввода-вывода, а также между ЦВУ и первым и последним терминалами является двухпроводная телефонная линия.

**Режимы работы СМ-1621.** Терминальная станция контроля прихода-ухода СМ-1621 может работать в следующих режимах: генерации, эксплуатации, изменений генерированной системы, восстановления генерированной системы, теста.

Основные режимы работы СМ-1621 — генерация и эксплуатация.

В режим «генерация» станция входит при первоначальном пуске. Печатающее устройство ЦВУ отпечатывает последовательность сообщений-вопросов, на которые осуществляющий генерацию оператор отвечает в диалоговом режиме, используя цифровую часть функциональной клавиатуры ЦВУ. Своими ответами на задаваемые ЦВУ вопросы оператор задает параметры и условия для работы

станции, например число категорий, параметры скользящего рабочего времени для этих категорий, коэффициенты ночного и сверхурочного труда, официальные праздники и пр. В конце генерации осуществляется автоматическое абонирование контингента персонала, обслуживаемого терминальной станцией, и переход в режим «эксплуатация».

С входом СМ-1621 в режим «эксплуатация» цифровая индикация терминалов ввода-вывода показывает заданное оператором в режиме «генерация» текущее астрономическое время. Терминалы готовы к диалогу с ЦВУ. В режиме «эксплуатация» станция выполняет следующие функции: регистрирует входные и выходные транзакции абонентов, вычисляет баланс скользящего рабочего времени, автоматического завершения суток и учетного периода, выводит информацию на печать и дискету, записывает данные на магнитные карты, осуществляет операторские вмешательства при коррекциях и выдает справки об абоненте или группе абонентов и пр. В этом режиме могут быть абонированы новые и устранены старые абоненты станции. Если в режиме «генерация» абонирование контингента абонентов не осуществлено, то абонирование можно выполнить в режиме «эксплуатация», но не по блокам, а одиночно. При отключении напряжения питания содержание полупроводниковой оперативной памяти в ЦВУ сохраняется в течение 3 ч посредством аккумуляторных батарей. Если в течение этого времени напряжение питания восстанавливается, терминальная станция автоматически входит в режим «эксплуатация» с актуализированным астрономическим временем (так как часы ЦВУ имеют собственное аккумуляторное питание) и с сохраненными данными об абонентах.

В режиме «изменения генерированной системы» предоставляется возможность изменять некоторые параметры, заданные в режиме «генерация». ЦВУ опять вводят в диалоговый режим с оператором. В этом диалоге оператор вводит новые и устраняет старые категории, изменяет параметры графика рабочего времени некоторых категорий, изменяет коэффициенты и продолжительность сверхурочного и ночного труда и пр. В конце этого режима терминальная станция снова переходит в режим «эксплуатация».

Режим «восстановление генерированной системы» используется при отключении напряжения питания на время свыше 3 ч. В результате этого содержание оперативной памяти ЦВУ, являющейся буфером для данных об абонентах, теряется, и восстановление буфера, а также ввод терминальной станции в нормальный режим работы осуществляются с последней системной дискеты, полученной из НГМДО, автоматически или путем операторского вмешательства. После считывания с нее данных и краткого диалога между ЦВУ и оператором станция переходит в режим «эксплуатация».

Режим «тест» предназначен для отладки и профилактики СМ-1621. В постоянной памяти ЦВУ ПЗУ с самым старшим адре-

сом заменяется тестовым ПЗУ. Посредством устройства считывания с перфоленты FS-1501 и инженерного пульта, представляющего собой сервисное оборудование к терминальной станции, в оперативную память ЦВУ вводятся тестовые программы, осуществляющие проверку памяти, управляющих устройств, периферийных устройств и линии к терминалам станции. Режим «тест» реализуется только сервисным инженером.

Возможности рассматриваемой терминальной станции СМ-1621 в отношении статистической обработки информации достаточны, причем в конце учетного периода АЦПУ выдает документ, близкий по форме и содержанию к документам бухгалтерии, предназначенным для учета труда. Кроме того, благодаря выводу на дискету информации в стандартном формате станция может быть связана с ЭВМ более высокого уровня для расширения статистической обработки данных. Данные терминальной станции в конце учетного периода могут быть использованы для создания заключительного документа об оплате отработанного времени — ведомости заработной платы.

СМ-1621 внедрена на нескольких предприятиях в НРБ. Наиболее богатый опыт ее эксплуатации (с 1981 г.) накоплен в Центральном институте вычислительной техники (г. София), являющемся разработчиком системы. Выпускается СМ-1621 комбинатом «Оргтехника» (г. Силистра) — одним из предприятий ДСО «ИЗОТ». С помощью системы решены такие административно-управленческие проблемы, как затрата компенсируемого или оплачиваемого сверхурочного труда, ввод скользящего графика рабочего времени в пределах, заданных руководством института, выход по личным делам с целью решения каждым сотрудником собственных проблем и пр.

Создан также пакет прикладных программ для статистической обработки данных на ЦВУ СМ-1621 для ЭВМ более высокого уровня. Пакет программ дает реальную картину отработанного и неотработанного времени абонентов терминальной станции, а также помогает решению проблемы соблюдения установленных границ рабочего времени в соответствии с кодексом о труде и укреплению трудовой дисциплины.

УДК 681.325.5-181.5:61

### ИСПОЛЬЗОВАНИЕ МИКРОПРОЦЕССОРА СМ-1630 В ОБЛАСТИ ЛУЧЕВОЙ ТЕРАПИИ

З. Мачке, д-р естеств. наук (ГДР)

Польза от лучевой терапии отмечается в случае, если удастся осуществить облучение, при котором раковая опухоль получает достаточную лечебную дозу с одновре-

менным сбережением ткани, окружающей опухоль. Точное определение дозы и ее распределение во время лечения служит исходной точкой для полного использования всех преимуществ и возможностей современных установок облучения. При расчете дозы должны быть учтены внешние размеры пациента и его органов, расположение опухолей. Целью расчетов является индивидуальное планирование облучения для каждого пациента. При этом помимо парамет-

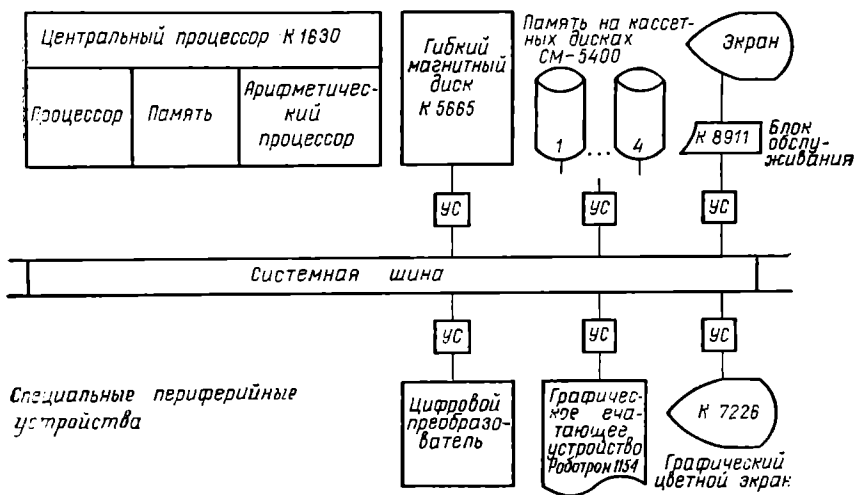


Рис. 1. Конфигурация аппаратного обеспечения системы планирования облучения «Роботрон DOPSY»

ров установки для облучения учитываются индивидуальные топометрические соотношения пациента в поле облучения. Косвенное или непосредственное использование топограмм, разработанных с помощью ЭВМ, очень важно для планирования облучения. При таком объеме обрабатываемых данных определить распределение дозы в необходимые (очень короткие) сроки можно только с использованием специальных систем обработки данных, например системы «Роботрон DOPSY».

Система содержит (рис. 1) микроЭВМ СМ-1630 в составе процессора и арифметического процессора для операций с плавающей запятой, работающих со словами в 16 бит и памятью емкостью до 256 Кбайт, ВЗУ на кассетных дисках (один постоянный диск и один кассетный диск емкостью 25 Мбит), ВЗУ на гибком магнитном диске (два привода по 3,2 Мбит), универсальный терминал с дисплеем для ввода и вывода данных и для диалогового режима, цифровой преобразователь для ввода контуров пациента и геометрических данных, графический дисплей для вывода контура пациента, геометрических параметров облучения и распределения дозы, печатающее устройство для вывода протокола вычислений, контура пациента и рассчитанного распределения дозы (рис. 2), операцион-

ную систему МООС для режима работы с несколькими пользователями.

Операционная система выполняет помимо расчетов для планирования облучения другие работы. Для этого предлагаются дополнительные периферийные устройства: устройство считывания с перфокар, ленточный перфоратор, устройство считывания с перфокарта, накопители на магнитной ленте и дополнительные терминалы.

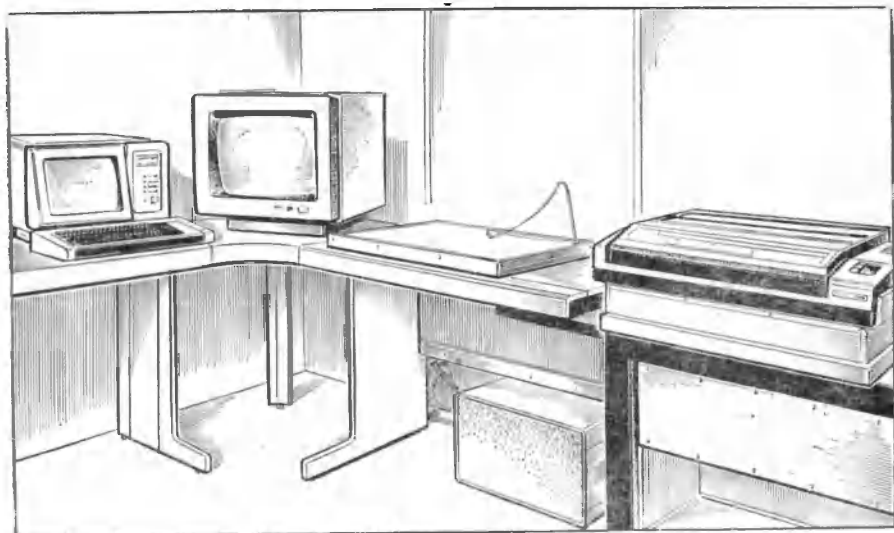


Рис. 2. Общий вид системы планирования облучения «Роботрон DOPSY» без ЭВМ

Программное обеспечение разработано с учетом многолетнего опыта применения лучевой терапии, накопленного в результате сотрудничества физиков и медиков. Пакет рабочих программ БСБ-4 создан Вычислительным центром Исследовательского центра молекулярной биологии и медицины в Берлине. В пакете приняты во внимание все встречающиеся в практике виды излучения, энергии излучения и методы облучения. Для работы с системой не требуется специальных знаний в области электронной обработки данных. Обслуживание системы осуществляется в диалоговом режиме с использованием естественного языка. Текст задаваемых программой вопросов и комментариев может быть составлен согласно представлениям пользователя.

Пакет программ БСБ-4 содержит следующие подпрограммы:

**ДОСМАТ** — программа расчета матрицы дозы для одного поля облучения и одного поперечного сечения;

**МАНТЕЛ** — программа расчета матрицы оболочного поля в одной плоскости;

**БРАХИ** — программа для распределения дозы для брахитерапии в одной плоскости;

**ВЕРСИО** — программа для количественного определения одной комбинации заранее рассчитанных полей;

**ОПТИМ** — программа для расчета весов полей с учетом разграничений доз в выбираемых точках;

**БИО** — программа для учета относительной биологической эффективности и номинальной стандартной дозы;

**ПЛАН** — программа для представления выбранного плана облучения и распределения дозы графопостроителем и печатающим устройством;

**ДАТПАТ, ДАТФЕЛ, ДАТКВ и ДАТМАН** — программы для ввода и исправления данных о пациенте, поля, поперечного сечения и оболочного поля;

**ДИМ 3** — расчеты и изображения в плоскостях, не являющихся плоскостями поперечного сечения;

**ЦТК** — программа для использования данных томограмм, полученных с помощью ЭВМ;

**МЕСДАТ** — программа для ввода, исправления и обслуживания измеряемых данных.

Ввод всех необходимых для расчета данных осуществляется с помощью дисплейного терминала и цифрового преобразователя. С дисплейного терминала осуществляется ввод данных в режиме диалога в виде простых вопросов и простых стандартных ответов. Для некоторых вопросов предусмотрены в качестве ответов постоянные величины, требующие только подтверждения. По желанию оператора могут быть выведены разъясняющие тексты. Возможен возврат в диалоге и тем самым изменение введенных данных.

С помощью терминала вводятся:

данные пациента (имя, фамилия, диагноз, поперечное сечение области облучения и т. д.);

данные прибора (номер прибора, энергия, расстояние между источником и центром поворота и т. д.);

данные поля (величина поля, угол клина, расстояние между источником и поверхностью и т. д.);

специальные вычислительно-технические данные (растр, место в памяти и т. д.);

дозиметрические данные (желаемая максимальная доза, дслеие и т. д.).

Цифровой преобразователь предназначен для ввода контуров и данных для облучения. С помощью командных полей можно, с одной стороны, вызвать части программы, а с другой — ввести определенные стандартные величины.

Вводимыми данными являются, например, позиция поля, угол облучения, изоцентры, угол качания. Все введенные данные контура и облучения сразу высвечиваются на графическом экране. Во время ввода они могут быть изменены после их оценки по изображению на экране. Если для планирования облучения используются сечения с томограмм, полученные с помощью ЭВМ, то соответст-

вующие контуры и данные могут быть прямо введены в процессор. Ввод параметров облучения (например, вид источника, расположение источника или носителя, активность источника) осуществляется при брахитерапии таким же образом. Сразу после ввода первого поля начинается процесс расчета, даже если за первым полем еще следуют другие поля. Таким образом, сокращается срок между окончанием ввода и выводом первого распределения дозы. Для источника излучения на базе  $\text{Co}^{60}$  расчеты производятся на основе аналитических функций для доз по глубине и поперечному распределению, которые специально согласовываются с особенностями каждого прибора. В случае квантового, электронного или нейтронного излучения высокой энергии вводимые данные измерения являются основой для расчета дозы облучения. При подвижном облучении распределение дозы рассчитывается сложением множества отдельных стационарных полей. Сразу по окончании расчета на графическом экране высвечивается изображение поперечного сечения области облучения с расчетными значениями изодоз. Изображение может быть напечатано в масштабе 1 : 1 в серых тонах.

Контур опухоли маркируется. На плане изодоз указываются фамилия пациента, номер памяти, в которой находится план изодоз в системе, плоскость поперечного сечения, в которой произведен расчет, дата облучения, номер счета, маркировка положения пациента и данные излучения, центральный луч и данные полей, угол облучения, номер поля, изоцентр, диапазон качания, первоначальный и конечный угол качания, расположение примененных клиновых фильтров и углы клина, расположение индивидуальных и стандартных уравнивательных блоков, направление и угол отклонения излучателя.

Вывод протокола-счета осуществляется с помощью печатающего устройства. Он содержит введенные данные (доза, базовая доза и время облучения или количество качаний, скорость качания и т. д.) и общие данные (название клиники, фамилия лица, отвечающего за данный план, и т. д.). Все данные одного пациента вводятся в память и сохраняются там до тех пор, пока не вводятся данные нового пациента. По вызову номера памяти все записанные в ней данные высвечиваются на графическом экране и дисплейном терминале. В стандартном варианте в памяти хранятся данные 100 пациентов. По желанию могут быть сохранены данные нескольких сотен пациентов. Примерно для десяти пациентов хранятся расчетные матрицы дозы, так что для них в любой момент времени могут быть вызваны не только введенные в систему данные, но и распределение дозы облучения и расчетные абсолютные значения. Если расчетное распределение дозы не соответствует требованиям, то могут быть изменены исходные параметры. После этого снова производятся расчеты. По вызову номера памяти отдельные исходные параметры могут быть изменены с помощью дисплейного терминала или цифрового преобразователя. Измененные данные можно опять записать в память. Кроме того, можно постоянно стирать и заново вводить поля облучения.

**ИСПОЛЬЗОВАНИЕ МИКРОЭВМ В СИСТЕМАХ БУХГАЛТЕРСКОГО УЧЕТА**

С. Христова, инженер (НРБ)

Эффективность управленческого труда можно повысить путем максимальной механизации и автоматизации обработки экономической информации, что требует современных высокопроизводительных технических средств, удовлетворяющих следующие требования:

- гарантированную достоверность обрабатываемой информации;
- наличие оперативной памяти достаточного объема;
- наличие внешних запоминающих устройств для ввода постоянных данных и хранения полученных результатов, которые могут быть использованы в последующих расчетах;
- возможность распечатки различных по форме и виду документов.

Для создания таких технических средств наиболее целесообразно использование микропроцессорных ЭВМ.

На базе микропроцессорного семейства МОП интегральных схем серии 600 в НРБ разработан и внедрен в производство микропроцессорный набор СМ-50/40-3, включающий в себя:

- микропроцессорный модуль, созданный на базе 8-битового параллельного микропроцессора. Модуль работает с тактовой частотой 1 мГц и адресует до 64 Кбайт памяти;

- статическую оперативную память емкостью 4 Кбайт на базе элементов  $1024 \times 1$  бит с временем доступа не более 500 нс;

- динамическую оперативную память емкостью 8 Кбайт на базе элементов  $4096 \times 1$  бит с временем доступа не более 500 нс;

- постоянную память емкостью 16 Кбайт на базе программируемых РОМ  $1024 \times 8$  бит;

- модуль связи с устройством памяти на гибком магнитном диске;

- модуль связи с буквенно-цифровым печатающим устройством;

- модуль связи с клавиатурой матричного типа с максимальным числом шин — 16 входных и 16 выходных;

- модуль связи с серийным интерфейсом ИРПС и модемом;

- дополнительные модули.

Все модули СМ-50/40-3 выполнены на элементах, производимых в НРБ или странах — членах СЭВ.

В качестве периферийных устройств для микроЭВМ обычно используются:

- алфавитно-цифровое печатающее устройство ЕС-7187 с возможностью подключения электромеханической приставки для передвижения карточек и сберегательных книжек (скорость печати — 30 зн./с, максимальное количество символов — 158 при максимальной ширине бумаги 380 мм, устройство работает с различными видами печатающих дисков (кириллица — прописные и строчные буквы, латиница — прописные и строчные буквы и смешанный вид —



кириллица и латиница — прописные буквы), с горизонтальной и вертикальной табуляцией в оба направления);

запоминающее устройство на гибком магнитном диске ЕС-5074 со скоростью обмена данными 250 Кбит/с, с частотным методом записи с плотностью 128 бит/мм, временем перехода головки записи считывания с одной дорожки на другую — 10 мс максимум; устройство работает с дискетами ИЗОТ 5257Е с односторонней записью, с единичной плотностью и максимальной плотностью на одну дискету — 250 Кбайт;

модем ЕС-8005 со скоростью обмена данными 600 и 1200 бит/с в асинхронном или синхронном режиме по четырехпроводной линии или двухпроводной линии в полудуплексе.

На основе микропроцессорных модулей СМ-50/40-3 и перечисленных периферийных устройств создана серия микроЭВМ для обработки экономической информации — устройство подготовки данных СМ-6913, терминал СМ-1613 и модернизированное устройство СМ-6913. Они построены на модульном принципе с использованием унифицированных плат и механических конструкций. Отдельные виды микроЭВМ отличаются емкостью оперативной памяти, количеством периферийных устройств, организацией данных и возможностями их обработки.

СМ-6913 включает в свою конфигурацию управляющее устройство с оперативной памятью для данных и программ 12 Кбайт, два устройства ЕС-5074, одно устройство ЕС-7187, клавиатуру и средства индикации.

СМ-1613 состоит из управляющего устройства с оперативной памятью 24 Кбайт, три устройства ЕС-5074, одно устройство ЕС-7187, одно устройство ЕС-8005, клавиатуры и средства индикации.

В модернизированном устройстве СМ-6913 оперативная память увеличена до 16 Кбайт и существует возможность подключения одного или двух запоминающих устройств на гибких магнитных дисках.

Описанные микроЭВМ работают с данными, организованными в файлы с прямым и последовательным доступом. Файлы содержат два типа элементов: реальные числа с естественной запятой, определенные как числовые регистры, и наборы символов — текстовые регистры. Длина числового регистра — 8 байт, в них записывается 14 десятичных разрядов, знак и порядок. В СМ-1613 существует возможность организации сегментов числовых регистров в четыре уровня. Информация в текстовые регистры записывается по одному символу на байт. Длина текстовых регистров в СМ-6913 постоянная — 32 байта, а в СМ-1613 — переменная — от 1 до 128 байт.

Во время обработки числовые элементы участвуют во всех арифметических и логических операциях. Текстовые элементы в СМ-6913 загружаются предварительно и остаются постоянными до окончания определенного этапа обработки, после чего их можно изменить. СМ-1613 обеспечивает выполнение загрузки, обмена, разделения, слияния и сравнения текстовых элементов.

Данные обрабатываются с помощью прикладных программ, написанных на специально разработанных входных символических языках, которые проблемно ориентированы на обработку экономической информации и учитывают функциональные возможности и конфигурации микроЭВМ.

Операторы входных языков выполняют:

процедуры ввода-вывода (для ввода с клавиатуры, вывода на индикацию или печать, обмена данными с ЗУГМД или по линии); присвоение значений переменным;

арифметические операции (сложение, вычитание, умножение, деление), вычисления процента и промилле с указанием количества разрядов после десятичной запятой и вида округления);

переходы — условные и безусловные;

вызов программы, обработки прерываний и пр.

При работе с микроЭВМ оператор вводит номер нужной прикладной программы, которая считывается с дискеты в оперативную память. При выполнении потребительской программы осуществляется диалог с оператором и контролируются его действия, проверяется правильность вводимых данных и при обнаружении ошибок выдаются сообщения. При обработке и печати первичных документов данные, необходимые для последующей обработки, записываются на дискеты. Информация из таких файлов на дискетах может быть использована и для других программ для получения различных видов справок и сведений. Обмен данными в СМ-1613 также осуществляется с помощью прикладных программ, определяющих действия над полученной информацией или режим обмена и передачи по линии.

В микроЭВМ предусмотрены дополнительные режимы работы:

формирование дискет, при котором записывается необходимая служебная информация для адресации секторов и для идентификации дискет в соответствии с требованиями стандартов;

ввод, редактирование и запись на дискеты текстов и прикладных программ в символическом виде с возможностью замены, добавления и стирания строк и символов, распечаткой текстов или частей текстов и пр.;

трансляция введенной в символическом виде программы в объектный код и его запись на дискету, с синтаксическим контролем символической программы и выдачей сообщений об ошибках;

копирование частей одной дискеты на другую или на ту же самую дискету;

полная перезапись информации с одной дискеты на другую;

контрольное тестирование для проверки функциональной пригодности микропроцессора, оперативной и постоянной памяти, периферийных устройств и носителей;

Эти режимы предоставляют возможность осуществлять обработку прикладных программ и предварительную подготовку данных и носителей.

Созданные микроЭВМ характеризуются современной архитектурой и высокими эксплуатационными параметрами; удобным и до-

ступным проблемно-ориентированным языком для описания документов; возможностью вторичной обработки информации; возможностью передачи информации по каналу или посредством машинного носителя (дискеты) к другому техническому средству — микро-, мини- или большой ЭВМ.

На базе описанных микроЭВМ и соответствующих пакетов прикладных программ разработаны и внедрены в производство разнообразные системы. Среди них можно назвать систему для автоматизации обработки финансово-бухгалтерской информации на промышленных и сельскохозяйственных предприятиях. Система охватывает учет произведенной и реализованной готовой продукции, вычисление амортизационных отчислений основных средств, вычисление заработной платы рабочих и служащих и др. Осуществляется фактурирование продукции и выдаются справки о ежедневном (или за определенный период времени) производстве, объединенные по соответствующим функциональным группам, складам, цехам и счетам.

Другая система служит для механизированного вычисления израсходованной электроэнергии гражданами и хозяйственными организациями.

На микроЭВМ СМ-1613 построена система для информационного управления складским хозяйством на промышленных предприятиях. С помощью этой системы обрабатываются и печатаются приходные и расходные накладные и на основе накопленной информации составляются ежедневные отчеты о приходе, расходе и наличии продукции по номенклатурным номерам в количестве и размере. Кроме того, выдаются справки о появившихся отклонениях от нормативов запаса каждого материала. При этом облегчается проведение инвентаризации и ускоряется учет материалов и получение ведомостей в конце учетного периода.

На этой же микроЭВМ построена система для механизации обработки информации в государственных сберегательных кассах. Устройства монтируются на местах обслуживания. С клавиатуры микроЭВМ вводятся данные, идентифицирующие каждого клиента, и данные об операции, соответствующая информация автоматически печатывается в журнале, карточке и сберегательной книжке клиента. Во время ежедневной обработки на дискеты записывается информация, используемая для автоматической выдачи учетных и статистических сведений и отчетов за различные периоды времени.

Для создания системы можно использовать любую из описанных микроЭВМ. Выбор ее определяется объемом информации, подлежащей обработке, кругом решаемых задач, удобствами работы оператора и возможностями программирования, предлагаемыми в данной конфигурации.

Работа любой информационной системы на базе микроЭВМ может быть разделена на три этапа:

первоначальная загрузка файлов с полупостоянными данными и исходными данными для работы системы;

ввод и обработка текущих данных, печать первичных документов и одновременно с этим актуализация рабочих файлов;

СК «ЭЛЕКТРОСНАБЖЕНИЕ» — СОФИЯ ФИЛ.1 ИНКАС.066 ЖУРНАЛ 002 \*\*\*\*\*  
 ФАКТУРА ИЗРАСХОДОВАННОЙ ЭЛЕКТРОЭНЕРГИИ ЭЛ. № 965 ГЕОРГИЙ НИКОЛОВ  
 АБОНЕНТ 106502135 ЗА М. 2, 1982 г. ЖК. МЛАДОСТ БЛ 331 Вх.А. Ап.2 \*\*\*\*\*

| Ц. ЛВ. | СТАРОЕ П. | НОВОЕ П. | РАЗНИЦА | СЛУЖ. | ЗА НАЧ. | СУММА ЛВ. | ВСЕГО ЛВ. |
|--------|-----------|----------|---------|-------|---------|-----------|-----------|
| 0,032  | 8390      | 850      |         |       |         |           | ОШИБКА    |

СК «ЭЛЕКТРОСНАБЖЕНИЕ» — СОФИЯ ФИЛ.1 ИНКАС.066 ЖУРНАЛ 002 \*\*\*\*\*  
 ФАКТУРА ИЗРАСХОДОВАННОЙ ЭЛЕКТРОЭНЕРГИИ ЭЛ. № 965 ГЕОРГИЙ НИКОЛОВ  
 АБОНЕНТ 106502135 ЗА М. 2, 1982 г. ЖК. МЛАДОСТ БЛ 331 Вх.А. Ап.2 \*\*\*\*\*

| Ц. ЛВ. | СТАРОЕ П. | НОВОЕ П. | РАЗНИЦА | СЛУЖ. | ЗА НАЧ. | СУММА ЛВ. | ВСЕГО ЛВ. |
|--------|-----------|----------|---------|-------|---------|-----------|-----------|
| 0,032  | 8390      | 8560     | 170     | 0     | 170     | 5,44      |           |
| 0,010  | 2250      | 2300     | 50      | 0     | 50      | 0,50      | 5,94      |
| ***    | ***       | ***      | ***     | ***   | ***     | ***       | ***       |

Рис. 1. Квитанции абонента

извлечение данных из рабочих файлов по определенному признаку и печать необходимых вторичных документов (справок, сведений и пр.).

В качестве примера может быть рассмотрена система механизированного вычисления израсходованной электроэнергии, созданная на базе СМ-6913. В данном случае полупостоянными данными являются номер, имя и адрес каждого абонента, стоимость электроэнергии для бытовых и промышленных целей, дневной и ночной тарифы и др. Эти данные, а также сведения о количестве электроэнергии, израсходованном каждым абонентом, до определенного момента записываются на дискеты. После подготовки таких файлов начинается их текущая обработка.

Оператор вводит с клавиатуры микроЭВМ только номер абонента и новое показание счетчика. На базе старого показания производится вычисление израсходованной электроэнергии, стоимость которой должна быть оплачена, и отпечатывается квитанция для клиента (рис. 1). Предусмотрен программный контроль вводимой информации по разрядности и границам номеров абонента, а также при необходимости по превышению нового показания счетчика над старым и др. В случае ошибки слово «ошибка» и ввод информации для того же абонента повторяются. После печати квитанции производится актуализация рабочих файлов (запись нового показания счетчика и количества израсходованной электроэнергии).

В конце текущей обработки на дискетах находится информация, которая может быть использована для автоматического получения различных видов справок (рис. 2), например о количестве и стоимости израсходованной электроэнергии за данный период (для промышленных и бытовых целей, дневной и ночной тариф), о сумме, которую должен собрать и отчитаться каждый инкассатор, и др. Перед началом обработки на следующий период можно изменить данные об имени и адресе абонента или стоимости электроэнергии.

СК «ЭЛЕКТРОСНАБЖЕНИЕ» РЕКАПИТУЛЯЦИЯ № —1 МЕС. 2/1982 г.  
СОФИЯ ФИЛИАЛ—1 ПАРТИЯ—1095 ОТРАСЛЬ—0

| ТАРИФЫ<br>ЦЕНА*<br>КОД | 1<br>КВТЧ | КИЛО-<br>ВАТТ-<br>ЧАСЫ | *ЛЕВА     | *СНИЖЕ-<br>НИЕ*<br>ЛЕВА | УВЕЛИ-<br>ЧЕНИЕ ЗА<br>COS ЛЕВА | СНИЖЕ-<br>НИЕ ЗА<br>COS ЛЕВА | СУММА<br>ОПЛАТЫ<br>ЛЕВА |
|------------------------|-----------|------------------------|-----------|-------------------------|--------------------------------|------------------------------|-------------------------|
| 1                      | 0,076     | 1218000                | 92568,00  | 925,68                  | 0,00                           | 0,00                         | 91642,32                |
| 2                      | 0,034     | 3186000                | 108324,00 | 1083,24                 | 0,00                           | 0,00                         | 107240,76               |
| 3                      | 0,012     | 1932000                | 23184,00  | 231,84                  | 0,00                           | 0,00                         | 22952,16                |
| 4                      | 0,060     | 0                      | 0,00      | 0,00                    | 0,00                           | 0,00                         | 0,00                    |
| 5                      | 0,038     | 0                      | 0,00      | 0,00                    | 0,00                           | 0,00                         | 0,00                    |
| 6                      | 0,032     | 856000                 | 27392,00  | 0,00                    | 0,00                           | 0,00                         | 27392,00                |
| 7                      | 0,010     | 217000                 | 2170,00   | 0,00                    | 0,00                           | 0,00                         | 2170,00                 |
| ВСЕГО                  |           | 7409000                | 253638,00 | 2240,76                 | 0,00                           | 0,00                         | 251397,24               |

Рис. 2. Образец справочной таблицы

Аналогичным способом производится обработка информации и при использовании других систем.

Применение микроЭВМ в системах бухгалтерского учета обеспечивает:

создание высокопроизводительных и надежных систем регистрации, первичной обработки и распространения полученной информации;

разработку эффективных систем для решения широкого круга экономических задач и оперативного управления отдельными хозяйственными единицами;

условия для создания иерархических информационных систем с подключением микроЭВМ в различных уровнях;

разгрузку больших ЭВМ от несвойственных для них задач, в результате чего уменьшаются общие расходы на машинную обработку информации.

УДК 681.324

#### **ПРИМЕНЕНИЕ КОМПЛЕКСА КОДИАК ПРИ ПРОИЗВОДСТВЕ СМ ЭВМ**

**В. П. Сидоренко**, канд. техн. наук (СССР)  
**О. Д. Руккас**, канд. техн. наук (СССР)  
**М. С. Берштейн**, канд. техн. наук (СССР)  
**Е. Н. Чичирин**, инженер (СССР)

Непрерывное усложнение схем цифровых и аналоговых средств вычислительной техники при одновременном увеличении объемов их производства привело к недопустимому разрыву между возникшими потребностями и существующими возможностями методов и средств контроля и диагностики этих объектов. Этот разрыв объясняется прежде всего трудностями получения качественных тестовых последовательностей при использовании традиционных методов контроля с хранением и подачей на вход блока заранее синтезированных тестовых наборов. Разрабатываются новые оригинальные методы контроля, облегчающие создание испытательных последовательностей и обеспечивающие необходимую полноту контроля за счет повышения скорости смены тестовых воздействий и резкого увеличения их количества. Например, высокопроизводительный способ контроля, предусматривающий использование технологических образцов изделий [1], может обеспечить эффективную проверку блоков ЭВМ, однако применение его в производстве с широкой номенклатурой затруднено. Все большее распространение получают устройства с аппаратными генераторами испытательных последовательностей и компактным представлением ответных реакций объекта контроля [2, 3], но созданное

оборудование не позволяет строить на его базе автоматизированные технологические комплексы.

Решению задачи обеспечения высокой эффективности контрольно-отладочных процессов в производственных и лабораторных условиях способствует разработанный в Киевском научно-исследовательском и конструкторском институте периферийного оборудования контрольно-диагностический автоматизированный комплекс КОДИАК [4].

Комплекс обеспечивает контроль и диагностику логических и аналоговых блоков по программам, хранимым во внешней памяти, автоматизацию кодирования и отладки программ проверки с использованием проблемно-ориентированного языка ЗАЛП и средств отладки, автоматизированное формирование документации, содержащей тексты программ проверки, справочные материалы (используемые при диагностировании блоков), оформленные в соответствии с требованиями ЕСКД, сбор и вывод на машинный носитель данных об объекте контроля для последующей статистической обработки и автоматизированную настройку программного обеспечения при изменении состава оборудования.

Комплекс КОДИАК содержит управляющую ЭВМ типа СМ-4 (с ОЗУ не менее 32 Кслов и внешней памятью на дисках емкостью не менее 5 Мбайт), а также до 16 проблемно-ориентированных устройств для контроля блоков логических (УКБЛ) и аналоговых (УКБА). Стыковка УКБЛ и УКБА с СМ-4 осуществляется через устройство СМ-5402 с помощью интерфейсных модулей А491-3М (удаленность устройств — до 50 м) или А723/1 (удаленность — до 1000 м). Каждое из устройств контроля способно функционировать и автономно благодаря внутреннему контроллеру, который использует заданный набор команд и управляет потоками данных, поступающих от отдельного фотосчитывающего устройства.

Программное обеспечение (ПО) комплекса базируется на стандартной операционной системе реального времени (ОС РВ), оно состоит из многоканального интерпретатора языка ЗАЛП, подсистемы предварительной подготовки программ проверки, подсистемы генерации, мониторинга задачи, драйвера УКБЛ и УКБА и набора прикладных обслуживающих программ.

Прикладное ПО написано на языке ЗАЛП, открыто для доработки и развития, причем для этого не требуется знания архитектуры СМ-4 или ОС РВ. Остальные компоненты ПО написаны на языке Ассемблера СМ-4. Они поставляются в виде объектных библиотек или в формате образа памяти и подлежат модификации в определенных пределах в рамках подсистемы генерации.

Архив комплекса организован как совокупность файлов ОС РВ на магнитном диске, поэтому для работы с архивом полностью применим весь комплекс обслуживающих программ ОС РВ, позволяющих включать в архив новые программы, исключать имеющиеся, переименовывать, получать распечатки каталогов и т. д.

Тестовые воздействия, которым должен подвергаться объект контроля, и способы обработки его выходных реакций описывают-

ся в виде программы проверки на языке ЗАЛП. С помощью этого языка осуществляется удобная адаптация к составу программно-управляемых аппаратных блоков устройств контроля. Диалоговый режим работы наряду с возможностью трансляции программ (полностью или частично) обеспечивает широкие возможности при постановке отладочных и исследовательских экспериментов в сочетании с достаточно высокой производительностью комплекса.

Возможна многотерминальная работа в режиме разделения времени по различным программам пользователей. Язык позволяет описывать контроль цифровой, аналоговой и комбинированной аппаратуры в простой и удобной форме. Возможности кодирования алгоритмов обработки информации на языке ЗАЛП примерно соответствуют возможностям языков программирования общего назначения (Бейсик, минимальные версии Фортран и пр.).

Кроме директив языка ЗАЛП и директив управления заданиями ПО, КОДИАК выполняет также набор команд построчного редактирования текстовой информации.

Устройство контроля логических блоков предназначено для автоматизированного контроля и отладки блоков, содержащих элементы, входные и выходные сигналы которых имеют дискретный характер (0 или 1).

В УКБЛ реализованы принципы высокоскоростной аппаратной интерпретации входных испытательных последовательностей и компактного представления ответных реакций объекта контроля в виде некоторых интегральных характеристик логических сигналов (количество единиц или импульсов).

Устройство содержит генератор испытательных последовательностей (ГИП) [6], блок счетчиков-анализаторов, блок логического анализатора. ГИП представляет собой специализированный процессор (разрядность соответствует числу входов объекта контроля) с микропрограммным управлением, содержащим 16 регистров общего назначения, а также такие специальные регистры, как генератор псевдослучайных чисел, регистр маски и буферный регистр, связанный с входами объекта контроля. Программа и исходная информация в регистры общего назначения загружаются извне, после чего процессор начинает циклически выполнять программу со скоростью  $5 \cdot 10^6$  оп./с. В результате этого состояние разрядов буферного регистра изменяется по законам, отвечающим заданным испытательным последовательностям, используемым для тестовых воздействий на объект контроля. В зависимости от исходного заполнения одноименных разрядов универсальных регистров на соответствующем выходе ГИП вырабатывается одна из следующих базовых последовательностей:

свободно программируемые конфигурации сигналов на 12 тактах, повторяющихся в каждом цикле программы. Частными случаями такого рода последовательностей являются синхросерии шести фаз обеих полярностей и константные сигналы 1 и 0; псевдослучайные последовательности;



группы псевдослучайных последовательностей, для которых в одном цикле программы не более чем в одной последовательности из группы может произойти изменение логического уровня;

группы псевдослучайных последовательностей, для которых в одном цикле программы запрещено появление нуля (единиц) более чем в одной последовательности из группы.

Для последовательностей с псевдослучайным характером формирования сигналов вероятность логической единицы может программно изменяться. При задании вероятности, равной нулю (единице), в группе последовательностей, относящихся к последнему из перечисленных выше виду, образуется сигнал типа «бегущего нуля» (единицы) в пределах этой группы. Помимо воздействия на объект контроля такими специализированными последовательностями УКБЛ позволяет проводить тестирование путем подачи наборов, каждый из которых был определен заранее и записан в память ЭВМ.

Положение моментов возможных переключений сигналов каждого вида в общем цикле работы ГИП известно, в большинстве случаев разнесено во времени между собой и неизменно от запуска к запуску. Таким образом, программирование собственно процесса контроля в самом общем виде состоит в выборе нужного вида последовательностей для каждого входа объекта, указания длины последовательностей и рабочей частоты ГИП (от 20 кГц до 5 МГц).

Процесс использования комплекса КОДИАК для проверки логических блоков включает в себя следующие этапы:

анализ принципиальной электрической схемы блока и составление первоначального варианта программы проверки;

ввод текста программы проверки во внешнюю память комплекса;

практическую апробацию программы и ее корректировку;

получение перечня характеристик сигналов в контрольных точках блока;

получение и вывод карты проверки, оформленной по ЕСКД;

обработку программы проверки подсистемой предварительной подготовки.

На первом этапе осуществляется теоретический анализ схемы блока с целью обеспечения полноты контроля и эффективной диагностики неисправностей, учета ограничений, накладываемых на процесс контроля реальными характеристиками аппаратуры, определения правильной реакции блока на испытательные последовательности. Как показала практика, первоначальный вариант программы проверки подготавливается специалистом, владеющим основами технической диагностики и детально знакомым с возможностями аппаратуры и ПО КОДИАК, за 0,5—8 ч. Программное управление настройкой ГИП позволяет осуществлять декомпозицию схемы проверяемого блока (т. е. логически разрывать связи, в том числе обратные), что радикально облегчает поиск дефектов в неисправных блоках.

Кодирование программы на языке ЗАЛП обычно не требует много времени, а ее ввод может быть выполнен оператором, зна-

комы лишь с порядком работы на алфавитно-цифровых видеотерминалах, которыми снабжены рабочие места комплекса КОДИАК.

Следующий этап работы — апробация программы — выполняется ее разработчиком и занимает 4—30 ч работы на УКБЛ, оснащенном приспособлениями для подключения и видеотерминалом, который используется для оперативной коррекции программы и проведения отладочных экспериментов.

Дальнейшие операции, необходимые для получения действующей программы — снятие характеристик с эталонного объекта контроля, подключенного к УКБЛ, получение документации на карту проверки и работа с подсистемой предварительной подготовки — автоматизированы, поэтому действия оператора носят чисто формальный характер. Для завершения этих операций требуется 1—8 ч работы на комплексе. Алгоритмы исследования характеристик объекта контроля закодированы на языке ЗАЛП.

Диагностирование логического блока по готовой программе включает в себя: прогон программы с целью получения выходных сигналов с характеристиками, не совпадающими с эталонными; поиск места неисправности с помощью таблиц характеристик сигналов во внутренних точках блока (при необходимости в программу может быть введена процедура типа «ведомого щупа»); локализацию дефектов средствами блока логического анализатора. Этот блок УКБЛ обеспечивает:

потактную работу с визуальным контролем на мнемосхеме логических состояний выводов микросхем;

останов ГИП по заданному набору на выбранных выводах «клипсы» с широким выбором вида синхронизации (асинхронная, внутренняя, внешняя);

проверку правильности функционирования микросхем объекта контроля путем аппаратного сравнения ее работы с работой эталонной микросхемы, программно выбираемой по условному коду. Набор эталонных микросхем, устанавливаемых в УКБЛ, может определяться пользователем.

Устройство контроля аналоговых блоков включает в себя блок задания и сравнения логических сигналов, коммутаторы низко- и высокочастотных аналоговых сигналов, измерительные и задающие приборы с дистанционным управлением, которые через магистральный приборный интерфейс соединены с контроллером устройства и через него — с ЭВМ.

Технология работы с аналоговыми блоками включает в себя те же этапы: составление программы, ее ввод и отладку, выпуск документации, контроль и отладку блоков по готовой программе. Особенности составления программы для УКБА состоят в том, что входная и выходная реакция аналоговых блоков обычно оговаривается в технической документации, регламентирующей порядок контроля данного блока (технические условия, программа и методика испытаний).

Комплекс КОДИАК находится в промышленной эксплуатации. С его помощью автоматизированы контрольно-отладочные опера-

ции серийного производства мультиплексора СМ-8514, контроллера устройства внешней памяти на сменных магнитных дисках СМ-5407, процессоров СМ-2104, СМ-2410, расширителя интерфейса СМ-4101, адаптера дистанционной связи СМ-8502, модулей УВК СМ-1800. Номенклатура проверяемых блоков составляет свыше ста наименований. Общее количество микросхем в блоках достигает 60 шт. В их составе — микропроцессорные наборы, оперативная и постоянная память сравнительно большой емкости, шинные формирователи и приемники, обслуживающие двунаправленные шины, многоразрядные счетчики и разнообразные формирователи импульсных сигналов, асинхронные автоматы. Время подготовки карт проверок на логические блоки указанных выше устройств составило в среднем по 24 ч на блок. Время отладки блока зависит от количества имеющихся в нем неисправностей, например при пяти неисправностях оно составляет в среднем 15—20 мин.

Полнота контроля, оцениваемая по случаям отбраковки блоков при комплексной отладке устройств составляет около 98%. Эти показатели были получены при анализе результатов контроля примерно 5 тыс. блоков СМ ЭВМ.

#### ЛИТЕРАТУРА

1. Филинов Е. Н., Гробман Д. М., Сергеев Б. Г. **Высокопроизводительный способ контроля блоков ЭВМ.** — В кн.: Вычислительная техника социалистических стран. — М.: Финансы и статистика, 1981, вып. 10, с. 47—53.
2. **Программные и аппаратные средства автоматизированной системы диагностирования цифровых блоков/В. П. Немченко, В. А. Рустин, В. И. Хаханов и др.** — В кн.: V Всесоюзное совещание по технической диагностике. — М.: Ин-т проблем управления, 1982, с. 143—145.
3. Турлов П. А., Елзов В. А. **Устройство ТЕСТ-1 для тестового диагностирования логических плат.** — В кн.: V Всесоюзное совещание по технической диагностике. — М.: Ин-т проблем управления, 1982, с. 173—175.
4. **Контрольно-диагностический автоматизированный комплекс/В. П. Сидоренко, О. Д. Руккас, М. С. Берштейн и др.** — Приборы и системы управления, 1981, № 5, с. 4—5.
5. **Проблемно-ориентированный язык программирования ТЕСТ/Под ред. И. Ф. Клисторина.** — Кишинев: Штиинца, 1978, с. 128.
6. **Организация процессов контроля и диагностики цифровых блоков/В. П. Сидоренко, О. Д. Руккас, Е. Н. Чичирин и др.** — Управляющие системы и машины, 1982, № 4, с. 18—22.

УДК 65.011.56:681.325.5-181.5

### ОСОБЕННОСТИ ОРГАНИЗАЦИИ АСУ ПРОМЫШЛЕННОГО ПРЕДПРИЯТИЯ С ИСПОЛЬЗОВАНИЕМ МИКРОПРОЦЕССОРНЫХ СИСТЕМ

А. И. Андерсон, инженер (СССР)  
А. П. Глазов, инженер (СССР)  
А. А. Кривченков, инженер (СССР)

Широкое применение современных технических средств сбора, обработки, отображения и передачи информации позволяет на принципиально новой основе решать вопросы построения АСУ промышленного предприятия (ПП). Характер-

ной особенностью современной АСУ ПП является наличие развитой информационной модели производства, а также высокоорганизованной распределенной вычислительной системы (РВС), обеспечивающей сбор информации через терминальные устройства непосредственно с рабочих мест, передачу данных на центральную ЭВМ с целью интегрированной обработки их и дальнейшего перераспределения в соответствии с требованиями системы.

В АСУ ПП реализуется тенденция, наметившаяся в последние годы, по слиянию функций организационно-экономических АСУ с автоматизированными системами управления технологическими процессами в рамках комплексной автоматизации производства. При создании подобной АСУ ПП необходимо решить ряд сложных технических и организационных проблем, в частности проблему обеспечения надежности функционирования и организации технического обслуживания РВС. В процессе функционирования РВС в интерактивный режим вовлекаются широкие круги управленческого персонала предприятия. Это требует разработки простых, но достаточно гибких средств диалога пользователя с системой, допускающих формулировку или адаптацию заданий в соответствии с конкретными запросами.

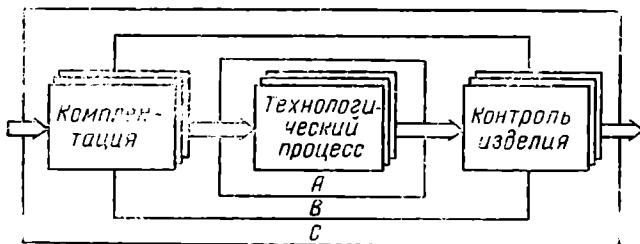
Развитие элементной базы и создание на основе больших интегральных схем микропроцессорной (МП) аппаратуры, имеющей очевидные преимущества в РВС [1, 2], позволяют проектировать АСУ ПП с учетом этих новых технических возможностей. Однако современное состояние МП средств таково, что построение многоуровневой АСУ с большими объемами обрабатываемой и передаваемой информации, какой является АСУ ПП, с применением только этих средств практически невозможно. В этой связи актуален вопрос о месте микропроцессорных систем в общем комплексе технических средств АСУ ПП.

В данной статье рассмотрены два вопроса, связанные с применением микропроцессорных систем: выделение контуров управления при организации современного производства, в которых могут быть эффективно использованы МП средства, и оценка эффективности реализации алгоритмов АСУ ПП на МП средствах.

Производственный процесс применительно к одному изделию может быть разбит на три этапа — комплектация, технологический процесс и контроль. Эти процессы взаимосвязаны определенными материальными потоками и охвачены разноуровневыми контурами управления (см. рисунок).

Можно выделить три типа контуров управления — управление технологическим процессом (А), оперативное управление и контроль (В), управление кооперацией (С). Изделие, характерное, например, для радиоэлектронной промышленности, является совокупностью тысяч элементов, участвующих в производственном процессе, что приводит к соответственному возрастанию количества контуров управления.

Каждый тип контуров управления характеризуется определенным набором задач, решаемых в условиях автоматизации управле-



Контуры управления в производственном процессе

ния, относительно которых можно определить значение таких функциональных параметров, как цикл обработки информации и ее объем. В частности, для производства радиоэлектронной аппаратуры в условиях современной технологии и методов управления характерны следующие циклы обработки информации в указанных контурах управления:  $T(A) \sim 10^{-3}$  с,  $T(B) \sim 10$  с,  $T(C) \sim 10^4$  с. Соответственно объемы условно-постоянной информации имеют следующие значения:  $A - 10^2$  слов,  $B - 10^4$  слов,  $C - 10^7$  слов. Эти данные получены в условиях массового производства телефонной аппаратуры и бытовой радиоаппаратуры.

В контуре  $A$  применение МП оборудования становится традиционным. Малые габариты, высокая надежность, гибкость благодаря программируемости, а также приемлемые функциональные требования ( $10^{-3}$  с,  $10^2$  слов) позволяют создавать эффективные микропроцессорные системы, которые получили отражение в большом количестве работ [3, 4]. Подобные управляющие системы содержат 8—16 внутренних контуров управления на одну микроЭВМ (при этом следует ожидать, что объем программного обеспечения для 8-битового МП составит 1,5—8 Кбайт).

Контур  $B$  в гораздо меньшей степени охвачен применением МП оборудования. Здесь преобладает автоматизация контрольно-измерительных операций [1]. В то же время задачи автоматизации управления пока решаются в основном на уровне статистической обработки результатов автоматизированного или автоматического контроля изделий.

Эффективность работы микропроцессорных систем в этом контуре заметно повышается при решении вопросов обмена данными с АСУ верхнего уровня, так как функциональные требования (10 с,  $10^4$  слов) не позволяют обеспечить эффективную работу контура в автономном, не зависящем от остальных контуров АСУ, режиме. Во всяком случае ряд успешных реализаций микропроцессорных систем в этом контуре управления [1] позволяет сделать вывод, что с развитием МП средств контур будет в основном строиться на их базе. При этом могут быть эффективно решены следующие задачи АСУ:

автоматизированная и автоматическая комплектация изделий согласно сменному заданию;

автоматическая транспортировка изделий в рамках технологического участка;

автоматизированный или автоматический контроль изделий по широкой программе после технологических операций;

сбор данных о количестве и качестве труда рабочих, предварительная обработка этих данных и их оценка;

интеграция данных до уровня отчетности технологических участков.

Контур *C* — управление входными и выходными для технологического участка материальными потоками — является сферой, контролируемой большинством современных организационно-экономических АСУ. Успешное применение микропроцессорных систем в этом контуре с учетом повышенных функциональных требований ( $10^4$  с,  $10^7$  слов) представляется последовательным внедрением контуров типа *B*, объединенных развитой системой обмена информацией и связью с центральной ЭВМ.

Таким образом, при последовательном совершенствовании МП средств для их внедрения в АСУ ПП характерно движение снизу вверх. Анализ функционирования требований и оценка таких параметров, как цикл обработки информации и объем условно-постоянной информации, позволяют определить применимость имеющихся в распоряжении проектировщика МП средств в выбранном контуре управления. Наиболее эффективны с точки зрения использования в них МП контур АСУ технологическим процессом и контур оперативного управления и контроля. Для расширения сферы применения микропроцессоров в АСУ необходимо дальнейшее совершенствование аппаратуры передачи данных и выработка в этой области стандартных решений.

Обратимся теперь к вопросам проектирования микропроцессорных систем для их использования в АСУ. Разработка систем данного класса имеет ряд отличительных особенностей:

сравнительно низкую стоимость разработки аппаратных средств; наличие ограничений по производительности процессоров и объемам запоминающих устройств, которые должны быть учтены разработчиками при проектировании;

использование при разработке программного обеспечения (ПО) языков низкого уровня, что определяет в некоторых случаях значительные затраты в этой области.

Следовательно, при создании микропроцессорных систем информационно-управляющего характера наблюдается значительное смещение затрат в сторону разработки ПО.

Один из традиционно сложных для АСУ вопросов — предпроектная оценка экономической эффективности от внедрения той или иной системы информационно-управляющего характера.

С точки зрения определения эффективности микропроцессорные системы АСУ можно разделить на три категории:

системы, экономическая эффективность которых определяется на основе предпроектных оценок;

системы, эффективность которых может быть определена после

их внедрения на основе измерения соответствующих параметров до и после внедрения;

системы, эффективность которых не может быть непосредственно выражена в стоимостном выражении.

В большинстве случаев микропроцессорные системы АСУ, имеющие информационно-управленческий характер, относятся к третьей категории. В этой ситуации особо остро стоит вопрос о критериях проектирования и внедрения соответствующих систем. Однако в ряде случаев пользоваться предложенными в литературе критериями затруднительно, поэтому предлагается некоторая характеристика — «удельная стоимость алгоритма» работы системы обработки информации. Детальное обсуждение этого критерия выходит за рамки статьи, а его суть сводится к следующим положениям.

В любой системе обработки информации в АСУ (ручной, автоматизированной или автоматической) можно выделить ряд формализованных переменных  $P_i$  (например, «код изделия», «трудоемкость операции», «фамилия работника» и т. п.), которые участвуют в общем алгоритме работы этой системы. Если изобразить алгоритм работы системы как  $A(P_1, P_2, \dots, P_n)$  и ввести «объем» алгоритма  $V^* = (2+n) \log_2(2+n)$ , где  $n$  — количество существенно различных переменных, то  $V^*$  будет соответствовать минимально необходимому количеству бит для формальной записи алгоритма.

Известно, что при детализации описания алгоритмов на процедурных языках (Ассемблер, Фортран и т. п.) объем их записи  $V$  будет некоторой функцией  $F(V^*, \lambda)$  от минимального (потенциального) объема и уровня языка  $\lambda$ . Уровни  $\lambda$  могут быть определены путем статистических исследований описаний различных алгоритмов для различных языков. Параметр  $V^*$  для алгоритма системы будет в определенном смысле инвариант, не зависящий от его записи. Так как в рассматриваемых микропроцессорных системах АСУ алгоритм их работы в конечном итоге изображается на языке программирования (программируемые системы), то, зная параметры  $V^*$  и  $\lambda$ , можно оценить размеры проектируемого ПО, а следовательно, и затраты на его разработку.

В [6] было предложено соотношение для оценки времени программирования алгоритма  $T = \frac{1}{S} \cdot \frac{(V^*)^3}{\lambda^2}$ , где  $S$  — постоянная. Авторами данной работы установлено, что оно с достаточной степенью точности справедливо при программировании на Ассемблере для МП К580ИК80.

Есть основания полагать, что надежность ПО после определенного времени отладки также может быть оценена по параметрам  $V^*$  и  $\lambda$ .

Таким образом, удельная стоимость алгоритма работы  $q$  в качестве критерия проектирования систем рассматриваемого класса определяется соотношением

$$q = \frac{C}{V^*},$$

где  $C$  — затраты на разработку аппаратуры и ПО.

Если до внедрения разрабатываемой системы уже имелась та или иная система обработки информации, то целесообразно считать, что

$$q = \frac{C}{V^* - V_0^*},$$

где  $V_0^*$  — потенциальный объем исходного алгоритма.

В ряде случаев  $q$  можно оценить по формуле

$$q = \frac{C_a + C_{по}}{V^*} = M \cdot \frac{C_m}{V^*} + \frac{C_{п} K_{п}}{V^*} \cdot \frac{(V^*)^3}{S \lambda^2},$$

где  $C_a$  — стоимость аппаратуры;

$C_{по}$  — стоимость ПО;

$C_m$  — стоимость микроЭВМ для данной организации в данное время;

$M$  — количество микроЭВМ в системе;

$C_{п}$  — стоимость труда программистов;

$K_{п}$  — коэффициент использования рабочего времени программистами.

Очевидно, что существует оптимум этого выражения. В данном случае применялась простейшая модель стоимости МП оборудования, которая может быть уточнена на основе подхода, развитого, например, в работе [8].

| Область применения                               | Количество уровней иерархии | Количество операторов в системе | Количество микроЭВМ | Работа в РМВ <sup>1</sup> | Наличие внешнего банка данных | Контур управления АСУ | 1/q, бат. руб. |
|--|-----------------------------|---------------------------------|---------------------|---------------------------|-------------------------------|-----------------------|----------------|
| Диагностика цифровых печатных плат               | 1                           | 1                               | 1                   | —                         | +                             | <b>B</b>              | 0,02           |
| Диагностика блоков телефонных станций            | 1                           | 1                               | 1                   | —                         | +                             | <b>B</b>              | 0,02           |
| Измерение и контроль параметров радиоприемников  | 2                           | —                               | 6                   | +                         | +                             | <b>A—B</b>            | 0,01           |
| Управление автоматизированным складом            | 1                           | 1                               | 1                   | +                         | —                             | <b>A</b>              | 0,01           |
| Централизованное управление полуавтоматами       | 2                           | 31                              | 1                   | +                         | +                             | <b>A—B</b>            | 0,02           |
| Управление и организация спортивных соревнований | 4                           | 51                              | 7                   | +                         | +                             | <b>C</b>              | 0,01           |

<sup>1</sup> РМВ — реальный масштаб времени.



В заключение приведем ряд значений, полученных на основе анализа нескольких микропроцессорных систем (тип микропроцессора К580ИК80, контура типа А и В, язык программирования — Ассемблер, 176 программ):  $\lambda \sim 1,2$  бит;  $S \sim 10 \div 20$  бит/с;  $(V_m^*)_{\text{ср}} \sim 21$  бит;  $(K)_{\text{ср}} \sim 170$  операторов, где  $(V_m^*)_{\text{ср}}$  — средний потенциальный объем модуля, написанного на языке Ассемблер;  $(K)_{\text{ср}}$  — среднее количество операторов Ассемблера в модуле. Значения критерия  $q$  в ряде систем представлены в таблице (см. с. 159).

Как видно из таблицы, разработанные в рамках АСУ микропроцессорные системы разнородны по функциональной направленности, а их параметр «удельная стоимость алгоритма» достаточно близок. Этот параметр может быть оценен на этапе начального проектирования на основе статистических данных, приведенных в настоящей статье. Его значение 50—100 руб./бит характерно для микропроцессорных систем при современном уровне используемых авторами статьи МП средств. Значительное превышение указанных значений при реализации МП АСУ означает неэффективность ее в указанном выше смысле, что должно быть оправдано особыми функциональными требованиями. Если таковых нет, то рассматриваемый проект следует переработать.

#### ЛИТЕРАТУРА

1. Микропроцессорные системы контроля и управления/А. И. Андерсон, Я. Я. Вентиньш, П. О. Видениекс и др. — Рига: Зинатне, 1981.
2. Топфер Х., Крузель В. Тенденции функционального и структурного развития автоматизированных систем. — Зарубежная радиоэлектроника, 1982, № 6.
3. Сobotка З., Стары Я. Микропроцессорные системы. — М.: Энергоиздат, 1981.
4. Статьи в разделе «Изделия электронной техники и их применение». — Электронная промышленность, вып. 7 (91), 1980, с. 24—45.
5. Куприянов М. С. Опыт разработки микропроцессорных систем управления. — М.: Радио и связь, 1981, с. 4—7.
6. Липаев В. В. Проектирование математического обеспечения АСУ. — М.: Сов. радио, 1977. 400 с.
7. Холстед М. Х. Начала науки о программах. — М.: Финансы и статистика, 1981. 128 с.
8. О критерии стоимость/эффективность для оценки вычислительных систем в США. — Экспресс-информация, сер. Вычислительная техника. М.: ВИНТИ, 1982, № 39, с. 15—16.

# V

## Эксплуатация и обслуживание средств вычислительной техники

---

УДК 681.3.06:33

### ТЕХНИКО-ЭКОНОМИЧЕСКАЯ ОЦЕНКА МУЛЬТИПРОГРАММИРОВАНИЯ

С. Л. Козлов, инженер (СССР)  
Э. А. Петросян, канд. техн. наук (СССР)  
А. Е. Фатеев, канд. экон. наук (СССР)

Широкое распространение дорогостоящих средств вычислительной техники (ВТ), представленных у массового потребителя в основном вычислительными машинами общего назначения — моделями Единой системы ЭВМ (ЕС ЭВМ), обуславливает необходимость постоянного анализа эффективности их использования, что в свою очередь требует систематического совершенствования методов технико-экономической оценки средств ВТ.

В последнее время системное программное обеспечение большинства ЭВМ общего назначения поддерживает мультипрограммный режим работы, который позволяет интенсифицировать работу технических средств (ТС) и программного обеспечения (ПО) за счет совершенствования технологии решения задач.

Мультипрограммирование в конкретной вычислительной системе ЕС ЭВМ как способ использования ТС осуществляется в результате того, что в каждый момент времени существуют несколько готовых к выполнению задач, а выполняется одна задача (это верно для однопроцессорной вычислительной системы). При этом существует механизм переключения процессора с одной задачи на другую.

Хотя в каждый момент времени обслуживается одна задача, рассматривая определенный интервал времени, можно говорить об одновременном или, точнее, квазисовременном выполнении ряда задач. Процесс переключения с задачи на задачу отвлекает часть ресурсов ЭВМ, поэтому возникает оптимизационная задача выбора количества задач, находящихся в состоянии готовности к выполнению.

Следует особо подчеркнуть, что при мультипрограммном режиме функционирования ЭВМ абсолютные значения «накладных расходов» (общих расходов на обеспечение функционирования системы, которые прямо невозможно отнести на решение конкретных прикладных задач) увеличиваются по сравнению с однопрограммным

режимом. Относительные же показатели, характеризующие эффективность операций, т. е. отношение общих затрат к получаемым результатам, в этом случае существенно улучшаются.

В процессе эксплуатации вычислительной техники существует специфика определения «накладных расходов», связанная с организацией функционирования современных ЭВМ. Например, для обеспечения виртуальной памяти требуются ресурсы системы, которые отвлекаются от пользовательских задач и не могут быть прямо отнесены на данную конкретную задачу, но в свою очередь виртуальность памяти обеспечивает пользователей ресурсами, которые в ином случае были бы им недоступны. Таким образом, можно констатировать, что режимы работы ЭВМ обуславливают методы и формы распределения затрат на пользовательские задачи. В свою очередь экономически обоснованные формы расчетов с потребителями диктуют поиск более совершенных методов организации вычислительного процесса.

В настоящее время для ЕС ЭВМ можно перечислить следующие широко используемые режимы:

однопрограммный;

выполнения одной пользовательской задачи в режимах ОС ЕС — MFT, MVT и SVS;

квазиодновременного выполнения нескольких пользовательских задач в режимах ОС ЕС — MFT, MVT и SVS.

С точки зрения удобств пользователя наиболее предпочтительными являются однопрограммный режим и режим выполнения одной пользовательской задачи, но в этих случаях непомерно возрастает стоимость ее решения в современных дорогостоящих вычислительных системах. Введение мультипрограммирования для любого пользователя во всех аспектах, кроме стоимостного, означает возникновение некоторого неудобства в работе по сравнению со случаем, когда он использует ресурсы ЭВМ монопольно. Это выражается в следующем:

в результате одновременного нахождения в системе в активном состоянии нескольких заданий стартстопное время их решения существенно больше значения аналогичного показателя при монопольном владении ресурсами ЭВМ; одним из следствий этого является уменьшение вероятности запланированного завершения выполнения задания;

увеличивается объем требующихся для выполнения заданий ресурсов.

Положительные стороны мультипрограммирования проявляются по отношению к совокупности пользователей или, точнее, ко всем пользователям, взятым вместе. Они заключаются в следующем:

а) уменьшается среднее время для пользователя от момента передачи подготовленного для выполнения задания до момента получения результатов;

б) за счет интенсификации использования ресурсов вычислительной системы (ВС) увеличивается число ее пользователей;

в) несмотря на то что на выполнение задания приходится абсолютно больший объем потребляемых ресурсов, стоимость решения каждой конкретной задачи из совокупности решенных существенно уменьшается.

В настоящее время формируется отношение к экономически обоснованному использованию мультипрограммирования. Один из его аспектов связан с определением совокупности показателей, обуславливающих эффективность применения данного режима в конкретной ситуации. В этой связи становится важным обсуждение опыта эксплуатации конкретных систем и его преломление с точки зрения тех или иных предлагаемых показателей.

В различных коллективах, эксплуатирующих ВС, сложилось несколько отличных друг от друга подходов к определению показателем мультипрограммирования: коэффициента мультипрограммирования, уровня мультипрограммирования и т. п. Излагая сущность этих подходов, можно утверждать, что все они при определении уровня мультипрограммирования сводятся к значению, во-первых, отношения

$$h_1 = \frac{T_{пз}}{T_{рс}},$$

где  $T_{пз}$  — суммарное время работы центрального процессора, затраченное на решение прикладных задач, а  $T_{рс}$  — время работы вычислительной системы, а во-вторых, отношения

$$h_2 = \frac{T_{спз}}{T_{рс}},$$

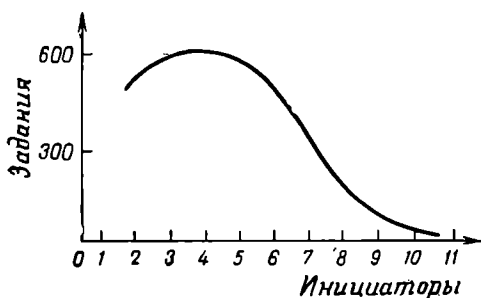
где под  $T_{спз}$  понимается суммарное стартстопное время (т. е. время от момента инициирования задания до момента его завершения) решения прикладных задач.

Часто не делается различий между понятиями «уровень» и «коэффициент» мультипрограммирования. При этом используемые соотношения ближе характеризуют понятие «уровень мультипрограммирования». Отвлекаясь от среднего числа находящихся в системе активных задач, можно утверждать, что показатель  $h_1$  отражает степень интенсивности использования ресурса центрального процессора и поэтому по сути адекватен понятию «уровень мультипрограммирования» для наиболее многочисленной группы задач обработки данных. Более того, этот показатель используется для характеристики уровня интенсивности использования основного в функциональном плане ресурса ЭВМ. Уровень мультипрограммирования в этом случае считается тем выше, чем отношение  $\frac{T_{пз}}{T_{рс}}$  ближе к единице.

Использование рассмотренного показателя имеет ряд недостатков. Если в ЭВМ первого и второго поколений процессор являлся основным ресурсом, то для машин третьего поколения положение существенно изменилось. Исследуя приведенное отношение более подробно,  $T_{рс}$  можно рассматривать при работе в среде ОС как точную сумму  $T_{пз}$ , времени ожидания ( $T_0$ ) и времени, требуемого

на обеспечение функционирования самой системы ( $T_{\text{фс}}$ ) — «накладных расходов». Поэтому отношение  $h_1$  преобразовывается к виду  $1 - \frac{T_{\text{фс}} + T_0}{T_{\text{рс}}}$ .

Таким образом, уровень мультипрограммирования является в том числе функцией доли «накладных расходов» функционирования системы, т. е. при рассматриваемом подходе к определению уровня мультипрограммирования многое сводится к доле «накладных расходов», а не к вопросу эффективности решения задач (вопросу,



Зависимость количества решенных за смену заданий от числа инициаторов в вычислительной системе ЕС-1055

ради решения которого и вводилось мультипрограммирование). Так как сама операционная система, являясь одной из самых сложных существующих программных систем, недостаточно исследована с точки зрения оптимальности функционирования с учетом количественных оценок, то, по мнению авторов, изложенный подход к определению уровня мультипрограммирования является не всегда корректным.

Рассмотрим также отношение  $h_2$ . В связи с тем, что значение показателя уровня мультипрограммирования прямо пропорционально стартстопному времени решения задач, важно определить, насколько устойчивым показателем для задач является стартстопное время или отношение суммарного стартстопного времени решения задач ко времени работы вычислительной системы.

Опыт авторов, накопленный при эксплуатации ВС в режиме мультипрограммирования на задачах отладки, типичных для многих ВЦ, позволяет сформулировать следующие положения:

1) стартстопное время есть функция, зависящая от времени, затрачиваемого на различного рода ожидания при выполнении задания. При увеличении времени ожидания стартстопное время увеличивается нелинейным образом;

2) использование многих инициаторов в системе также увеличивает стартстопное время, во-первых, за счет увеличения ресурсов для обеспечения этого режима и, во-вторых, за счет увеличения времени ожидания по крайней мере при работе с системными наборами с диспозицией  $DISP = SHR$ , т. е. когда набор данных может «разделяться» многими заданиями;

3) опыт использования монитора СМП и средства ОРГВЫЦ для систематизации собранной СМП информации показывает, что в ситуации, когда одновременно решается пакет заданий и функционирует ППП КАМА, время решения задач пакета увеличивается в среднем на 15% по сравнению со случаем, когда КАМА не функционирует.

На рисунке приведены данные о зависимости производительности ВС от числа работающих инициаторов, которые иллюстрируют приведенные выше положения. Эти данные получены для ЕС-1055, в конфигурацию которой входит шесть дисков емкостью по 200 Мбайт: она используется для подготовки с помощью ДУВЗ и отладки задач, написанных на языках ПЛ/1, Кобол и Ассемблер, под управлением ОС ЕС 6.1 в режиме SVS.

Можно проследить явную зависимость увеличения стартстопного времени вместе с увеличением «накладных расходов». В этой связи естественно рассматривать время решения задачи  $i$  как сумму времени выполнения команд задания ( $T_{\text{вк}}i$ ), которое почти всегда постоянно для задания, и времени на обеспечение решения задачи в мультипрограммном режиме на конкретной ВС при данном классе выполняющихся задач ( $T_{\text{ом}}i$ ). Тогда суммарное стартстопное время решения  $n$  задач можно представить следующим образом:

$$\sum_{i=1}^n (T_{\text{вк}}i + T_{\text{ом}}i).$$

Общее время работы вычислительной системы ( $T_{\text{рс}}$ ), как правило, является константой, равной 8, 16 или 24 ч в сутки. Так как сутки для большинства ВЦ являются минимальным временем отчетности и, следовательно, периодом условно-законченного цикла решения совокупности задач, то рассматриваемая формула определения уровня мультипрограммирования представляется в следующем виде:

$$\frac{\sum_{i=1}^n T_{\text{вк}}i + \sum_{i=1}^n T_{\text{ом}}i}{T_{\text{рс}}},$$

где  $\sum_{i=1}^n T_{\text{вк}}i$  и  $T_{\text{рс}}$  являются константами. В этой связи рассматриваемый показатель — уровень мультипрограммирования — является функцией только затрат, требуемых для решения задач в режиме мультипрограммирования. В таком виде показатель не отражает степени увеличения интенсификации работы ЭВМ для решения прикладных задач.

Таким образом, при мультипрограммном режиме стартстопное время не является показателем, сравнимым с показателем времени работы процессора.

Важным аспектом эффективности того или иного режима эксплуатации ЭВМ является то, что объемы вычислительных работ и работ по вводу-выводу сильно изменяются от задачи к задаче (в особенности с учетом классов решаемых задач), поэтому если на вычислительной системе решается не одна, а несколько задач, то в однопрограммном режиме эти задачи заведомо не будут решаться оптимальным в отношении затрат ресурсов ЭВМ образом. Наряду с этим не всякий режим мультипрограммирования может обеспечивать оптимальные показатели решения задач или их совокупности.

При решении любой задачи с помощью ЭВМ необходимо исходить из того, что каждая задача определяется рядом параметров: минимальной конфигурацией ЭВМ, на которой она может быть решена, и оптимальными конфигурациями, на которых при решении некоторой совокупности задач могут быть достигнуты оптимальные значения показателей (например, наименьшее время решения задачи, минимум «накладных расходов» и т. п.).

Введем обозначения. Для задачи  $i$  обозначим конфигурации ТС и ПС как  $\overline{KTC}_j^i$  и  $\overline{KPC}_k^i$ , оптимальные по классам  $j$  и  $k$  соответственно. На практике возникает ситуация, когда задача решается с определенным периодом (час, день, неделя и т. п.). В этом случае периодичность в ее решении приводит к тому, что либо простаивает дорогостоящее оборудование, либо на данной конфигурации решается другая задача, для которой эта конфигурация с очень большой долей вероятности не будет оптимальной. В этой связи встает вопрос об оптимизации КТС и КПС не для конкретной задачи, а для их совокупности. Сущность мультипрограммирования как технологии использования вычислительных средств состоит в том, что вычислительная система состоит из КТС и КПС избыточных, не оптимальных с точки зрения решения каждой конкретной задачи, но позволяющих оптимально решать совокупность задач. Используя введенные выше обозначения, это можно выразить следующим образом:

$$C(\overline{KTC}_j^i) < C(\overline{KTC}_j^i) < \sum_{i=1}^n C(\overline{KTC}_j^i)$$

$$C(\overline{KPC}_k^i) < C(\overline{KPC}_k^i) < \sum_{i=1}^n C(\overline{KPC}_k^i)$$

$$\forall i \in \{l/l=1, \dots, n\} \text{ и } \forall k, j \in \{p/p=1, \dots, n\},$$

где функция  $C$  ставит в соответствие каждому вектору, определяющему конфигурацию, стоимость этой конфигурации, выраженной, например, в денежных единицах;  $\overline{KTC}_j^i$ ,  $\overline{KPC}_k^i$  — это реально используемые для решения совокупности задач конфигурации технических и программных средств, оптимальные по классу  $j$  и  $k$ .

Для определения уровня мультипрограммирования будем использовать порядковую шкалу, используемую для классификации конкретных ситуаций по классам; она позволяет также упорядочивать сами классы. В этом случае если уровень мультипрограммирования нумеруется, то с этими числами нельзя производить арифметических операций. Если обозначить классы словами, то их порядок соответствует смыслу слов. Например, однопрограммный режим, низкий уровень мультипрограммирования, средний уровень мультипрограммирования, высокий уровень мультипрограммирования. Отметим, что значение «низкий уровень мультипрограммирования» не является синонимом плохой работы ЭВМ для конкретного приложения.

Для определения коэффициента мультипрограммирования будем использовать шкалу отношений, которая позволяет не только клас-

сифицировать и упорядочивать конкретные случаи, но и качественно оценивать различия между классами. (За абсолютное начало отсчета будем принимать однопрограммный режим работы.) В этой шкале можно определить, во сколько раз одно измерение превосходит другое. Структура шкалы отношений не изменяется при преобразовании  $X' = CX$ ,  $C > 0$ . Для этой шкалы имеют смысл все арифметические операции.

Различный качественный подход к «уровню» и «коэффициенту» мультипрограммирования, обусловленный во многом смыслом этих понятий, позволяет более корректно отражать требования эффективного использования рассматриваемого режима.

Показатель «уровень мультипрограммирования», связанный с соотношением объемов операций ввода-вывода и вычислительных операций в совокупности решаемых на некоторой данной ВС задач, можно использовать для координации требований конкретной задачи к ВС и требований эффективности использования этой ВС для решения всей совокупности задач. Таким образом, оптимальный уровень мультипрограммирования определяется для каждой задачи; он зависит от соотношения объемов команд и запросов ввода-вывода и обуславливает требования задачи к той среде и совокупности задач, в которой она будет выполняться.

Коэффициент мультипрограммирования для задачи  $i$  определим в виде вектора:  $\bar{A}^i = (a_1^i, a_2^i, a_3^i)$ , где  $a_r^i$  представляет собой коэффициент уменьшения времени или увеличения объема требуемых ресурсов ЭВМ для решения задачи  $i$ :  $r=1$  — при переходе от РСР к МFT;  $r=2$  — при переходе МFT к MVT,  $r=3$  — при переходе от MVT к SVS.

При обратном переходе можно использовать коэффициент  $\frac{1}{a_r^i}$ .

Положительной стороной использования такого коэффициента мультипрограммирования является возможность рассчитывать оптимальные конфигурации ТС и ПО для создания режима эффективного решения задач приложений. Таким образом, все существующие режимы — РСР, МFT, MVT и SVS — имеют сферы, где их применение дает наиболее оптимальные результаты. Поэтому тезис о том, что МFT лучше РСР, MVT лучше МFT, а SVS лучше MVT, неправилен. При этом исходя из имеющихся конфигураций ТС, ПО и совокупности подлежащих решению задач имеет смысл рассматривать наиболее оптимальный режим использования ЭВМ.

К положительным свойствам предложенного выше определения коэффициента мультипрограммирования относится то, что при переходе от одного режима к другому можно получить, умножив все ресурсы на соответствующий коэффициент  $a_r^i$ , требуемые объемы ресурсов при новом режиме исходя из их значения при старом режиме. Эта же черта — простота является в то же время отрицательной, так как новый режим предоставляет и новые возможности, которые в этом случае не учитываются.

Более предпочтительно, хотя и более сложно, рассчитывать требуемые ресурсы, подразумевая под  $a_r^i$  не число, а вектор такой же



размерности, что и  $\overline{\text{KTC}}_j^i$ . В этом случае процедура перерасчета ресурсов при переходе на новый режим более корректна и выглядит следующим образом:

$$A^i \cdot \overline{\text{KTC}}_j^i = (\bar{a}_1^i, \bar{a}_2^i, \bar{a}_3^i). \quad \overline{\text{KTC}}_j^i = \overline{\text{KTC}}_j'^i,$$

где  $\bar{a}_r^i = (a_{r_1}^i, a_{r_2}^i, \dots, a_{r_n}^i)$ ,  $r = 1, 2, 3$ , а  $\overline{\text{KTC}}_j^i$  — вектор, определяющий конфигурацию ТС для задачи в новом режиме.

При таком подходе возникает вопрос определения коэффициентов матрицы  $A^i$ . В этой связи необходимо отметить, что основной объем задач ВЦ составляют задачи функционирования и задачи отладки. Рассматривая задачи функционирования, т. е. задачи, решаемые на ВЦ постоянно с неким периодом, для определения коэффициентов матрицы следует рекомендовать метод натурального эксперимента и измерения. При решении задач отладки эти коэффициенты можно определить более просто: так как в этом случае значения коэффициентов есть функции, зависящие от характеристик используемого компилятора и объема отлаживаемой программы, то можно составить таблицы с большой долей вероятности, верно отражающие процесс для всех работ такого рода.

Суммируя вышензложенное, можно констатировать, что некорректное использование возможностей режима мультипрограммирования приводит к тому, что пропускная способность ВС часто бывает занижена по сравнению с возможной. Имея в виду, что часть из параметров, характеризующих работу ВС, носит отчетный характер, можно утверждать, что к ним не должны относиться параметры мультипрограммирования. Это связано с тем, что, во-первых, они наиболее явно могут учитываться в показателе «стоимость решения задачи», а во-вторых, некоторыми искусственными способами может быть достигнуто любое значение этих параметров. Наряду с этим использование показателей мультипрограммирования эффективно для выбора оптимальных параметров хода вычислительного процесса.

УДК 681.323:33

### СИСТЕМА ПОКАЗАТЕЛЕЙ ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ ДЕЯТЕЛЬНОСТИ ВЫЧИСЛИТЕЛЬНЫХ ЦЕНТРОВ

Ю. М. Новицкас, канд. экон. наук (СССР)  
Ю. П. Селиванов, инженер (СССР)

В Советском Союзе функционирует более 3150 вычислительных центров (ВЦ) [1], являющихся основной организационной формой применения ЭВМ для автоматизированной обработки информации в самых различных областях и для самых разных целей. Большинство ВЦ оснащены вычислительными средствами Единой системы ЭВМ, позволяющими выполнять слож-

ные технологические процессы обработки информации. Для современных ВЦ характерны многочисленные штаты, широкие связи с пользователем, большие мощности используемых технических средств. Таким образом, такие ВЦ можно назвать своеобразными информационными предприятиями, производящими специфический продукт — результативную (конечную, исходную) информацию, используемую в сфере организационного и технологического управления [2].

Производственный процесс на ВЦ, как и на предприятиях материального производства, представляет собой единство трех элементов: орудий труда, предметов труда и труда человека. Управление научно-технической и производственно-хозяйственной деятельностью ВЦ состоит в выполнении действий по обеспечению оптимального сочетания и взаимодействия всех трех элементов производственного процесса, направленных на бесперебойное функционирование ВЦ с достижением максимума полезного для общества результата, измеряемого экономическим эффектом. Анализируя эффективность деятельности ВЦ, можно выявить источники потерь и сделать выводы, полезные для разработки технических и программных средств ЭВМ (орудий труда), подготовки первичной информации и правильного определения необходимых результатов, которые могут быть получены при обработке информации (предметов труда) и организации труда персонала ВЦ (труда человека).

Решение задачи оценки результатов деятельности ВЦ требует разработки и внедрения научно обоснованной системы показателей. Для повышения эффективности деятельности ВЦ необходимо, возможно, более точно измерять ее и анализировать в различных аспектах. Показатели системы должны быть взаимодополняющими и составлять единое целое. Однако в настоящее время такая система отсутствует.

Чаще всего для оценки эффективности функционирования ВЦ предлагается использовать отдельные показатели, односторонне и упрощенно характеризующие некоторые стороны работы ВЦ. На практике экономическая эффективность деятельности ВЦ при помощи обобщенных показателей и вовсе не анализируется. Таких показателей или каких-либо исходных данных для их расчета не требует от ВЦ и ЦСУ СССР. Отсутствие научно обоснованной и принятой официально системы обобщающих, прежде всего стоимостных, показателей является одной из основных причин того, что народное хозяйство не получает полной отдачи от использования современной дорогостоящей вычислительной техники.

Ниже сделана попытка изложить принципы построения системы обобщенных показателей оценки экономической эффективности деятельности ВЦ и показать место каждого ее измерителя в предлагаемой системе на базе теории оценки результатов хозяйственной деятельности промышленных предприятий [3].

Эффективность деятельности ВЦ представляет собой отношение полученного в ее результате народнохозяйственного экономического эффекта к произведенным затратам. Как получаемый народно-

хозяйственный эффект, так и затраты не могут быть измерены одним показателем. Для этого нужна система показателей, в которой каждый показатель определяется тем, что принимается в качестве эффекта и что — в качестве затрат. Каждый показатель системы представляет самостоятельный интерес, так как характеризует определенную сторону деятельности коллектива работников ВЦ. Однако только взятые вместе они могут дать всестороннюю и обобщенную оценку деятельности ВЦ.

Прежде всего для оценки результатов деятельности ВЦ необходимо выявить полученный от него народным хозяйством экономический эффект, который характеризуется по крайней мере двумя показателями — продукцией ВЦ и полученной им прибылью (последнее характерно только для хозрасчетных ВЦ).

Главным показателем является **продукция ВЦ** — объем выполненных работ по обработке информации, который может измеряться как в натуральном, так и стоимостном выражении.

На практике чаще всего применяют такие показатели объема работ ВЦ, как норма-смена, норма-час, машино-час, человеко-день и др. В последнее время появились предложения за единицу продукции ВЦ принять символ выходного документа. Наиболее перспективным является измерение продукции ВЦ количеством решенных условных задач. При таком измерении в отличие от наиболее распространенного сейчас показателя в машино-часах стимулируется интенсивное использование оборудования ВЦ, этому способствует также мультипрограммный пакетный режим, наиболее полно использующий ресурсы вычислительной системы. Измерение же объема работ ВЦ в человеко-днях вообще не может быть показателем его продукции, поскольку этот показатель служит для измерения затрат живого труда на ВЦ. Более подробный анализ натуральных показателей продукции ВЦ не входит в задачу статьи. Необходимо лишь отметить, что установление научно обоснованной номенклатуры натуральных показателей объема работ ВЦ — практически не решенная проблема, тесно связана с измерением производительности ЭВМ и определением производственной мощности ВЦ в целом.

Обобщающие стоимостные показатели продукции ВЦ определяются умножением тех или иных объемов работ в натуральном выражении на их установленную отпускную цену. В результате можно получить, например, объем вычислительных работ в рублях, объем реализуемых услуг в рублях и др. Понятно, что обобщенным денежным показателем продукции ВЦ присущи известные недостатки, характерные для всех денежных показателей объемов продукции и услуг.

Очень важным показателем, характеризующим созданную на ВЦ продукцию, является качество выходной информации, которое может характеризоваться такими показателями, как ее содержательность, доступность восприятия, достоверность, актуальность, своевременность и т. д. [4].

Для хозрасчетных ВЦ вторым показателем экономического эффекта является прибыль. Если продукция ВЦ удовлетворяет потребность общества в определенных потребительских ценностях, то прибыль прежде всего удовлетворяет потребность в дальнейшем расширении деятельности ВЦ. Часть прибыли идет на стимулирование материальной заинтересованности работников ВЦ. Объективно прибыль — более общий показатель экономического эффекта, получаемого от работы ВЦ, чем его продукция, поскольку она является результатом как увеличения объема работ и услуг, так и снижения себестоимости единицы работ. В то же время показатель прибыли может скрыть некоторые существенные недостатки в работе хозрасчетного ВЦ, а иногда вообще искаженно представить результаты его деятельности, если, например, завышены отпускные цены на продукцию и услуги ВЦ или существует диспропорция в характере этих работ в пользу наиболее прибыльных работ в их общем объеме и т. д.

Затраты ВЦ, необходимые для создания экономического эффекта, могут быть измерены с помощью трех показателей — затрат живого труда, себестоимости продукции и объема основных и оборотных средств.

Важнейшим показателем затрат ВЦ является показатель затрат живого труда, который, зная среднесписочное количество работников, отработавших определенный период времени, можно выразить в человеко-часах, человеко-днях, человеко-месяцах или человеко-годах. Каждый из этих измерителей имеет свою область применения. Так, например, в человеко-часах принято определять затраты живого труда на техническое обслуживание ЭВМ, а в человеко-днях — затраты живого труда на разработку алгоритмов и программ. Анализируя структуру затрат живого труда, например, по категориям работ на ВЦ, можно сделать выводы для повышения эффективности деятельности работников ВЦ.

По своим функциям в производственном процессе на ВЦ весь персонал ВЦ можно разделить на следующие категории: работники, осуществляющие разработку алгоритмов и программ; работники, занятые эксплуатацией и техническим обслуживанием ЭВМ; работники, занятые эксплуатацией, техническим обслуживанием и ремонтом вспомогательного оборудования; работники, выполняющие подготовку данных на носителях; работники, занятые приемом, выпуском и сдачей материалов заказчику; административные работники; технический персонал и прочие работники. Ниже приводится состав персонала (в %) ВЦ промышленного региона по годам (см. с. 172).

Вышеприведенные данные показывают, что наибольшие затраты труда на ВЦ приходится на техническую эксплуатацию и обслуживание ВЦ (более 40%). В то же время доля затрат на разработку алгоритмов и программ за десять лет несколько упала, что связано с тем, что на ВЦ больше стали использоваться программы из фондов алгоритмов и программ, а программы стали чаще разрабатываться не на ВЦ, а в специализированных организациях.

Для народного хозяйства важно знать, насколько велики ресурсы живого труда, отвлекаемые на автоматизированную обработку информации. За последние годы их величина стала весьма значительной. В исследованном нами промышленном регионе количество работников ВЦ предприятий и организаций выросло за период 1970—1982 гг. более чем в 12 раз. Следовательно, разработка мер, направленных на сокращение затрат живого труда в процессе автоматизированной обработки информации, становится важной народнохозяйственной задачей.

|   | 1970 | 1975 | 1980 | 1982 |
|---|------|------|------|------|
| Сотрудники, занятые:  |      |      |      |      |
| разработкой алгоритмов  | 6,5  | 9,1  | 7,4  | 7,0  |
| разработкой программ  | 14,6 | 13,6 | 11,4 | 10,7 |
| эксплуатацией и техническим обслуживанием ЭВМ                           | 15,4 | 19,7 | 17,8 | 19,9 |
| эксплуатацией и техническим обслуживанием вспомогательного оборудования | 26,0 | 21,8 | 25,9 | 24,2 |
| подготовкой посетителей данных  | 9,8  | 11,3 | 10,4 | 11,1 |
| приемом, выпуском и сдачей материалов заказчику                         | 15,6 | 12,2 | 10,4 | 12,0 |
| Администрация, технический персонал, прочие работники                   | 12,1 | 12,3 | 12,7 | 15,1 |

На уменьшение затрат живого труда должно повлиять уменьшение объема работ по технической эксплуатации, обслуживанию и ремонту средств вычислительной техники как за счет развития системы централизованного обслуживания, так и за счет увеличения надежности ЭВМ и развития средств автоматизированной диагностики и восстановления.

Вторым показателем затрат ВЦ является **себестоимость произведенной на нем продукции**. Стоимость выполненных на ВЦ работ представляет собой выражение в денежной форме всех затрат и всех израсходованных средств; основной и дополнительной заработной платы персонала ВЦ с отчислениями, затрат на материалы и электроэнергию, амортизации оборудования и здания, аренды помещений, административно-управленческих и прочих общепроизводственных расходов и др. На уровень себестоимости продукции ВЦ оказывают влияние все организационно-экономические и технико-технологические факторы, имеющие место в процессе функционирования ВЦ. Здесь весьма актуальным является круг проблем, связанных с совершенствованием методов расчета себестоимости продукции ВЦ и направленных на четкое определение затрат на продукцию ВЦ, выбор объектов и калькуляции, разработку методов калькулирования и др.

Третьим из стоимостных показателей затрат ВЦ является **объем его основных и оборотных средств**. Хотя сущность и структура данного показателя хорошо известны, ряд вопросов требует решения. Среди них очень важным является вопрос включения в состав основных производственных фондов ВЦ средств програм-

много обеспечения, стоимость которых может быть сопоставимой со стоимостью технических средств. По своей экономической природе и роли в производственном процессе на ВЦ средства программного обеспечения, прежде всего системного, полностью соответствуют сущности основных производственных фондов как экономической категории.

Располагая рассмотренными выше измерителями созданного на ВЦ народнохозяйственного эффекта и показателями произведенных при этом затрат, можно получить целостную систему показателей для оценки результатов деятельности ВЦ с точки зрения основополагающих экономических понятий народнохозяйственного эффекта, затрат и эффективности. Соотношение показателя, характеризующего объем продукции ВЦ, и всех трех измерителей затрат (затрат живого труда, себестоимости продукции и величины основных и оборотных средств) дает соответственно три первых показателя эффективности деятельности ВЦ:

производительность труда работников ВЦ;

величину, обратную себестоимости единицы информационно-вычислительных работ, выполненных на ВЦ за определенный период времени;

годовой объем информационно-вычислительных работ, выполненных на ВЦ, приходящийся на 1 рубль основных и оборотных средств данного ВЦ.

Соотношение показателя, характеризующего созданный на ВЦ народнохозяйственный эффект, — прибыли и соответствующих измерителей затрат дает следующие три показателя экономической эффективности деятельности ВЦ — показатели рентабельности:

рентабельность ВЦ, исчисленную на одного работающего;

рентабельность работы ВЦ, исчисленную по отношению к себестоимости продукции ВЦ;

рентабельность работы ВЦ, исчисленную по отношению к сумме основных и оборотных средств.

Каждый из шести приведенных показателей имеет свои, только ему присущие особенности, собственный экономический смысл, специфику расчета, особый механизм действия, свою область применения. Но только взятые вместе они могут дать системное представление о результатах функционирования сложного производственного механизма современных ВЦ, а анализ этих показателей — дать информацию о мерах, которые можно предпринять для повышения эффективности ВЦ как на уровне разработчика средств ВТ, так и на уровне организации централизованного обслуживания и организации технологического процесса на ВЦ.

#### ЛИТЕРАТУРА

1. От микропроцессоров до ОГАС. — Экономическая газета, 1983, № 13, март.
2. Куденко С. П., Маринченко Б. В. Организация планирования, управления и комплектации на вычислительных центрах. — М.: Статистика, 1979.
3. Консон А. С. Экономика приборостроения. — М.: Высшая школа, 1980.
4. Скоромнюк М. А. Эффективность организационных форм использования ЭВМ. — М.: Экономика, 1978.

## VI

# Информация о новых средствах ЕС ЭВМ и СМ ЭВМ

---

УДК 681.3.06

### ПРОГРАММИРУЕМАЯ ТЕРМИНАЛЬНАЯ СИСТЕМА ЕС-8534 (ТАР-34)

Ч. Мача, инженер (ВНР)

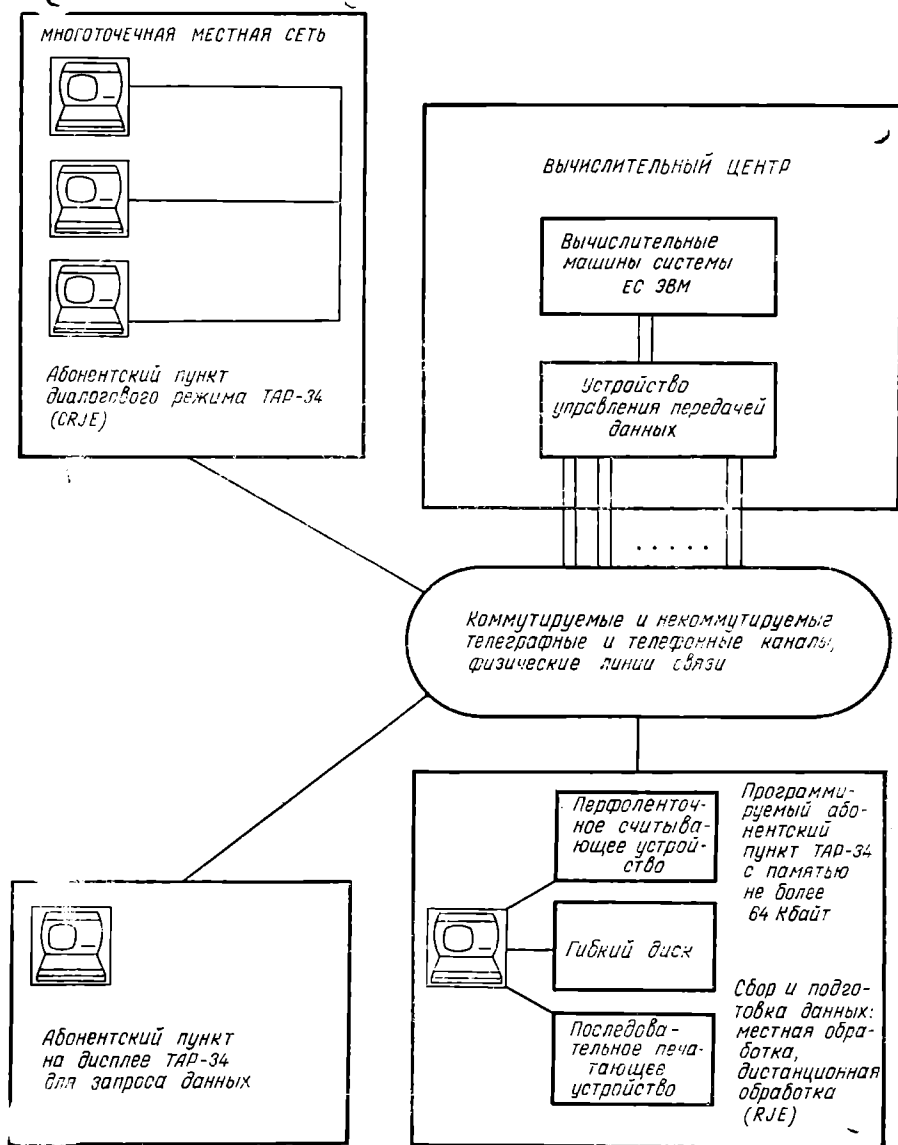
Программируемая система телеобработки данных ЕС-8534 (ТАР-34), построенная на предприятии ТЕРТА, предназначена для работы с вычислительными машинами ЕС ЭВМ. Многие устройства системы без изменений, а некоторые устройства с небольшими изменениями могут стыковаться с системой телеобработки данных, управляемой центральной вычислительной машиной другого типа.

При помощи системы ЕС-8534 (см. рисунок) осуществляется современная подготовка данных — для оператора на экране появляется сгенерированные программой форматы. Программа может не только выбрать нужный формат, но и обеспечить автоматическое заполнение фиксированных и повторяющихся полей, проверить логическую, арифметическую и синтаксическую правильность данных. Ввод данных происходит с клавиатуры с применением гибкого магнитного диска для их сбора. Двойной накопитель на гибком диске способствует выполнению сортировки данных. Шесть «мягких» клавиш и индикация их состояний на экране обеспечивают практически неограниченное количество и любое назначение функциональных клавиш, необходимых для решения пользовательской задачи. Все это не требует высокой квалификации и длительного обучения оператора.

Подготовка и обработка данных облегчаются добавочными функциями дисплея. Имеется генератор символов латинских строчных и прописных букв, а также различных национальных алфавитов. Для вычерчивания таблиц и графиков служат псевдографические символы. Возможна генерация негативного изображения, подчеркивание текста и мигание символов, вставка и стирание слов и строчки. Для редакторских целей экран разделяется на подэкраны, на которых независимо реализуются запись, стирание и исправление.

Для большого количества пользовательских задач обработка подготовленных данных может выполняться на месте при помощи

микроЭВМ терминальной системы с оперативной памятью до 64 Кбайт. Результаты обработки могут выводиться на дисплей, на матричное печатающее устройство, на накопитель с гибким магнитным диском — в зависимости от характера пользовательской задачи. Имеется возможность обработки данных, которые были подготовлены на гибком диске или на перфоленте в другом месте. Про-



Общая структура системы ЕС-8534



граммы местной обработки могут быть написаны и отлажены на языках Ассемблера и Бейсик. Поставщик системы ЕС-8534 обеспечивает ее стандартным пакетом программ и специальными пользовательскими программами.

С применением ЕС-8534 в качестве терминала на месте, где нет средней или большой ЭВМ, решаются и такие задачи, которые требуют большей памяти, большей базы данных или программ, что обеспечивается самой системой ЕС-8534 в автономном режиме. С терминала можно запускать различные задачи (в режиме ДУВЗ и УВЗ) и создавать информационно-справочные системы. Однако, если в системе пользователя в некоторых пунктах или во всех удаленных местах обработки нет необходимости в «локальной программируемости», целесообразно применять более простые и дешевые конфигурации терминала. Это возможно благодаря его модульности, позволяющей выбрать необходимые для данного пользователя внешние устройства. Терминальная система подключается к центральной ЭВМ через имеющиеся каналы связи (телефонные и телеграфные каналы, физические линии).

Система обеспечивает использование ранее приобретенных устройств и абонентских пунктов, применение малых вычислительных машин в качестве абонентских пунктов и расширение возможностей их использования.

#### Технические характеристики системы ЕС-8534

|   |   |
|---|---|
| Центральный процессор   | на базе микропроцессора<br>ИНТЕЛ8080  |
| Максимальная емкость памяти,<br>Кбайт:                            |   |
| ОЗУ   | 40  |
| ППЗУ  | 24  |
| Экран   | 1920 символов (24 строк×80<br>символов)   |
| Клавиатура  | алфавитно-цифровая, цифро-<br>вая, псевдографическая, функ-<br>циональная и «мягкие» клави-<br>ши   |
| Емкость накопителей на гибком<br>диске, Мбайт                     | 0,5   |
| Печатающее устройство:  |   |
| скорость, символов/с  | 180   |
| ширина бумаги, мм   | до 420  |
| Скорость считывающего перфолен-<br>точного устройства, символов/с | до 300  |
| Алгоритм обмена информацией                                       | стандартные стартстопные и<br>синхронные алгоритмы ЕС<br>ЭВМ, например АП-70, ЕС-7920   |
| Аппаратура передачи данных  | дуплексный модем для скоро-<br>сти 300 бит/с, полудуплексный<br>модем для скорости до 600/1200<br>бит/с, УПСНУ для скорости<br>9600 бит/с |
| Языки программирования  | Ассемблер, Бейсик   |
| Тесты   | микропрограммная диагности-<br>ческая система   |
| Диапазон рабочих температур, °С                                   | от 5 до 40  |
| Параметры сети, В; Гц   | 220 $\pm 10\%$ ; 50 $\pm 2\%$<br>220 $-15\%$ ; 50 $\pm 2\%$   |

П. П. Подпорин, инженер (СССР)  
И. Д. Савин, инженер (СССР)  
З. Ф. Олейникова, инженер (СССР)

При массовом внедрении СМ ЭВМ разработка прикладных программ представляет наибольший интерес, поскольку они являются средством автоматизации программирования, ускоряют внедрение средств вычислительной техники в различные сферы человеческой деятельности и в значительной степени определяют экономическую эффективность использования вычислительных комплексов.

Пакеты прикладных программ по своей структуре можно организовать в библиотеку программ. При этом нужно хорошо различать эти понятия. *Пакеты прикладных программ* — это обособленные элементы, из которых состоит система прикладного программного обеспечения. Внутренняя реализация различных пакетов может быть неодинаковой. В понятие *библиотека программ* вкладывается один из возможных принципов внутренней реализации пакета и его эксплуатации [1].

Библиотека программ для научно-технических расчетов (БНТР), которая рассматривается ниже, предназначена для использования в точных теоретических расчетах, в инженерном проектировании, в лабораторном эксперименте и других областях науки и техники. Сфера применения БНТР благодаря постоянному расширению использования СМ ЭВМ не ограничивается только рамками инженерных и научных задач.

При разработке БНТР, учитывая многогранность ее применения, были использованы следующие технологические принципы построения [2]:

конструктивная универсальность. БНТР имеет универсальную конструкцию, одинаковую с конструкциями ранее разработанных пакетов;

модульность программных объектов. Структура БНТР является дискретной и ее программные единицы (программы) унифицированы. Программы библиотеки оформлены как подпрограммы в соответствии с требованиями базисного языка. Это дает возможность пользователю формировать различные вычислительные схемы для решения конкретных задач. Унификация программ БНТР заключается прежде всего во взаимодействии с данными на основе соглашений базисного языка;

стандартизация взаимодействия программных модулей с данными. Программы библиотеки предусматривают единую методику и единый механизм объединения программ и данных. Программы БНТР не содержат операций ввода-вывода, которые должны обеспечиваться программами пользователя;

машинная независимость. БНТР предлагает преемственность ее программного и информационного обеспечения при смене типов

ЭВМ. Реализация данного принципа обеспечена тем, что в качестве базисного языка принят Фортран-IV в силу его распространения и значительной библиотечной оснащенности;

максимальная независимость от операционных систем. Библиотека может быть реализована на различных ЭВМ, имеющих в составе операционных систем транслятор Фортран-IV. В частности, БНТР выполняется под управлением операционных систем ДОС СМ, ОС РВ, ФОБОС, РАФОС на УВК СМ-3 и СМ-4;

расширяемость. БНТР является открытой библиотекой, и пользователь имеет возможность непрерывно пополнять ее новыми программами;

«коммуникабельность». Библиотека программ проста в обращении.

Отдельными программами БНТР или их комбинациями можно обрабатывать данные, используя методы математической статистики, линейной алгебры, численного анализа.

В большинстве программ реализованы широко используемые методы математики, например:

при решении систем линейных уравнений применяется метод Краута;

диагонализация с помощью QR-метода и формирование сдвигов по методу Ньютона используются для нахождения собственных значений трехдиагональной матрицы;

с помощью итерационного метода Ньютона — Рафсона вычисляются действительные и мнимые корни полинома;

корни системы нелинейных уравнений вычисляются по методу Брауна;

метод Рунге — Кутта применяется для решения системы обыкновенных дифференциальных уравнений. Аналогичные системы, но с заданными начальными значениями решаются трехточечным методом Адамса;

для вычисления интеграла от функции, заданной пользователем, применяется правило трапеций с экстраполяцией по методу Ромберга или приближенная формула Гаусса.

Некоторые программы не используют общие методы, и вычисления производятся по формулам или способам, специально разработанными для конкретной задачи. К ним относятся вычисление среднего значения, дисперсии, статистик Колмогорова — Смирнова, статистик Стьюдента, преобразования временных рядов.

С позиций повышения эффективности применения БНТР существенное значение имеет точность вычислений. Достоинством библиотеки является возможность ряда программ обеспечивать как одинарную, так и двойную точность.

По своему функциональному назначению библиотека, насчитывающая в своем составе более 230 программ, состоит из пяти разделов: линейная алгебра, численные методы анализа, статистика, специальные функции, сервисные программы, которые для удобства использования разбиты на 22 подраздела.

Раздел «Линейная алгебра» [3] включает программы, выполняющие функции хранения матриц, линейных преобразований, умножения и деления матриц, факторизации матриц и определения ранга, решения систем линейных уравнений и анализа собственных значений.

Программы раздела «Численные методы анализа» [4] выполняют операции с полиномами, решают системы нелинейных уравнений и обыкновенные дифференциальные уравнения, выполняют численное интегрирование.

Раздел библиотеки «Статистика» [5] обеспечивает отбор данных, корреляцию и регрессию, анализ планирования, дискриминантный и факторный анализы, преобразование временных рядов, вычисление непараметрических статистик, образование случайных чисел, вычисление функций распределения и вычисление элементарных статистик.

Программы раздела «Специальные функции» выполняют вычисления значений специальных функций: гамма-функций, Бесселя, Неймана, Кельвина, эллиптических интегралов, интегральных синуса и косинуса, гиперболических функций.

Сервисные программы библиотеки обеспечивают: чтение матриц и векторов, вычисление времени работы программного блока, печать различных форм представления экспериментальных данных (гистограммы, функции распределения), печать матриц, векторов и графиков функций.

Особенностью разработки БНТР является простота генерации, в процессе которой обеспечивается одно из преимуществ библиотеки — мобильность. В это понятие вкладывается возможность изменения состава (и, естественно, сферы применения) библиотеки, который определяется при генерации в интерактивном режиме.

Генерация выполняется под управлением интерактивного макропроцесса ПАГЕН [6]. Во время генерации пользователю задается ряд вопросов, ответы на которые определяют используемую конфигурацию технических средств ЭВМ, состав библиотеки, необходимый пользователю, и имена устройств, на которые установлены носители с дистрибутивными файлами БНТР и целевой носитель для сгенерированной библиотеки.

Для выполнения генерации полного объема БНТР особое значение имеют размеры свободных областей дистрибутивного и целевого носителей, которые должны иметь не менее 2 тыс. свободных блоков на дистрибутивном носителе (для размещения объектных модулей после трансляции) и 3 тыс. свободных последовательных блоков на целевом носителе (для записи сгенерированной библиотеки). При генерации неполного состава БНТР эти требования к размерам свободных областей могут быть значительно снижены.

В процессе генерации пользователь решает вопрос, включать или нет в состав БНТР тот или иной подраздел, учитывая взаимосвязь программ различных подразделов при их выполнении и вероятность превышения максимального объема сгенерированной библиотеки — 2 тыс. блоков. Информация о взаимосвязи программ

представлена в документации на БНТР. При затруднении ответа по типу «да/нет» имеется возможность получить во время генерации дополнительную информацию по каждому вопросу. Вся процедура занимает от 1—2 до 35—40 мин в зависимости от выбранного состава библиотеки. В случае получения правильно сгенерированной библиотеки на экране терминала появляется сообщение об успешном завершении процесса генерации.

Применение библиотеки программ БНТР позволяет:

повышать эффективность труда научных и инженерно-технических работников;

сокращать сроки выполнения работ;

сокращать затраты на организацию и проведение лабораторных экспериментов;

расширять возможности и повышать эффективность использования конкретной операционной системы и вычислительных средств СМ ЭВМ;

обеспечивать сервис и сокращать время на создание пользовательской библиотеки программ.

Программная документация на библиотеку программ выполнена машинным способом и занимает на сменной магнитной кассете ЕС-5269—01 около 1000 блоков. Библиотека программ для научно-технических расчетов и документации на нее (в составе операционной системы РАФОС) передана в НПО «Центр программсистем» (г. Калинин).

#### ЛИТЕРАТУРА

1. Фатеев А. Е., Ройтман А. И., Фатеева Т. П. **Прикладные программы в системе математического обеспечения ЕС ЭВМ.** — М.: Статистика, 1976.
2. **Принципы построения и архитектура пакета прикладных программ**/Д. Е. Заполоцкий, С. Н. Карпенко, С. Г. Кузин и др. — УСиМ, 1978, № 1.
3. Тьюарсон Р. **Разряженные матрицы.** — М.: Мир, 1977.
4. Данилина Н. И., Дубровская Н. С. и др. **Численные методы.** — М.: Высшая школа, 1976.
5. Корн Г., Корн Т. **Справочник по математике для научных работников и инженеров.** — М.: Наука, 1978.
6. Вигдорчик Г. В. **Интерактивный макропроцессор на языке Фортран-IV.** — В кн.: Программное обеспечение управляющих комплексов системы малых ЭВМ. — Труды ИНЭУМ. М., 1980, вып. 80.

УДК 681.3.06

#### ЯЗЫК ПРОГРАММИРОВАНИЯ ДИКОЛ

Я. Трглик, канд. техн. наук (ЧССР)

Для непосредственного управления производством, оборудованном и технологическими процессами разрабатывается ряд специализированных вычислительных си-

стем на базе СМ ЭВМ. Эффективность эксплуатации современной вычислительной техники повышается при использовании языков программирования, ориентированных на эту область применения. Одним из новых языков программирования является ДИКОЛ (Discrete Control Language), представляющий собой проблемно-ориентированный язык для управления дискретными процессами.

Основная секция программы на языке ДИКОЛ называется *процессом*. Отдельные алгоритмы процесса могут быть представлены в виде процедур. Задача разделяется между процессами дистрибутивно. Из этого, например, следует, что ввод и вывод должны быть разделены между отдельными процессами. Такое правило ведет к структуризации разработанных программ и помогает распределенному проектированию управляющей системы.

Запись всех операторов осуществляется одинаково. Она содержит следующие четыре части:

название: ОПЕРАТОР параметры комментариев

Последовательность операторов одной строки образует операторную строку. Отдельные операторы в операторной строке должны быть отделены двоеточием. Название может быть указано только перед первым оператором в строке.

Операторная строка является простейшей управляющей структурой. В операторах, в которых разветвляется последовательность реализации операторов, всегда осуществляется выбор между продолжением на следующий оператор в операторной строке и продолжением на первый оператор следующей операторной строки.

Из операторных строк образуются программные секции процедуры, основная часть процесса и заголовок программы. Основная часть процесса вместе с соответствующими процедурами образует процесс, а заголовок программы вместе с процессами — программу.

**Основные операторы.** Операторами PROGRAM, PROCESS и PROCEDURE начинаются программная секция, заголовок программы, основная часть процесса или процедура. Для окончания этих секций служит оператор END.

В каждой программной секции специфицируются формальные связи. Оператором SLAVE в заголовке программы описываются включенные процессы, в основной части процесса этим оператором специфицируются включенные процедуры. Оператором MASTER, наоборот, в основной части процесса описывается программа, для которой процесс был разработан, а в процедуре — процесс, которому процедура принадлежит. С помощью оператора EXTERN в основном процессе описываются внешние процедуры, которые будут использоваться в этом основном процессе.

С помощью языка ДИКОЛ можно представить три типа информации: однобитовую — логическую [L], восьмибитовую — знаковую [C] и многобитовую — числовую [N].

Оператором CONST определяются логические, знаковые или числовые константы. Величины приводятся всегда за названием определяемой постоянной в круглых скобках, например:

CONST.L. PRAVDA (1), LOZ (0) — логические постоянные 1 и 0;  
 CONST.C PLUS ('+') — знаковая постоянная PLUS знака +;  
 CONST.N DESAT (10) — числовая постоянная 10 с названием DESAT.

Кроме того, переменные разделяются на входные (INPUT) выходные (OUTP) и рабочие (VAR), например:  
 INPUT.L SIGNAL — входная логическая переменная SIGNAL;  
 OUTP.L MOTOR — выходная логическая переменная MOTOR;  
 VAR.C INDEX — рабочая знаковая переменная INDEX;  
 VAR.N CITAC (5) — рабочая числовая переменная CITAC с исходной установкой на величину 5.

Если в программной секции используются процедуры, то их нужно определить оператором USE. Названия процедур, приведенных в параметрах оператора USE, одновременно являются переменными логического типа, которые постоянно хранят информацию о том, действует ли означенная процедура или нет.

**Исполнительные операторы.** Первому исполнительному оператору каждой программной секции должно предшествовать системное название BEGIN. Всегда при запуске программной секции или при ее продолжении от этой метки все выходные и рабочие переменные устанавливаются на исходную величину.

Основными исполнительными операторами являются операторы ASSIGN, DELAY, COMP, WAIT, CALL, START, STOP и EXIT. Этими операторами можно выполнить большинство управляющих алгоритмов.

Оператор ASSIGN присваивает переменным, указанным во втором и следующих параметрах, величину первого параметра.

Оператор DELAY фиксирует неизменность состояния программной секции на интервал времени, равный мгновенному значению его параметра.

Оператор COMP решает вопрос о дальнейшем продолжении обработки. Если результат сравнения имеет логическую величину 1, обрабатывается следующая операторная строка; если величину 0, процесс продолжается исполнением следующего оператора той же самой операторной строки.

Оператор JUMP определяет безусловный переход на исполнение операторной строки, которая отмечена меткой в параметре оператора JUMP.

Оператор WAIT имеет два параметра. В первом параметре должно быть указано название переменной или выражение логического типа. Если этот параметр равен 1, обрабатывается следующая операторная строка, если 0, исход решения определяет второй параметр. Если оператор не специфицирован, исполнение программной секции задерживается по оператору WAIT, пока первый параметр не будет равен 1. Если второй параметр — специфицированная переменная или выражение числового типа, задержка по команде WAIT ограничивается временным интервалом, определенным мгновенной величиной этого параметра. Если с этого времени

первый параметр не достигнет 1, продолжается использование следующего оператора в той же самой операторной строке.

Оператор CALL служит для вызова процедуры того самого или другого процесса или подпрограммы. После исполнения оператора EXIT в стартовой процедуре продолжается исполнение оператора, следующего за оператором CALL.

Оператор START вызывает одновременный запуск процедур, названия которых указаны в параметрах оператора. Обработка программной секции не останавливается, а продолжается исполнением оператора, следующего после оператора START, несмотря на продолжение в обработке запускаемых процедур.

STOP — оператор, противоположный оператору START, он вызывает остановку исполнения всех процедур, названия которых указаны в параметрах оператора STOP.

Оператор EXIT означает конец всех операций алгоритма процедуры или основной части процесса.

**Дополнительные операторы.** Оператор присвоения ASSIGN чаще всего употребляется при отладке величины логической переменной. Для установки ее на единичное значение удобнее использовать оператор SET, для установки на нулевое значение — оператор RESET, для изменения на обратную величину — оператор INVERT. Чаще всего применение оператора ASSIGN на знаковые и числовые переменные упрощают операторы сброса CLEAR, прибавления единицы INCR и вычитания единицы DECR. Этот оператор одновременно проверяет результат на нуль, после чего осуществляет ветвление программной секции. Следующими модификациями оператора ASSIGN являются операторы логических и арифметических операций.

Более простое программирование счетчиков обеспечивается определением числовых переменных оператором COUNT с заданием границы переполнения. Оператор INCR осуществляет автоматическое ветвление исполнения процедуры.

Со знаковыми и числовыми данными часто работают не отдельно, а группой. Оператор STRING позволяет определять точную цепочку знакового [C] или числового [N] типа. Аналогично оператор RECORD позволяет определять знаковый [C] или числовой список, элементы которого и размер могут меняться.

Оператор CLEAR очищает список, оператор ASSIGN присваивает списку наполнение цепочки или другого списка. Оператор COMP сравнивает два объекта на идентичность и в зависимости от результата осуществляет ветвление процесса.

Оперирование с элементами списка позволяет установить системное имя POINTER, указывающее на актуальный элемент списка. Оператором FETCH этот элемент копируется в обозначенную переменную, оператором PUSH, наоборот, величина определенной переменной записывается как элемент в список, на который указывает установка переменной POINTER. Оператор SEARCH позволяет найти в списке положение элемента с данной величиной.



Следующими операторами для работы с цепями и списками являются операторы ORIGIN, CHANGE, JOIN, CUT.

Связь и синхронизацию между процедурами осуществляют операторы SEND, TAKE, PRIVATE, PUBLIK и LOOK.

Оператором SEND информация отсылается к процедуре, которая принимает ее оператором TAKE. Оператором PRIVATE процедура присваивает себе использование объектов, определенных в основной части процесса, для исключительного использования. Оператор PUBLIK возвращает их для общего пользования.

Оператор LOOK предотвращает остановку процедуры на операторах TAKE или PRIVATE, он позволяет также разветвлять исполнение процедуры в зависимости от того, является ли ожидаемая информация отосланной или готова ли определенная переменная к присвоению.

Системные названия цепей CLOCK и DATE дополняют мониторинговые сообщения точным значением реального времени.

Операции ввода-вывода через системную периферию осуществляют операторы READ и WRITE.

Основные операции над набором данных можно реализовать с помощью операторов FILE, OPEN, CLOSE, GET, PUT.

**Обработка отказов.** Запись управляющих алгоритмов с помощью операторов языка ДИКОЛ предполагает детерминистический характер алгоритмов. Несмотря на это, могут возникнуть ошибки в исполнении операторов (например, при делении на нуль, переполнении и т. д.) или в программной секции могут быть обнаружены отклонения от предполагаемого хода обработки. Эти так называемые «отказы» можно обработать последовательностью обработки, которая начинается операторной строкой с системным названием ABORT. Причину отказа анализирует ряд дополнительных системных имен.

Возвращение из этого режима в нормальный режим сигнализируется оператором CORRECT. Оператор CONTINUE обеспечивает продолжение программы на том месте, где был отказ.

Ошибку внутри программной секции обнаруживает оператор ERROR.

Оператор SCRATCH служит для перехода в режим обработки отказа программной секции, работа которой непосредственно связана с программной секцией, где была обнаружена неисправность.

**Пример**

Манипулятор выдвигает захват, захватывает деталь, поднимает ее, поворачивает на 180°, кладет ее обратно мягко, легко и в исходном положении ожидает повторного пуска.

Решение задачи разработано в форме процедуры с названием OBRACAC.

OBRACAC: PROCEDURE : поворот детали  
: входные сигналы

INPUT.L VYSUNUTY,ZASUNUTY,HORE,DOLE,OTOCENY : выходные сигналы

VYSUN:OUTP.L TAM/O,SPAT/1/ : управление движением

ZDIN:OUTP.L NAHOR/O,/,DOLU/1/ : управление подъемом

|        |                    |                 |                            |
|--------|--------------------|-----------------|----------------------------|
| OUTP.L | KLIESTE/O/         |                 | смыкание захвата           |
| BEGIN: | CONST.N ZOTRV/200/ |                 | числовая постоянная 200 мс |
|        | INVERT VYSUN       | : WAIT VYSUNTY  | производительные команды   |
|        | SET KLIESTE        | : DELAY ZOTRV   | выдвижение к детали        |
|        | INVERT ZDVIH       | WAIT HORE       | захват детали              |
|        | SET OTOC           |                 | подъем                     |
|        | WAIT.R OTOCENY     | : WAIT OTOCENY  | поворот                    |
|        | INVERT ZDVIH       | : WAIT DOLE     | опускание                  |
|        | RESET ZDVIH.DOLU   |                 | останов опускания          |
|        | RESET KLIESTE      | : DELAY ZOTRV   | освобождение детали        |
|        | SET ZDVIH, DALU    |                 | окончание опускания        |
|        | INVERT VYSUM       | : WAIT ZASUNUTY | включение                  |
|        | EXIT               |                 | конец работы               |
|        | END CBRACAC        |                 | конец процедуры            |

Заключение. Свойства языка ДИКОЛ были проверены при записи алгоритмов для разных областей применения, например для управления автоматизированными сборочными линиями, роботизированными обрабатывающими центрами, позиционными столами с шаговыми двигателями и т. д. Другие дискретные процессы можно обрабатывать с помощью транслятора языка ДИКОЛ, разработанный также на языке ДИКОЛ.

Транслятор языка ДИКОЛ предназначен для вычислительных машин СМ-3, СМ-4 и СМ-50/50, а также для микроЭВМ типа СМ-50/40.

УДК 681.3.06

### ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УПРАВЛЕНИЯ ДАНЫМИ ДЛЯ МИКРОЭВМ СМ-1630

К.-Х. Виггерт, инженер (ГДР)

Для системы микроЭВМ СМ-1630 предприятие Комбинат «Роботрон» предлагает системное и стандартное программное обеспечение. Системное программное обеспечение включает операционные системы, например модульную операционную систему MOOS 1600 [1], а также компиляторы. Под стандартным программным обеспечением понимаются централизованно разработанные и проверенные программные средства широкого применения: модульная база MOVA 1600; организация данных DATO 1600; программное обеспечение для математических процедур MAVЕ 1600.

При помощи этих программных средств могут обрабатываться задачи в пакетном режиме, а также в режиме реального времени.

Во всех областях применения СМ-1630 центральное место занимают средства организации данных. К ним относится DATO 1600, которая применяется для построения, управления и использования баз данных.

ДАТО 1600 предлагает три варианта средств управления данными различного функционального объема (рис. 1):  
 систему обработки файлов DAVE 1600;  
 систему управления записями RMS 1600;  
 операционную систему управления банком данных DABA 1600.  
 На рис. 1 показано содержание этих программных средств.

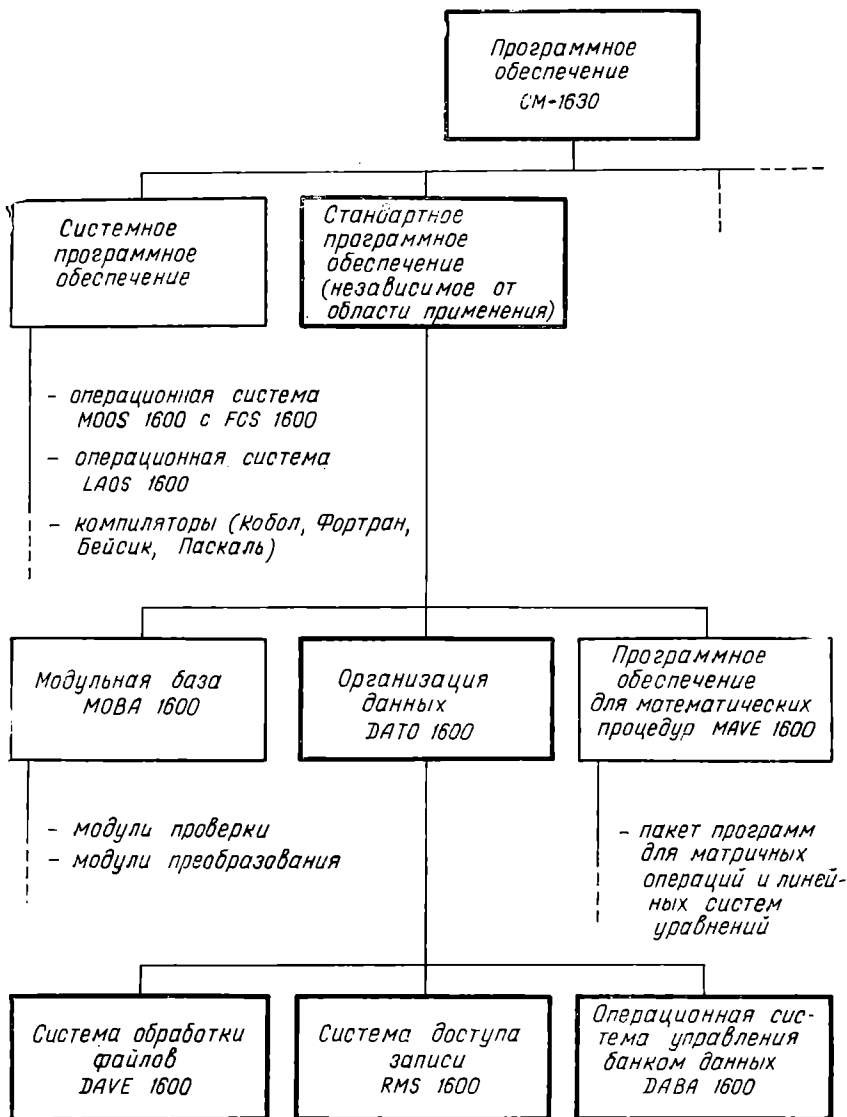


Рис. 1. Место организации данных ДАТО 1600 в системе программного обеспечения СМ-1630

DATE 1600 разработана для использования на микроЭВМ СМ-1630. Ее можно принять и на совместимых ЭВМ, например на СМ-3/СМ-4. Система разрабатывалась в Научно-исследовательском центре комбината «Роботрон» при участии факультета обработки информации Дрезденского технического университета.

Первая система DAVE 1600 помогает пользователю при организации, управлении и обработке больших систем файлов. Она соответствует требованиям, предъявляемым к технологической системе программного обеспечения для микроЭВМ СМ-1630 [3]. DAVE 1600 является набором программно-технических средств; основные из них:

- модульная операционная система MOOS 1600 с программами доступа FCS 1600 (File Control Service 1600) [4];

- модули из модульной базы MOVA 1600.

DAVE 1600 состоит из отдельных модулей, комплексных модулей и подпрограмм обработки на языке Ассемблера MACRO 1600.

Эта система предоставляет следующие средства для описания базы данных (рис. 2):

- статические элементы, которые определяют количество одновременно обрабатываемых файлов, дополняя тем самым функции описания записи или файлов;

- динамические элементы определения, которые поддерживают построение и использование таблиц индексов;

- модули открытия и закрытия файлов, а также модули для обработки записей файлов с последовательной, прямой и индексно-последовательной организацией;

- комплексные модули для типичных процессов обработки (дублирование, разделение, объединение);

- программы обработки для сортировки последовательных файлов на магнитных дисках.

Другие компоненты системы служат для протоколирования и ведения диалога. В соответствии с имеющимися версиями, описанными в [3], DAVE 1600 может применяться в программах пользователя с языками MACRO 1600, Фортран 1600 и Кобол 1600. При разработке программы пользователя предоставляемые через DAVE 1600 процессы обработки данных или другие типичные процессы хорошо вписываются в блок-схему проекта или в структурную схему. При программировании эти компоненты выступают в программе пользователя как готовые макропрограммы или подпрограммы. Таким образом, пользователь освобождается от организации ввода-вывода и управления. Подобная информация о DAVE 1600 имеется в [5].

**Система управления записями RMS 1600 (Record Management System 1600)** упрощает обработку записей и файлов в программах пользователя, в которых должны накапливаться и обрабатываться большие объемы данных. Система построена по модульному принципу и имеет удобный интерфейс для встраивания в программы пользователя.

При помощи RMS 1600 могут быть обработаны последовательно, прямо или индексно-последовательно организованные файлы. Она обеспечивает следующие виды доступа:

- последовательный для всех видов организации файлов;
- свободный выбор с доступом через:
  - номер записи при прямо организованных файлах;
  - ключ при индексно-последовательно организованных файлах;
- адрес записи файла при любых видах организации файлов на магнитных дисках.

Система RMS 1600 позволяет сменить вид доступа в процессе обработки. Возможен параллельный доступ к одному файлу из нескольких программ пользователя. Интерфейс RMS 1600 (рис. 3) содержит:

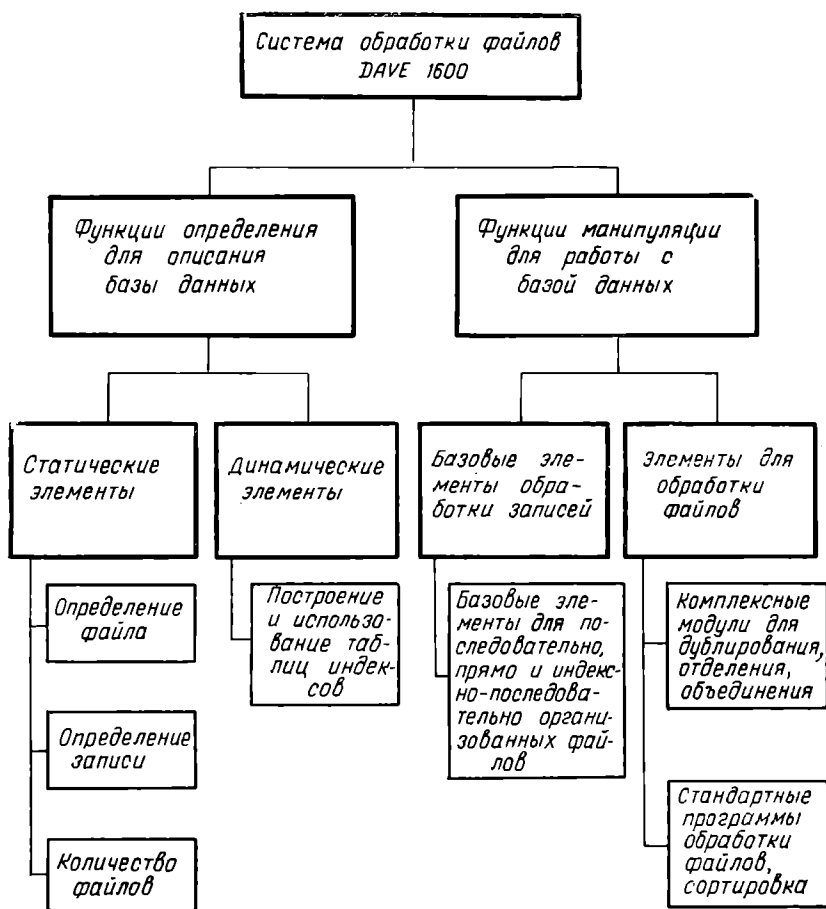


Рис. 2. Составные части системы обработки файлов  
DAVE 1600

описание программной среды — описание области памяти и определение процесса обработки данных, т. е. конкретно указываются требуемые от RMS 1600 элементы, определяется процесс обработки записей, например вид доступа, его направление и т. д.;

доступ к полям управляющего блока во время обработки (для этого имеются макроинструкции, влияющие на значение текущего времени в полях блока управления).

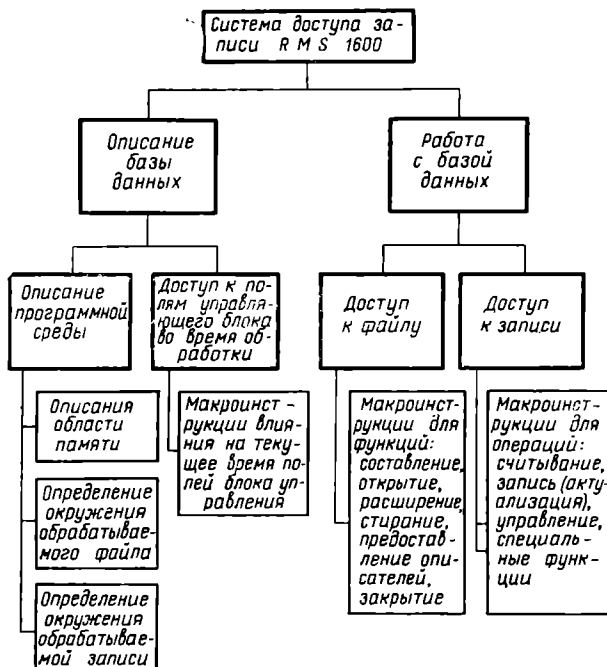


Рис. 3. Основные компоненты системы управления записями RMS 1600

Для работы с базой данных предоставляются:

средства доступа к файлу (макроинструкции для составления, открытия, расширения, стирания, предоставления описателей и закрытия файлов);

средства доступа к записи (макроинструкции для операций считывания, записи (актуализации), управления, например позиционирование или возврат, и специальные функции, например стирание или сокращение записей).

Наряду с набором макроинструкций и сервисных средств, реализованных в виде объектных модулей, диалоговая система управляет обменом информации между программой пользователя и пользователем в обоих направлениях. Ввод-вывод, которые осуществляются через терминал, могут быть проведены без дополнительных затрат на программирование. Все доступные для пользовате-

ля компоненты RMS 1600 написаны на языке Ассемблера MACRO 1600.

Более подробные сведения о RMS 1600 приведены в [6].

Система DABA 1600 представляет собой современную операционную систему управления банком данных. Ее основой является модуль отношений CODD, описанный в [7], причем пользователю предоставляется вся информация об управляемых данных в виде

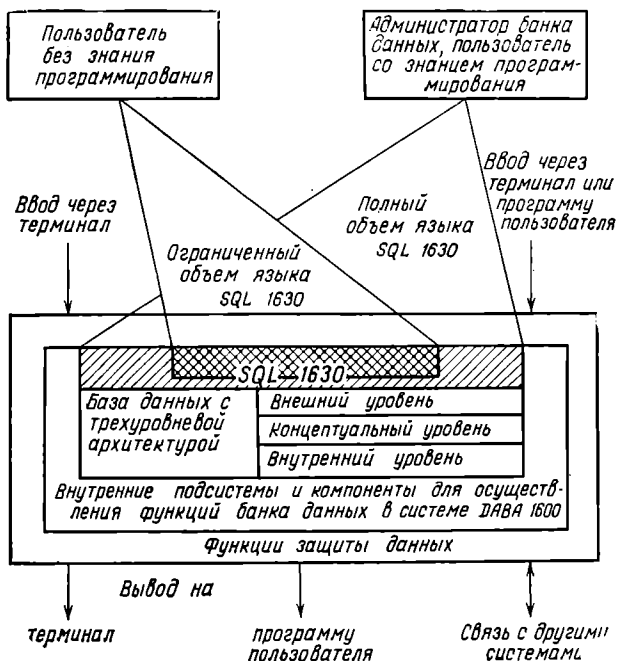


Рис. 4. Возможности использования операционной системы управления банком данных DABA 1600

таблиц. Возможно представление данных в сетевой или в древовидной структуре. DABA 1600 со своей трехуровневой структурой (рис. 4) в основном соответствует рекомендациям ANSI [8]:

концептуальный уровень отражает единое множество информационной базы всех представленных в банке данных областей применения;

внешнему уровню присваиваются подмножества данных конечного пользователя (как правило, они представляют собой часть концептуального общего множества и могут обладать собственной структурой в относительном представлении);

во внутреннем уровне реализуется конкретный массив информации.

Для коммуникации пользователя с системой DABA 1600 предназначен язык управления банком данных SQL 1630 — специальная версия SEQUEL 2 [9] для СМ-1630.

Язык SQL 1630 обеспечивает:

определение данных;

манипуляции с данными;

запросы, после которых проводится некоторая подготовка данных, например, упорядочение по максимальным значениям, определение экстремальных и средних значений или подсчет значений результатов;

контрольные функции;

инструкции для включения элементов языка SQL 1630 в программы пользователя, написанные на языке Ассемблера MACRO 1600 или на Коболе 1600.

Пользователи, имеющие некоторые знания в области программирования, могут формулировать запросы к массиву данных на этом языке, так как его ядро можно легко изучить. Опытные программисты могут полностью использовать возможности языка.

Доступ к записям или наборам записей может производиться не только последовательно, но и выборочно при использовании внутренней стратегии поиска:

по первичному ключу (по признаку порядка, присвоенному однозначно каждой записи);

по вторичному ключу (по общим признакам, которые могут быть в нескольких записях).

Доступ к данным при этом обеспечивается системой в зависимости от запроса. DABA 1600 обладает необходимыми для многопользовательской системы функциями защиты данных.

Система DABA 1600 применяется в основном в справочных системах (диалоговый опрос постоянного массива данных), в системах бронирования мест (диалоговый опрос и связанные с этим многочисленные изменения в массиве данных), в системах прикладных программ (запрос информации из массива данных для любых программ пользователя).

Для целесообразного использования ресурсов пользователь сможет заранее сгенерировать нужные формы применения. При помощи обмена носителями может быть установлена связь с другими системами управления банками данных, имеющими тот же или более высокий уровень.

Более подробная информация о DABA 1600 приведена в [10].

DAVE 1600 и RMS 1600 действуют в системах мини-ЭВМ СМ-1630 как в модификациях robotron К 1620 и robotron К 1630. Эти модели отличаются друг от друга быстродействием и емкостью памяти, причем К 1630 является более производительной моделью. Выбор наиболее эффективной системы для работы с файлами при использовании DAVE 1600 или RMS 1600 в первую очередь зависит от конкретных требований. Готовыми конфигурациями являются, например, коммерческие базовые системы ЭВМ А 6401 (с



моделью К 1620) и А 6402 (с моделью К 1630). DABA 1600 предполагает работу с моделью К 1630, например, в коммерческой базовой системе ЭВМ А 6402.

Другие аппаратно-технические требования вытекают из условий применения для работы с файлами или с банком данных, особенно с учетом устройства ввода-вывода и накопителей. Во всех случаях в конфигурации СМ-1630 требуется наличие устройства управления и накопителя на магнитном диске СМ ЭВМ, например СМ-5400. Для работы системы RMS 1600 требуется как минимум два накопителя на магнитном диске, для DABA 1600 — четыре накопителя на магнитном диске. В справочных системах или системах бронирования мест количество подключаемых терминалов определяется специфическими требованиями пользователя. Ограничения задаются возможностями подключения периферийных устройств вообще и нагрузкой на магистраль ЭВМ от подключенных устройств.

Для работы с базами данных все системы используют системное программное обеспечение MOOS 1600, причем DABA 1600 требует версии начиная с 1.2.

Опыт применения DAVE 1600 подтверждает широкие возможности системы в сочетании с модулями из MOVA 1600.

Готовые модули ввода-вывода или модули обработки и независимые от пользователя стандартные программы упрощают в значительной степени работу с файлами. Они обеспечивают наглядность структуры программы, позволяют отказаться от программирования и отладки стандартных функций обработки файлов и значительно уменьшить затраты на программирование (в некоторых случаях экономия времени достигает 30—40%).

Система RMS 1600 применяется, например, для обработки файлов при управлении с помощью ЭВМ складами. Система объектных модулей позволяет осуществлять передачу данных в виде логических записей из программы к файлу и наоборот. Пользователю не надо знать физической структуры накопителей и формы хранения в памяти. Макроинструкции и запись параметров отличаются наглядностью. Большое количество средств диагностики ошибок помогает пользователю при разработке его программ. Приемлемо более длительное время трансляции, если учитывать большие затраты на локализацию и устранение ошибок, которые обычно обнаруживаются только во время обработки программы.

Пользователю DABA 1600 предоставляются все преимущества, которые открываются при работе с банками данных, например удобный для пользователя и похожий на естественный язык диалога, более простая актуализация данных при разделении программы и файлов, более простое обслуживание с низкой избыточностью накопленных данных, концентрация всех средств защиты данных различных групп пользователей, простая привязка системы к предусмотренной цели применения. Сохранность данных достигается с помощью дубликатов. Это позволяет восстанавливать данные после аварии. Язык SQL 1630 удобен для применения как пользовате-

лю, так и программисту. Система DABA 1600 имеет гибкую структуру и позволяет вводить расширения, нечувствительна к изменению конфигурации ЭВМ.

#### ЛИТЕРАТУРА

1. Фойгт К.-У. **Модульная операционная система МООС 1600 для СМ-1630.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1983, вып. 14.
2. Hientzsch E. U. a. **Problemorientierte Systemunterlagen für das Mikrorechnergerätesystem robotron K 1600.** — Übersicht und DATO 1600. — Neue Technik im Büro (NTB), 26 (1982), H. 5.
3. Хорн Э., Тач Б. **Технологическая система для разработки программного обеспечения микроЭВМ СМ-1630.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1983, вып. 14.
4. Ghädig M. **Die maschinenorientierten Systemunterlagen für das Mikrorechnergerätesystem robotron K 1600.** — NTB 27 (1983), H. 4.
5. Stumpf J. **Problemorientierte Systemunterlagen für das Mikrorechnergerätesystem K 1600.** — Dateiverarbeitungssystem DAVE 1600. — NTB 27 (1983), H. 2.
6. Hientzsch E. **Problemorientierte Systemunterlagen für das Mikrorechnergerätesystem robotron K 1600.** — Satzzugriffssystem RMS 1600. — NTB 27 (1983), H. 3.
7. Codd E. F. **A relational model for large shared data banks.** — Communications of the ACM, 1970, vol. 13, No. 6.
8. **ANSI/X3/SPARC Study Group on DBMS.** — Interim Report, Febr. 1975.
9. Chamberlin D. D. et al. **SEQUEL 2: A unified approach to data definition, manipulation and control.** — Journal of Research and Development, 1976, vol. 20, No. 6.
10. Bulla G. u. a. **Problemorientierte Systemunterlagen für das Mikrorechnergerätesystem robotron K 1600.** — Datenbankbetriebssystem DABA 1600. — NTB 26 (1982), H. 6.

## СОДЕРЖАНИЕ

### I. Международное сотрудничество социалистических стран в области вычислительной техники

|   |   |
|---|---|
| <i>Терехов Ю. В.</i> Микроэлектронная элементная база для средств вычислительной техники . . . . .    | 3 |
| <i>Раковский М. Е., Жукова Т. В.</i> Организация работ по развитию программного обеспечения . . . . . | 8 |

### II. Технические средства вычислительной техники

|   |    |
|---|----|
| <i>Ломов Ю. С.</i> Основные направления развития высокопроизводительных ЭВМ общего назначения на основе БИС . . . . .                                     | 14 |
| <i>Файзулаев Б. Н., Малярский Н. М.</i> БИС на основе базового матричного кристалла — элементная база высокопроизводительных ЭВМ Единой системы . . . . . | 18 |
| <i>Щетинин Ю. И.</i> Повышение быстродействия и коэффициента программируемости биполярных ППЗУ серии КР556 . . . . .                                      | 24 |
| <i>Знаменачек З., Штепанек Я., Лакатош И.</i> Матричные БИС для средств вычислительной техники . . . . .  | 31 |
| <i>Динев Р., Мукарева Кр., Петров М.</i> Система для логического моделирования цифровых схем . . . . .  | 37 |

### III. Программное обеспечение ЭВМ

|  |    |
|--|----|
| <i>Ротхард Г.</i> Использование структурного программирования при подготовке программ для автоматизированных рабочих мест проектировщика, построенных на базе СМ-3 . . . . . | 44 |
| <i>Яворский О. А., Чернявский В. Н.</i> Пакет программ обработки текстовой информации . . . . .  | 50 |
| <i>Асталош И., Конц Я., Шурань А.</i> Пакет программ телеобработки данных на ЕС ЭВМ . . . . .  | 55 |
| <i>Фишер П.</i> Братиславская программная система BPS . . . . .  | 62 |
| <i>Кузнецов В. С., Куприянов В. П., Азеев А. А.</i> Прикладное программное обеспечение для СМ ЭВМ . . . . .  | 69 |
| <i>Перевалов Ю. Н., Островский В. В., Судаков С. Н.</i> Система перевода и выполнения программ СМ-2, разработанных на подмножестве языка Кобол ЕС ЭВМ . . . . .              | 75 |
| <i>Либерман М. В.</i> Один подход к проблеме повышения надежности программного обеспечения . . . . .   | 83 |

### IV. Применение средств вычислительной техники

|   |     |
|---|-----|
| <i>Луль Б., Реллер В.</i> Рабочие места для конструкторов и технологов на основе микроЭВМ СМ-1630 . . . . .   | 87  |
| <i>Забара С. С., Мильнер А. Д., Богачев А. В.</i> Проблемно-ориентированный комплекс автоматизированных рабочих мест для программирования и отладки микропроцессорных устройств АРМ2-05 . . . . . | 94  |
| <i>Пилуха И., Глоушек П., Нойманн Л., Пейчох Я.</i> Автоматизированное проектирование БИС и СБИС в ЧССР . . . . .   | 101 |

|   |     |
|---|-----|
| <i>Кацетл В.</i> Использование микропроцессоров в системах управления электрогидравлическими испытательными машинами . . . . .                                  | 108 |
| <i>Синкевич Т.</i> Микропроцессорная система вспомогательных средств проектирования . . . . .   | 115 |
| <i>Раммахер Т., Шомодьвари Г.</i> Система управления производством на базе микроЭВМ для мелких и средних промышленных кооперативов                              | 121 |
| <i>Вашиш Н.</i> Машинная система для нахождения партнеров при обмене квартирами . . . . .   | 127 |
| <i>Кашимов Э., Господинова Ю.</i> Автоматизация табельного учета с помощью микропроцессорной терминальной станции СМ-1621 . . . . .                             | 131 |
| <i>Мачке З.</i> Использование микропроцессора СМ-1630 в области лучевой терапии . . . . .   | 138 |
| <i>Христова С.</i> Использование микроЭВМ в системах бухгалтерского учета . . . . .   | 143 |
| <i>Сидоренко В. П., Руккас О. Д., Берштейн М. С., Чичирин Е. Н.</i> Применение комплекса КОДИАК при производстве СМ ЭВМ . . . . .                               | 149 |
| <i>Андерсон А. И., Глазов А. П., Кривченков А. А.</i> Особенности организации АСУ промышленного предприятия с использованием микропроцессорных систем . . . . . | 154 |

## **V. Эксплуатация и обслуживание средств вычислительной техники**

|  |     |
|--|-----|
| <i>Козлов С. Л., Петросян Э. А., Фатеев А. Е.</i> Техничко-экономическая оценка мультипрограммирования . . . . .                     | 161 |
| <i>Новицкас Ю. М., Селиванов Ю. П.</i> Система показателей экономической эффективности деятельности вычислительных центров . . . . . | 168 |

## **VI. Информация о новых средствах ЕС ЭВМ и СМ ЭВМ**

|  |     |
|--|-----|
| <i>Мача Ч.</i> Программируемая терминальная система ЕС-8534 (ТАР-34)   | 174 |
| <i>Подпорич П. П., Савин И. Д., Олейникова З. Ф.</i> Библиотека программ для научно-технических расчетов . . . . . | 177 |
| <i>Треглик Я.</i> Язык программирования ДИКОЛ . . . . .  | 180 |
| <i>Виггерт К.-Х.</i> Программное обеспечение управления данными для микроЭВМ СМ-1630 . . . . .                     | 185 |

## CONTENTS

### I. International Cooperation of the Socialist Countries in the Field of Computer Technology

|   |   |
|---|---|
| <i>Terekhov Yu. V.</i> Integrated Circuits for Computer Technology . . . .                                      | 3 |
| <i>Rakovskiy M. E., Zhukova T. V.</i> Organization of Activities in the Field of Software Development . . . . . | 8 |

### II. Computer Hardware

|  |    |
|--|----|
| <i>Lomov Yu. S.</i> Basic Directions in the Development of the High Performance General Purpose Computers Based on LSI Circuits . . . . .                            | 14 |
| <i>Fizulaev B. N., Malyarsky N. M.</i> The LSI Circuits Based on the Array Integrated Circuit Chips — Basis Circuits for the High Performance ES Computers . . . . . | 18 |
| <i>Shchetinin Yu. I.</i> Increasing of Performance and Programming Factor of the Bipolar PROM Series KR556 . . . . .   | 24 |
| <i>Znamenáček Z., Štěpánek Ya., Lakatoš I.</i> The Array LSI Cicuits for Computer Hardware . . . . .   | 31 |
| <i>Dinev R., Mukareva Kr., Petrov M.</i> System for Logic Modeling of Digital Cicuits . . . . .  | 37 |

### III. Computer Software

|   |    |
|---|----|
| <i>Rothard G.</i> Structured Programming in Program Development for a Design Working Station Based on an SM-3 Computer . . . . .  | 44 |
| <i>Yavorsky O. A., Chernyavskiy V. N.</i> A Program Package for Word Processing . . . . .   | 50 |
| <i>Asztalos L., Koncz J., Surányi A.</i> A Program Package for Data Tele processing on the ES Computers . . . . .   | 55 |
| <i>Fišer P.</i> The Bratislava Program System BPS . . . . .   | 62 |
| <i>Kuznetsov V. S., Kuprianov V. P., Azeev A. A.</i> Application Software for SM Computers . . . . .  | 69 |
| <i>Perevalov Yu. N., Ostrovskiy V. V., Sudakov S. N.</i> System of Translation and Running of Programs for the SM-2 Computers Developed with the Language Subset KOBOL ES . . . . . | 75 |
| <i>Lieberman M. V.</i> One Approach to the Problem of Software Reliability . . . . .  | 83 |

### IV. Application of Computer Facilities

|   |     |
|---|-----|
| <i>Lull B., Reller W.</i> Working Station for Designer and Production Engineer Based on the Micro Computer SM-1630 . . . . .  | 87  |
| <i>Zabarra S. S., Milner A. D., Bogachev A. V.</i> A Problem-oriented System of Working Stations for Programming and Debugging of Micro Processor Devices ARM2-05 . . . . . | 94  |
| <i>Pilucha I., Hloušek P., Neumenn L., Pejčoch J.</i> Computer Aided Design of LSI and SLSI Circuits in the CzSSR . . . . .   | 101 |
| <i>Kacel V.</i> Use of Micro Processors in Control Systems for Electro — Hydraulic Testing Machines . . . . .   | 108 |
| <i>Sinkiewicz T. A.</i> Micro Processor System of Auxilliary Design Aids . . . . .  | 115 |

|   |     |
|---|-----|
| <i>Rammacher T., Somogyváry G.</i> A System Based on Micro Computer for Small and Medium Scale Industrial Cooperatives . . . . .              | 121 |
| <i>Vass N.</i> Computer Aided Partners Retrieval System in Apartment Exchange Process . . . . .   | 127 |
| <i>Kashimov E., Gospodinova Yu.</i> Computer Aided Attendance Record Using the Multi Processor Terminal Station SM-1621 . . . . .             | 131 |
| <i>Matschke S.</i> Use of the Micro Processor SM-1630 in the Field of Radiant Therapeutics . . . . .  | 138 |
| <i>Khristova S.</i> Use of Micro Computers in Accounting Systems . . . .  | 143 |
| <i>Sidorenko V. P., Rukkas O. D., Berstein M. S., Tchitchirin E. N.</i> Application of the Complex KODIAK in SM Computer Production . . . . . | 149 |
| <i>Anderson A. I., Glazov A. P., Krivchenkov A. A.</i> Specific Features of a Micro Computer Control System of Industrial Plant . . . . .     | 154 |

## V. Operation and Maintenance of Computer Systems

|  |     |
|--|-----|
| <i>Kozlov S. L., Petrosyan E. A., Fateev A. E.</i> Technical and Economical Evaluation of Multiprogramming . . . . .               | 161 |
| <i>Novitskas Yu. M., Selivanov Yu. P.</i> The System of Indicators of Economical Efficiency of Computer Centre Operation . . . . . | 168 |

## VI. New Facilities in Unified System of Computers and Unified System of Minicomputers

|  |     |
|--|-----|
| <i>Macha Cs.</i> The Intelligent Terminal System ES-8534 (TAR-34) . . . .  | 174 |
| <i>Podporin P. P., Savin I. D., Olyenikova Z. F.</i> A Program Library for Scientific and Technical Computations . . . . . | 177 |
| <i>Trhlik Y.</i> The Programming Language DICOL . . . . .  | 180 |
| <i>Wiggert K.-H.</i> Data Management for the Micro Computer SM-1630 . .  | 185 |

## АННОТАЦИИ СТАТЕЙ

УДК 681.325.65.181.5

Терехов Ю. В. **Микроэлектронная элементная база для средств вычислительной техники.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 3—8.

Рассмотрены роль микроэлектроники в развитии вычислительной техники, а также организационные формы сотрудничества в рамках МПК по ВТ по разработке и производству микроэлектронных изделий для технических средств ЕС ЭВМ и СМ ЭВМ. Дан состав и приведены основные параметры интегральных и больших интегральных схем (ИС), включенных в «Единый перечень ИС», рекомендуемых для применения в СВТ.

УДК 681.3.06

Раковский М. Е., Жукова Т. В. **Организация работ по развитию программного обеспечения.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 8—13.

Приведены сведения об объеме парка прикладных программ социалистических стран. Даны классификация прикладного программного обеспечения и сведения о количестве пакетов прикладных программ, разработанных в рамках МПК по ВТ по группам программных средств. Сформулированы основные задачи в развитии программного обеспечения, связанные с промышленным производством программ и развитием технологии программирования. Описана деятельность Совета по применению средств вычислительной техники в области технологии программирования.

УДК 681.3:621.3.049.77

Ломов Ю. С. **Основные направления развития высокопроизводительных ЭВМ общего назначения на основе БИС.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 14—18.

Показаны составляющие компоненты, определяющие производительность ЭВМ — быстродействие элементов и структура. Рассмотрены пути повышения производительности и предельные возможности для ЭВМ общего назначения. Сформулированы основные задачи, встающие перед разработчиками ЭВМ при переходе на создание ЭВМ с использованием БИС и СБИС.

УДК 621.3.049.77:325.65

Файзулаев Б. Н., Малярский Н. М. **БИС на основе базового матричного кристалла — элементная база высокопроизводительных ЭВМ Единой системы.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 18—24.

Рассматриваются пути повышения производительности высокопроизводительных ЭВМ Единой системы и обосновывается использование матричных БИС в качестве элементной базы таких ЭВМ. Приводятся основные требования к элементной и конструктивно-технологической базе высокопроизводительных моделей ЕС ЭВМ на основе матричных БИС. Дается прогноз повышения быстродействия процессоров ЭВМ с учетом развития микроэлектронной технологии элементной базы. Ил. 3.

УДК 681.327.28

Щетинин Ю. И. **Повышение быстродействия и коэффициента программируе-**

**мости биполярных ППЗУ серии КР556.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 24—30.

Представлена схема и приведены особенности функционирования программируемых постоянных запоминающих устройств (ППЗУ) типа КР556РТ4/5. Описан характер модернизаций в схеме, направленных на повышение программируемости. Предельное значение программируемости достигает 97%, что позволяет производить программирование ППЗУ группами по 16—32 прибора после установки на печатные платы. Ил. 4.

УДК 621.3.049.77:681.3

**Зпаменачек З., Штепанек Я., Лакатош И. Матричные БИС для средств вычислительной техники.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 31—37.

Показано, что для вычислительной техники, когда нужен большой ассортимент БИС при достаточно низкой серийности их производства, особенно подходят полужаказные матричные БИС. Рассмотрены основные операции, формирующие структуру МаБИС на полупроводниковой пластине. Приведены технические параметры современных и перспективных МаБИС, а также дано описание процесса их проектирования. Описываемые МаБИС предназначены для использования в СМ ЭВМ. Ил. 3.

УДК 621.3.049.77.001.57

**Динев Р., Мукарева Кр., Петров М. Система для логического моделирования цифровых схем.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 37—43.

Описана система логического моделирования NIL, предназначенная для использования при проектировании БИС. Она позволяет моделировать БИС объема процессора, имеет удобный входной язык и достаточно универсальна, хотя и рассчитана на МОП ИС. Система имеет модульную структуру, иерархическую организацию и обладает способностью к расширению. Система работает на ЭВМ Единой системы под управлением ДОС. Ил. 4.

УДК 681.3.06

**Ротхард Г. Использование структурного программирования при подготовке программ для автоматизированных рабочих мест проектировщика, построенных на базе СМ-3.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 44—50.

Использование автоматизированного рабочего места механика АРМ-М на базе СМ-3 требует больших затрат на разработку рабочих программ. На предприятии «Роботрон» эта проблема решается с помощью структурного программирования. Излагаются методические принципы применяемого метода, описываются вспомогательные программные средства для составления структурограмм. Ил. 3.

УДК 681.3.06

**Яворский О. А., Чернявский В. Н. Пакет программ обработки текстовой информации.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 50—55.

Дано общее описание пакета ТЕКСТ для ЭВМ СМ-4. Пакет функционирует под управлением операционной системы ДИАМС-2 и имеет в своем составе программы ввода текстовой информации, редактирования и поиска текстов, вывода текстовых документов на печатающее устройство. Пакет предназначен для подготовки текстовых документов и ведения делопроизводственной документации.

УДК 681.3.06

**Асталаш И., Конц Я., Шурань А. Пакет программ телеобработки данных на ЕС ЭВМ.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 55—62.

Предложен пакет программ, содержащий средства для создания современных диалоговых систем: монитор телеобработки ШЕДОУ II, генератор программ ЦПГ и диалоговую систему разработки программ ГУТС. Пакет дает возможность организовать систему телеобработки при ограниченных ресурсах центральной ЭВМ. Описаны особенности компонентов пакета и перспективы его развития.



УДК 681.3.06

Фишер П. **Братиславская программная система BPS.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 62—69.

Описана интегрированная программная система. Система отвечает принципам структурированного программирования. Она состоит из нескольких подсистем: управляющего языка, управляющей программы и библиотеки. Программа, написанная с помощью языковых средств системы, состоит из модулей, составляющих иерархическую структуру с точным заданием интерфейса между модулями. Система функционирует на ЭВМ Единой системы под управлением ОС. Ил. 2.

УДК 681.3.06

Кузнецов В. С., Куприянов В. П., Азеев А. А. **Прикладное программное обеспечение для СМ ЭВМ.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 69—75.

Приведены сведения о системах программного обеспечения (СПО) для СМ ЭВМ — пакете программных модулей для работы с базами данных ППМ «Банк», СПО «База-СМ», СПО «База-ввод-СМ», СУБД «СЕТОР-СМ», СПО «БСП-СМ», СПО «Сорт» и СПО «ДИАМС», которые позволяют проектировать системы ввода, контроля и преобразования информации, организовывать и вести базы данных различной структуры, разрабатывать информационно-поисковые системы, обеспечивать хранение данных и т. д. Дана краткая информация о программных средствах для организации ВЦКП, связи между ЭВМ различного типа, о средствах организации сети ЭВМ, а также о средствах для автоматизации программирования и некоторых методо-ориентированных расчетов. Рассмотрены пути развития прикладного программного обеспечения СМ ЭВМ.

УДК 681.3.06

Перевалов Ю. Н., Островский В. В., Судаков С. Н. **Система перевода и выполнения программ СМ-2, разработанных на подмножестве языка Кобол ЕС ЭВМ.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 75—82.

Описана система перевода исходных программ на языке Кобол ЕС ЭВМ, написанных с ограничениями для СМ-2. Этапы трансляции исходной программы, генерации команд расширенного мнемкода СМ-2 и перекодировка выполняются на ЕС ЭВМ, а перевод с расширенного мнемкода в стандартный мнемкокод и ассемблирование — на СМ-2. Система позволяет переносить кобельные программы, написанные и отлаженные на ЕС ЭВМ, при этом производительность труда программиста возрастает в 4—5 раз, а затраты машинного времени сокращаются в 10 раз.

УДК 681.3.06.019.3

Либерман М. В. **Один подход к проблеме повышения надежности программного обеспечения.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 83—86.

Изложен подход к проблеме повышения надежности программного обеспечения, основанный на рациональном выборе языковых средств. Приведена модель характеристики надежности и рассмотрен ряд показателей этой характеристики. В рамках этого подхода предложено новое средство разработки программ — корректирующий текстовый редактор, предназначенный для сокращения затрат на разработку надежных программ.

УДК 65.015.12:681.3

Луль Б., Реллер В. **Рабочие места для конструкторов и технологов на основе микроЭВМ СМ-1630.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 87—93.

Описано типовое рабочее место на базе СМ-1630, которое в зависимости от комплектации и назначения может работать с алфавитно-цифровой и графической информацией в интерактивном режиме (АРМ). Приведены сведения по базовому программному обеспечению АРМ. АРМ имеет модульную структуру и позволяет значительно повысить производительность труда в процессах технологической и конструкторской подготовки производства. Ил. 1.

УДК 65.015.12:681.3.06

**Забара С. С., Мильнер А. Д., Богачев А. В. Проблемно-ориентированный комплекс автоматизированных рабочих мест для программирования и отладки микропроцессорных устройств АРМ2-05.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 94—101.

Комплекс АРМ2-05 служит для подготовки программ и микропрограмм для микропроцессорных устройств, который обеспечивает работу с настройкой на различные микропроцессорные наборы. Комплекс имеет открытую структуру и обеспечивает многопультную работу, причем в нем предусмотрен выход через адаптер межмашинной связи на модель ЕС ЭВМ. Использован базовый язык Микросленг, инженерная ориентация диалога, большой объем сервисных возможностей для отладки. Описана технология проектирования с помощью АРМ2-05 и приведены технические характеристики комплекса. Ил. 2.

УДК 621.3.049.77:02.57

**Пилуха И., Глоушек П., Нойманн Л., Пейчох Я. Автоматизированное проектирование БИС и СБИС в ЧССР.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 102—108.

Система автоматизированного проектирования БИС и СБИС, разрабатываемая в ЧССР, имеет модульную структуру с концентрацией нужных для проектирования данных в жестко установленном основном массиве. Описаны некоторые компоненты системы, например программа для проверки масок, подсистема моделирования и т. д. Показана этапность разработки и внедрения системы и современное состояние проекта. Ил. 1.

УДК 681.325.5.181.5:62-82

**Кацетл В. Использование микропроцессоров в системах управления электрогидравлическими испытательными машинами.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 108—115.

Описано использование микропроцессора в системе управления испытательной машины для прочностных испытаний материалов. Показана схема машины, описаны принцип действия и роль микроЭВМ в контуре управления и обработки данных эксперимента. Рассмотрены возможности автоматизированного прибора и его некоторые характеристики. Ил. 3.

УДК 681.325.5-181.5

**Синкевич Т. Микропроцессорная система вспомогательных средств проектирования.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 115—121.

Описано программно-аппаратное средство для проектировщиков технических средств и программного обеспечения, проектируемых на базе микропроцессорных комплектов серии К580 и К589. Система имеет модульную расширяемую структуру, может применяться как для проектирования, так и для испытания микропроцессорных систем. Приведены сведения о компонентах программного обеспечения и даны рекомендации по их использованию. Ил. 1.

УДК 658.5:681.3

**Раммахер Т., Шомодьвари Г. Система управления производством на базе микроЭВМ для мелких и средних промышленных кооперативов.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 121—127.

Рассматривается дешевая система на базе венгерской ЭВМ ВТ-20 для управления производством на малых предприятиях. Программы написаны на языке Паскаль. Они включают модули для отдельных задач управления. Система работает в диалоговом режиме. Описаны функции и возможности отдельных модулей программного обеспечения. Показаны возможности расширения системы при появлении более мощных ВЗУ.

УДК 681.324

**Вашш Н. Машинная система для нахождения партнеров при обмене квартирами.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 127—130.

Описана автоматизированная система поиска партнеров по обмену квартирами для трех форм обмена: одну квартиру — на другую, одну квартиру — на две, две квартиры — на одну. Входные данные формируются в три группы: об-

шие данные, данные о предлагаемых квартирах, данные о желаемых квартирах. Показаны возможности уменьшения объема обработки информации за счет введения справочных таблиц.

УДК 658.3:681.324

Кашимов Э., Господинова Ю. **Автоматизация табельного учета с помощью микропроцессорной терминальной станции СМ-1621.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 131—138.

Показано рациональное решение проблемы табельного учета с помощью скользящего рабочего времени и его автоматизированного учета. Описан вариант системы автоматизированного табельного учета на базе терминальной станции СМ-1621, с помощью которой можно будет обслужить 14 видов рабочих графиков. Охарактеризованы возможности микропроцессорного комплекса терминальной станции в зависимости от емкости оперативной памяти. Система в общей сложности может рационально работать с 32 терминалами, обслуживая в час «пик» 100—250 человек в проходной предприятия. Даны результаты эксплуатации системы и ее возможности. Ил. 4.

УДК 681.325.5-181.5:61

Мачке З. **Использование микропроцессора СМ-1630 в области лучевой терапии.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 138—142.

Описана система, разработанная на предприятии «Роботрон», состоящая из достаточно производительного центрального процессора и специальных периферийных устройств. Даны характеристики и возможности системы, в которой учитываются особенности лучевой терапии — планирование процессов облучения и ведение индивидуального режима. Ил. 2.

УДК 681.3:657

Христова С. **Использование микроЭВМ в системах бухгалтерского учета.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 143—149.

На базе микропроцессорного МОП-комплекта БИС серии 600 в НРБ разработана и внедрена система СМ-50/40-3, на базе которой построена микроЭВМ для обработки экономической информации. Описаны функции микроЭВМ по обработке файлов, программные средства на специальных входных языках и все режимы системы. Представлены некоторые из систем: автоматизации бухгалтерских операций, управления складским хозяйством, исчисления израсходованной электроэнергии. Ил. 2.

УДК 681.324

Сидоренко В. П., Руккас О. Д., Берштейн М. С., Чичирин Е. Н. **Применение комплекса КОДИАК при производстве СМ ЭВМ.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 149—154.

Комплекс КОДИАК — высокоэффективное средство автоматизации контроля и отладки логических и аналоговых блоков ЭВМ. Комплекс включает ЭВМ СМ-4 и набор специальных устройств. Аппаратные и программные средства комплекса позволяют проверять блоки СМ ЭВМ с высокой полной контролем, составлять программы проверки и вести эффективный поиск неисправностей. В статье описаны возможности и технология работы комплекса.

УДК 65.011.56:681.325.5-181.5

Андерсон А. И., Глазов А. П., Кривченков А. А. **Особенности организации АСУ промышленного предприятия с использованием микропроцессорных систем.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 155—160.

Рассмотрены принципы разработки и применения микропроцессорных систем для сбора, обработки и отображения информации на промышленном предприятии. Показана перспективность создания комплекса, в котором терминальная часть построена на микропроцессорах, а центральная — на ЕС ЭВМ, объединяющей терминалы в единый комплекс с распределенной базой данных. Предложена методика и некоторые статистические данные для оценки трудозатрат на разработку программного обеспечения микропроцессорных АСУ. Ил. 1.

УДК 681.3.06:33

Козлов С. Л., Петросян Э. А., Фатеев А. Е. **Технико-экономическая оценка мультипрограммирования.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 161—168.

Показано, что существующие методики оценки уровня мультипрограммирования недостаточно корректны. Учитывая важность данного режима для эффективного использования вычислительной установки, предлагается метод оценки уровня мультипрограммирования и коэффициента мультипрограммирования, пригодный для оптимизации этого режима. Метод позволяет определить оптимальные для задач конфигурации технических и программных средств. Ил. 1.

УДК 681.323:33

Новицкас Ю. М., Селиванов Ю. П. **Система показателей экономической эффективности деятельности вычислительных центров.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 168—173.

Показана важность правильной оценки эффективности деятельности ВЦ, анализ которой позволит выявить источники потерь и повлиять на разработку ЭВМ, организацию их обслуживания, использование фондов программ и т. д. Предложено производить оценку не единичным показателем, а системой показателей, наиболее всесторонне характеризующих деятельность ВЦ как предприятий по обработке информации. Представлены нерешенные проблемы, среди которых наиболее важной является правильное определение единицы продукции ВЦ.

УДК 681.3.06

Мача Ч. **Программируемая терминальная система ЕС-8534 (ТАР-34).** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 174—176.

Описаны возможности, особенности и состав терминальной системы ТАР-34, позволяющей вести автономную обработку данных. Приведены основные технические параметры. Ил. 1.

УДК 681.3.06

Подпорин П. П., Савин И. Д., Олейникова З. Ф. **Библиотека программ для научно-технических расчетов.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 177—180.

Описаны технологические принципы построения, состав и функциональные возможности библиотеки программ для научно-технических расчетов (БНТР). Особенностью библиотеки БНТР является простота генерации, в процессе которой обеспечивается одно из преимуществ ее — мобильность, т. е. возможность изменения состава и сферы применения.

УДК 681.3.06

Трглик Я. **Язык программирования ДИКОЛ.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика, 1984, вып. 15, с. 180—185.

Язык ДИКОЛ проблемно ориентирован на управление дискретными процессами. Он предназначен для СМ ЭВМ. Дается краткая информация о построении программы на языке ДИКОЛ, его основных операторах и об особенностях программирования задач управления.

УДК 681.3.06

Виггерт К.-Х. **Программное обеспечение управления данными для микроЭВМ СМ-1630.** — В кн.: Вычисл. техника соц. стран. — М.: Финансы и статистика. 1984, вып. 15, с. 185—193.

Предложены три пакета программ для управления файлами, доступа к записям и управления банками данных с трехуровневой архитектурой. Язык системы управления данными легко осваивается специалистами в области программирования. Показаны особенности и преимущества описанной системы управления базами данных. Ил. 4.

## ABSTRACTS

### **Terekhov Yu. V. Integrated Circuits for Computer Technology.**

The role of integrated circuits in computer technology development as well as organizational forms of cooperation in the frame of the Intergovernmental Commission of the Socialist Countries in the Field of Computer Technology in development and production of micro electronics for ES and SM computer hardware is considered. List of basic specifications of IC and LSI circuits included in «The Unified List of the Integrated Circuits Recommended for Use in Computer Hardware» are given.

### **Rakovsky M. E., Zhukova T. V. Organization of Activities in the Field of Software Development.**

Information on the volume of application programs developed in the socialist countries is given. Classification of application software as well as information on application programs developed in the frame of the Intergovernmental Commission of the Socialist Countries in the Field of Computer Technology divided by groups is suggested. The basic tasks in software development related to the industrial production of programs as well as tasks in development of programming technology are formulated. The activities of the Council for Application of Electronic Computer Technology in the field of programming technology are described.

### **Lomov Yu. S. Basic Directions in the Development of the High Performance General Purpose Computers Based on LSI Circuits.**

Components which define computer performance — component speed and computer architecture are shown. Methods of performance increasing and their limits for general purpose computers are considered. Basic tasks for computer designers in use of LSI and SLSI circuits in computer design are formulated.

### **Fizulaev B. N., Malyarsky N. M. The LSI Circuits Based on the Array Integrated Circuit Chips — Basic Circuits for the High Performance ES Computers.**

Directions of performance increasing of high speed ES computers are considered. The use of the array LSI circuits as the component base of such computers is grounded. The basic requirements to the component design and technology for high speed ES computers based on array LSI circuits are given. Forecast for increasing of central processors performance taking into consideration progress of micro electronics technology of components is suggested.

### **Shchetinin Yu. I. Increasing of Performance and Programming Factor of the Bipolar PROM Series KR556.**

The structure and functioning features of programmable read only memories of the type KR 556 RT 4/5 are described. Characteristics of the structure updating used for higher programming are shown. The upper limits of programming are up to 97 per cent. This permits to program PROM by groups of 16 to 32 devices after their mounting on printed circuit board.

### **Znamenáček Z., Štěpánek Ya., Lakatoš I. The Array LSI Circuits for Computer Hardware.**

It is shown that for computer hardware, when a broad set of LSI circuits is produced in a rather lower production series, half-custom array LSI circuits particularly fit. The basic operations forming the LSI circuit on a semiconductor wafer are considered. Specifications of the modern and future array LSI circuits are given, as well as a design process is described. The described array LSI circuits are intended for use within the SM computers.

### **Dinev R., Mukareva Kr., Petrov M. System for Logic Modeling of Digital Circuits.**

System of logic modeling intended for use in LSI circuits design is described. It permits to model processor LSI circuits, it has convenient input language and is sufficiently universal, although is designed for MOS integrated circuits. The system has module structure, hierarchical organization, and can be extended. The system is functioning under the OS ES.

**Rothard G. Structured Programming in Program Development for a Design Working Station Based on SM-3 Computer.**

Use of the design working station ARM—M based on SM mini-computer requires high expenditures for working programs. At the enterprise ROBOTRON this problem is solved through programming. The methodology of the used method is given. Utility program products for composing of structuregrams are described.

**Yavorsky O. A., Chernyvasky V. N. A Program Package for Word Processing.**

General description of the package TEXT for SM-4 mini-computers is given. The package is functioning under the operating system DIAMS-2 and comprises programs for text editing and retrieval, and for output of text documentation to a printer. The package is intended for preparing of text documentation and for business correspondence.

**Asztalos I., Konec J., Surányi A. A Program Package for Data Teleprocessing on the ES Computers.**

A program package for designing of modern interactive systems is suggested. The package comprises the teleprocessing monitor SHADOW II, the program generator CPG, and the GUTS interactive system for program designing. The package suggests possibility to complex a teleprocessing system using the limited resources of a host computer. Specifications of package components and prospects of its development are described.

**Fišer P. The Bratislava Program System BPS.**

An integrated program system is described. The principles of structured programming are used in the system. The system comprises several subsystems: a control language, a control program, and a library. The program written by means of system language aids comprises modules composing hierarchical structure with precise setting of an interface between modules. The system is functioning under the ES operating system.

**Kuznetsov V. S., Kuprianov V. P., Azeev A. A. Application Software for SM Computers.**

Information on software systems (SS) for the SM computers—package of programming modules (PPM) for use with the data bases PPM «BANK», SS «BAZA-SM», SS «BAZA-VVOD-SM», DBMS «SETOR-SM», SS «BSP-SM», SS «SORT», and SS «DIAMS» is given. These products are used in designing of systems for data input, data control and conversion, in organization and running of various structured data bases, in designing of information storage and retrieval systems, in managing of data storage etc. Short information on programming products for organizing of multi users computer centres, for communication of various types computers, on hardware and software products for designing of computer network, as well as on aids for automation of programming and some method-oriented calculations is given. Directions of application software development for the SM computers are considered.

**Perevalov Yu. N., Ostrovsky V. V., Sudakov S. N. A System of Translation and Running of Programs for the SM-2 Computers Developed with the Language Subset COBOL ES.**

A system of translation of source programs in COBOL ES language written with limits for the SM-2 computers is described. Stages of source program translation, generation of extended mnemonic SM-2 and recording are performed on ES computer, and translation from extended mnemonic into standard mnemonic and assemblage—on SM-2. The system permits to transfer COBOL programs, written and debugged on ES computer. Ah that programmers productivity is raised up to 4 to 5 times, and computer time requirements are reduced 10 times.

**Liberman M. V. One Approach to the Problem of Software Reliability.**

An approach to the problem of raising software reliability based on rational selection of language aids is given. A model of reliability characteristics is given. Some characteristics variables are considered. Using this approach a new program development instrument is suggested — a correcting text editor intended for reduction of expenses in development of robust programs.

**Lull B., Reller W. Working Station for Designer and Production Engineer Based on the Micro Computer SM-1630.**

A standard working station based on the micro-computer SM-1630 is described. Depending on configuration, the working station can use alphanumeric and graphic information in interactive mode. Information on the working station basic software is given. The working station has module structure and permits to increase labor productivity sufficiently in process of tooling up for its production.

**Zabarra S. S., Milner A. D., Bogachev A. V. A Problem-oriented System of Working Stations for Programming and Debugging of Micro Processor Devices ARM2-05.**

The system ARM2-05 is intended for program and microprogram development for microprocessor devices and provides setting up for various microprocessor sets. The system has open structure and provides multiterminal activity. The system has an output to ES computer through a computer-to-computer adapter. The basic language Microsleng, technically oriented dialog, large volume of service aids for debugging are used. Design process using the working station ARM2-05 is described. The system specifications are given.

**Pilucha I., Hloušek P., Neumenn I., Pejčoch J. Computer Aided Design of LSI and SLSI Circuits in the CzSSR.**

A computer aided design system for LSI and SLSI circuits, developed in the CzSSR, has module structure with concentration of data required for design in an exactly established basic file. Some system components are described, for example, a program of mask check, a modeling subsystem etc. Stages of system development and its implementation and up-to-date state of the art of the project are shown.

**Kacetl V. Use of Micro Processors in Control Systems for Electro-Hydraulic Testing Machines.**

Use of a micro processor in a control system of testing machine for strength tests of materials is described. Structure of the machine is shown; principle of operation and role of a micro computer in a control loop, and processing of experiment data are described. Possibilities of the device are considered and some of its specifications are given.

**Sinkiewicz T. A Micro Processor System of Auxilliary Design Aids.**

Firmware for designers of hardware and software based on micro processor kits series K580 and K589 is described. The system has module extended structure and can be used for design as well as test of micro processor systems. Information on software components is given. Recommendations on their implementation are suggested.

**Ramacher T., Somogyváry G. A System Based on Micro Computer for Small and Medium Scale Industrial Cooperatives.**

A not expensive system based on a Hungarian computer VT-20 for production process control on small scale enterprises is considered. Programs are written on the Pascal language; they comprise modules for certain control tasks. The system operates in interactive mode. Functions and possibilities of certain software products are described. Possibilities of extending the system with more powerful external storages are shown.

**Vass N. Computer Aided Partners Retrieval System in Apartment Exchange Process.**

A computer aided system of partners retrieval in apartment exchange using three alternatives: one apartment to change for another one, one apartment to change for two ones, two apartments to change for one is described. Input information is formed into three groups: general data, data on suggested apartments, data on required apartments. Possibilities of lower volumes of data processing due to implementation of reference tables are shown.

**Kashimov E., Gospodinova Yu. Computer Aided Attendance Record Using the Multi Processor Terminal Station SM-1621.**

Rational solution of the attendance record problem using sliding working time and its automated account is shown. The computer aided system based on the

terminal station SM-1621 is described. Information on 14 types of attendance charts could be processed by using this terminal station. Specifications of a micro processor complex of the terminal station depending on volume of main memory are given. The system could use up to 32 terminals serving at the peak hour from 100 to 200 workers passing by. System operation results and its possibilities are given.

**Matschke S. Use of the Micro Processor SM-1630 in the Field of Radiant Therapeutics.**

A system developed at the enterprise «Robotron» comprising a central processing unit and specific peripherals is described. System specifications and possibilities are given. The system takes into consideration special features of radiant therapeutics—planning of radiation processing and management of individual features.

**Khristova S. Use of Micro Computers in Accounting Systems.**

The system SM 50/40-3 designed and implemented in Bulgaria is based on the micro processor MOS LSI series 600. A business micro computer is based on the system SM 50/40-3. Functions of the micro computer in file processing, program aids on specific input languages, and all system modes are described. Some systems are represented: automation of accounting operations; warehouse management; calculation of electric power consumption.

**Sidorenko V. P., Rukkas O. D., Berstein M. S., Tchitchirin E. N. Application of the Complex KODIAK in SM Computer Production.**

The complex KODIAK is a highly effective instrument of control and debugging of logical and analog computer modules. The complex comprises an SM-4 computer and a set of specialised devices. Complex hardware and software are used to check SM computer modules at high level, to compose checking programs, and to manage error search. Possibilities and operation process of the complex are described.

**Anderson A. I., Glazov A. P., Krivchenkov A. A. Specific Features of a Micro Computer Control System of Industrial Plant.**

Development and application principles of micro computer system for acquisition, processing, and display of data at an industrial plant is considered. Prospects of complex making, terminal part of which is built on micro processors and the central part is built on ES computer are shown. The central part integrates terminals into a unified complex with a distributed data base. Methods and some statistical data for evaluations of working hours for software development of a micro computer control system are suggested.

**Kozlov S. L., Petrosyan E. A., Fateev A. E. Technical and Economical Evaluation of Multiprogramming.**

It is shown that, the existing methods of evaluation of multiprogramming level are not adequately correct. Taking into consideration significance of this mode for effective computer application, the method of evaluation of multiprogramming level and multiprogramming ratio, suitable for this mode optimization, is suggested. This method is used for definition of optimum hardware and software configurations for various tasks.

**Novitskas Yu. M., Selivanov Yu. P. The System of Indicators of Economical Efficiency of Computer Centre Operation.**

Importance of the correct evaluation of a computer centre operation effectiveness is shown. Analysis of this effectiveness will permit to reveal the sources of waste and affect computer development, organization of its service, use of program libraries etc. It is suggested to make evaluations not by using individual indicators but by using a system of indicators, which thoroughly describe operation of a computer centre, as an information processing organization. Some unsolved problems are shown. Most important among them is correct definition of computer centre productivity unit.

**Macha Cs. The Intelligent Terminal System ES-8534 (TAP-34).**

Possibilities, specific features and structure of the terminal system TAP-34, which permits to provide off-line data processing are described.



**Podporin P. P., Savin I. D., Olyenikova Z. F. A Program Library for Scientific and Technical Computations.**

Engineering principles of structure and functional features of a program library for scientific and technical computations are described. The specific feature of the program library is simplicity of generation in the process of which it provides one of its advantages mobility, i. e. possibility of changes of structure and application sphere.

**Trhlik Y. The Programming Language DICOL.**

The language DICOL is a problem — oriented language for discret process control. It is intended for SM mini computers. Short information on program structure of DICOL language, its main operators, and specific features of programming control tasks is given.

**Wiggert K.-H. Data Management for the Micro Computer SM-1630.**

Three program packages for file control, access to records, and control of data banks with three level architecture are suggested. The language of a data control system is easily used by non specialists in programming. Specific features and advantages of the described data base control system are shown.

**Вычислительная техника социалистических стран**

**Сборник статей**

**Выпуск 15**

Науч. редактор *Ю. П. Селиванов*  
Зав. редакцией *И. Г. Дмитриева*  
Редактор *Л. А. Табакова*. Мл. редактор *Л. В. Речицкая*  
Худож. редактор *М. К. Гуров*,  
Техн. редакторы *Л. Г. Чельшева* и *И. В. Завгородняя*  
Корректоры *Т. М. Васильева* и *М. А. Синяговская*  
Обложка художника *Л. С. Эрмана*

ИБ № 1495

---

|                         |                              |                   |                                |
|-------------------------|------------------------------|-------------------|--------------------------------|
| Сдано в набор 10.01.84. | Подписано в печать 13.04.84. | A00995.           | Формат 60×90 <sup>1/16</sup> . |
| Бум. тип. № 1.          | Гарнитура «Литературная».    | Печать высокая.   | Усл. печ. л. 13.               |
| Усл. кр.-отт. 13,375.   | Уч.-изд. л. 14,82.           | Тираж 10 000 экз. | Заказ 77.                      |

Цена 1 руб.

---

Издательство «Финансы и статистика», 101000, Москва, ул. Чернышевского, 7.

Московская типография № 8 Союзполиграфпрома  
при Государственном комитете СССР  
по делам издательств, полиграфии и книжной торговли,  
101898, Москва, Центр, Хохловский пер., 7.

1 руб.

ФИНАНСЫ И СТАТИСТИКА