ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ

УДК 681.324

С. А. Зинкин

РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ УПРАВЛЕНИЯ ВНЕШНЕЙ ПАМЯТЬЮ ЭВМ И СЕТЕЙ С РАСШИРЕННЫМИ ФУНКЦИОНАЛЬНЫМИ ВОЗМОЖНОСТЯМИ

Рассмотрены вопросы проектирования интеллектуальных систем управления внешней памятью (СУВП) на основе функционального подхода, учитывающего в первую очередь состав и взаимосвязь функций системы, а также информационно-управляющие потоки, циркулирующие в СУВП. Наряду с диаграммами потоков данных для описания потоков управления, устанавливающих порядок выполнения функциональных модулей, используются сценарии с темпоральными операциями.

Введение

Функциональные подходы к проектированию информационно-вычислительных систем, основанные на выделении функций, реализуемых системой, и их взаимосвязей, в настоящее время находят широкое применение. В качестве примера можно привести ряд известных в литературе подходов: функционально-структурный подход, предполагающий формирование дерева функций системы и построение обобщенных операторных моделей; подход, основанный на построении информационно-потоковых схем параллельных многопроцессорных систем с общим управлением; подход, основанный на иерархической декомпозиции функций системы и покрытии их унифицированными программно-аппаратными модулями. Существуют и другие способы функционального описания систем, например, с помощью функциональных и операционных схем, диаграмм потоков данных.

Использование указанных функциональных подходов при проектировании интеллектуальных СУВП в силу их специфики может вызвать определенные трудности. Следует отметить прежде всего невычислительный характер процессов, протекающих в СУВП, последовательное выполнение операций, некритичность времени выполнения некоторых операций и связанная с этим возможность программной реализации функций, необходимость согласования скоростей передачи интерфейсов ввода-вывода центральной ЭВМ и внешних запоминающих устройств (ВЗУ). Кроме того, следует отметить отсутствие гибких процедур перехода от функционального описания к структуре аппаратно-программных средств системы. Для представления функциональной структуры СУВП будем использовать диаграммы потоков данных (ДПД) специального вида. Достоинством ДПД является возможность совмещения разных уровней абстракции при описании элементов функциональной

структуры как в плане их физической реализации, так и в плане взаимодействия элементов, например, возможность представления как байтового, так и файлового обмена информацией. По существу, ДПД являются определенной семантической интерпретацией графовой модели СУВП, в которой вершинам графа соответствуют функциональные модули (ФМ), представляющие собой определенным образом графически оформленные блоки, имеющие множество входов и выходов. Недостатком ранее использованных ДПД является отсутствие управляющих связей между ФМ, что не позволяет судить, хотя бы в общем виде, о динамике протекающих в СУВП процессов.

1 Парадигма согласования и координации процессов и объектов и ее использование при структурной реализации многофункциональных СУВП

При моделировании реальной технической системы цепочка рассуждений в сети абстрактных машин (СеАМ) [1, 2] позволяет получить заключение о состоянии процесса. Например, при моделировании системы ВЗУ может быть получено заключение о завершении позиционирования головок записи-считывания или о завершении операции ввода-вывода в целом.

Выполнение модуля СеАМ интерпретируется некоторым агентом или группой агентов. Выполнение правил-продукций модулей СеАМ и связанное с ним обновление фактов (функций и предикатов в некотором пространстве) могут влиять на систему и окружающую ее среду. При использовании модулей СеАМ в качестве непосредственно выполняемых спецификаций в реальной системе обработки и хранения данных некоторым фактам соответствуют специальные агенты-демоны, вызываемые путем модификации определенных функций СеАМ. При интерпретации СеАМ в реализованной на ЭВМ или на базе вычислительной сети поведенческой модели системы демоны могут выполнять роль различного рода сервисных процедур для сбора статистических данных о функционировании реализуемой системы, для управления списками текущих и планируемых событий, для реализации параллельного или квазипараллельного режима исполнения процессов в модели и др. Правила, представленные модулями СеАМ, объединяются в единую систему посредством глобального пространства функций и предикатов.

Таким образом, формализм СеАМ возможно использовать в качестве языка инженерии знаний для основанного на правилах условных замещений функций и предикатов иерархического, управляемого событиями дискретного концептуального моделирования.

Исходя из анализа требований к современным СУВП, сформируем дерево функций, используя для этого индексную и точечную нотацию.

Дерево функций содержит следующие уровни и подуровни:

I уровень:

F_0 – управление ВЗУ.

II уровень:

F_1 – организация сопряжения с интерпретатором СеАМ;

F_2 – организация сопряжения с ВЗУ;

F_3 – ведение и поддержка каталогов;

F_4 – инструкции интерпретатора и запросы оператора;

F_5 – контроль и диагностика;

F_6 – управление и обмен данными с интерпретатором;

F_7 – управление и обмен данными с ВЗУ.

Проведем декомпозицию выделенных базовых функций СУВП и определим нижний уровень иерархии.

III уровень:

– функция F_1:

$F_{1.1}$ – обмен данными в параллельной форме с интерпретатором;

$F_{1.2}$ – управление обменом информацией с интерпретатором;

$F_{1.3}$ – хранение служебной информации;

– функция F_2:

$F_{2.1}$ – обмен данными в параллельной форме ВЗУ;

$F_{2.2}$ – управление обменом информацией ВЗУ;

$F_{2.3}$ – хранение служебной информации;

– функция F_3:

$F_{3.1}$ – анализ команды СУВП;

$F_{3.2}$ – включение файлов в каталог;

$F_{3.3}$ – исключение файла из каталога;

$F_{3.4}$ – поиск файла в каталоге;

$F_{3.5}$ – чтение и передача каталога;

$F_{3.6}$ – формирование результатов выполнения команды СУВП;

– функция F_4:

$F_{4.1}$ – анализ инструкций интерпретатора СеАМ;

$F_{4.2}$ – трансляция инструкций интерпретатора и запросов оператора в команды СУВП;

$F_{4.3}$ – управление реконфигурацией ВЗУ;

$F_{4.4}$ – управление электропитанием;

$F_{4.5}$ – планирование и диспетчеризация команд СУВП;

$F_{4.6}$ – анализ результатов выполнения команды СУВП;

$F_{4.7}$ – формирование информации о состоянии СУВП;

$F_{4.8}$ – сбор и анализ трасс-обращений к файлам;

– функция F_5:

$F_{5.1}$ – анализ команды СУВП;

$F_{5.2}$ – обработка прерываний по ошибкам аппаратуры;

$F_{5.3}$ – обработка прерываний по ошибкам программного обеспечения;

$F_{5.4}$ – повторный запуск команд СУВП;

$F_{5.5}$ – формирование результатов выполнения команды СУВП;

– функция F_6:

$F_{6.1}$ – анализ команды СУВП;

$F_{6.2}$ – прием данных из интерпретатора СеАМ;

$F_{6.3}$ – передача данных в интерпретатор;

$F_{6.4}$ – формирование результатов выполнения команды СУВП;

$F_{6.5}$ – формирование служебной информации;

$F_{6.6}$ – управление аппаратными ресурсами;

$F_{6.7}$ – оформление инструкций интерпретатора СеАМ;

– функция F_7:

$F_{7.1}$ – анализ команды СУВП;

$F_{7.2}$ – трансляция команды СУВП в инструкции интерфейса ВЗУ;

$F_{7.3}$ – планирование и диспетчеризация инструкций интерфейса внешних ЗУ;

$F_{7.4}$ – прием данных от ВЗУ;
$F_{7.5}$ – передача данных в ВЗУ;
$F_{7.6}$ – формирование результатов выполнения команды СУВП;
$F_{7.7}$ – формирование служебной информации;
$F_{7.8}$ – управление аппаратными ресурсами.

2 Диаграммные макроопределения и сценарии управления внешней памятью ЭВМ

Общая структура интеллектуальной СУВП представлена на рис. 1. В работе [3] для описания функционирования СУВП использовались диаграммные макроопределения (ДМО). В настоящей работе мы будем рассматривать иерархию ДМО как разновидность иерархических сценарных моделей, предложенных в работах [4–6].

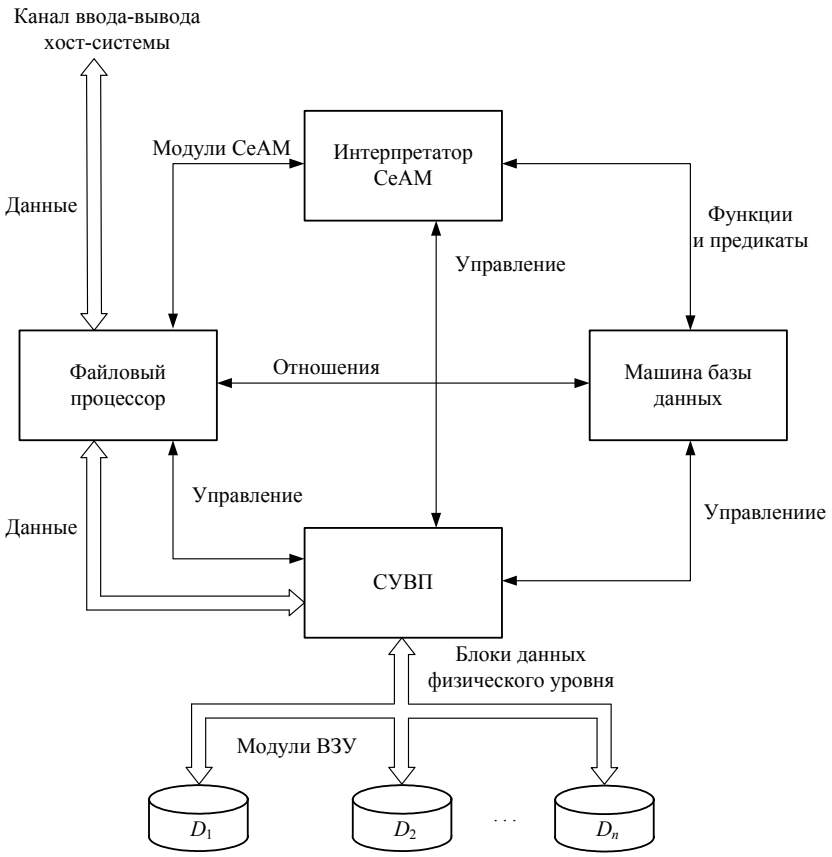


Рис. 1 Общая структурная схема интеллектуальной СУВП

На рис. 2 представлено ДМО высшего уровня дерева функций СУВП. При составлении ДМО высшего уровня сделаны следующие предположения: обмен информацией между интерпретатором СеАМ и ВЗУ осуществляется с использованием согласующей буферной памяти (БП), поэтому отсутствует связь по данным между ФМ F_4, F_6 и F_7, в то время как информация, необходимая для нормальной работы ФМ, передается в виде параметров команд

СУВП или инструкций интерпретатора СеАМ; режим чтения каталога в собственную память интерпретатора предусматривает предварительную передачу информации в БП. Звездочками на рисунках обозначены результаты выполнения команд.

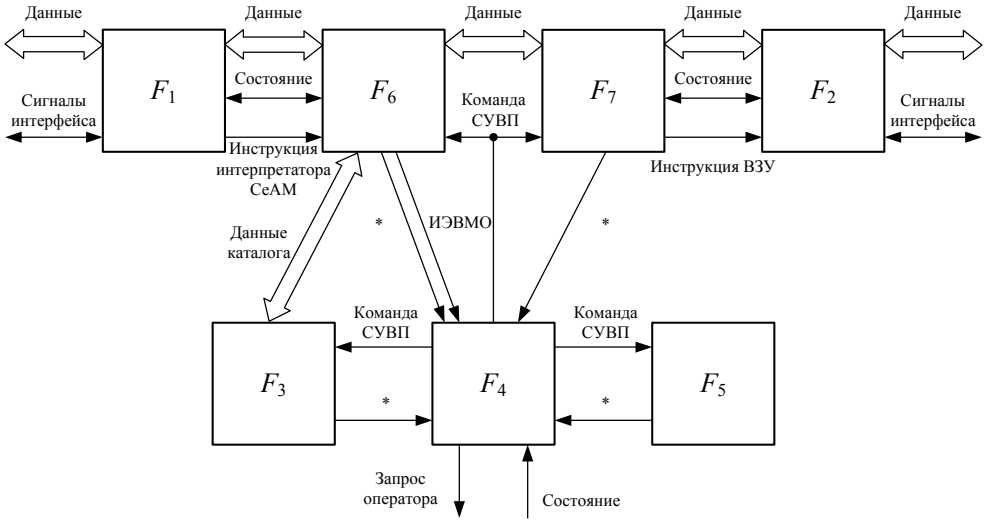


Рис. 2 Диаграммное макроопределение F_0 (управление ВЗУ)

Выполнение инструкции интерпретатора разбивается на три фазы: начальную фазу выполнения инструкции и фазу окончания инструкции. Начальная фаза заключается в формировании команды СУВП в соответствии с полученной инструкцией. Представим процесс выполнения инструкции в виде цепочки последовательно реализуемых действий, где каждому действию соответствует выполнение определенной операции отдельным или совокупностью ФМ; описание и реализация темпоральных операций ↑ (непосредственное следование) и $\|^{\cdot}$ (перекрытие временных интервалов).

Обозначим через ($\text{ФМ}_i \|^{\cdot} \text{ФМ}_j$) действие, при выполнении которого i-й и j-й ФМ работают одновременно и зависимо друг от друга, причем порядок следования модулей определяет направление передачи информации. Тогда цепочка действий, соответствующих начальной фазе, имеет вид

$$L_1 = (F_1 \|^{\cdot} F_6) \uparrow F_4,$$

где фрагмент $(F_1 \|^{\cdot} F_6)$ описывает операцию приема и форматирования инструкции интерпретатора СеАМ. Фаза окончания инструкции заключается в передаче интерпретатором информации о состоянии СУВП и представлена следующей цепочкой действий:

$$L_2 = F_4 \uparrow (F_6 \|^{\cdot} F_1) \uparrow F'_4,$$

в которой возврат к модулю F_4 означает выдачу результата выполнения команды СУВП модулем F_6 после завершения обмена данными с интерпретатором. Так как символ F_4 встречается второй раз, в цепочке он отмечен одним штрихом.

При каждом новом вхождении имени функционального модуля в одну и ту же цепочку будем добавлять к нему новый штрих. Ниже в общем виде

представлены цепочки действий, соответствующие фазам выполнения основных инструкций интерпретатора.

1. Чтение каталога:

$$L_3 = F_4 \uparrow (F_3 \|^{\cdot} F_6) \uparrow F_4' \uparrow (F_6 \|^{\cdot} F_1) \uparrow F_4'',$$

где фрагмент $(F_3 \|^{\cdot} F_6)$ описывает операцию предварительной передачи каталога в БП.

2. Поиск по каталогу:

$$L_4 = F_4 \uparrow F_3 \uparrow F_4'.$$

3. Чтение данных из ВЗУ:

$$L_5 = F_4 \uparrow (F_7 \|^{\cdot} F_2) \uparrow F_4' \uparrow (F_2 \|^{\cdot} F_7) \uparrow F_4'' \uparrow (F_6 \|^{\cdot} F_1) \uparrow F_4''',$$

где фрагмент $(F_7 \|^{\cdot} F_2)$ описывает операцию трансляции команды СУВП в инструкцию интерфейса ВЗУ и передачу управляющей информации во внешнее ЗУ; $(F_2 \|^{\cdot} F_7)$ описывает операцию загрузки БП данными, поступающими из ВЗУ.

4. Запись данных в ЗУ:

$$L_6 = F_4 \uparrow (F_1 \|^{\cdot} F_6) \uparrow F_4' \uparrow (F_7 \|^{\cdot} F_2) \uparrow F_4'' \uparrow (F_7 \|^{\cdot} F_2)^{*} \uparrow F_4''',$$

где фрагмент $(F_1 \|^{\cdot} F_6)$ описывает операцию загрузки БП данными, поступающими из собственной памяти интерпретатора, а фрагмент $(F_7 \|^{\cdot} F_2)^{*}$ описывает операцию передачи данных из БП в ВЗУ.

На рис. 3 и 4 представлены ДМО, соответствующие модулям F_6 и F_7.

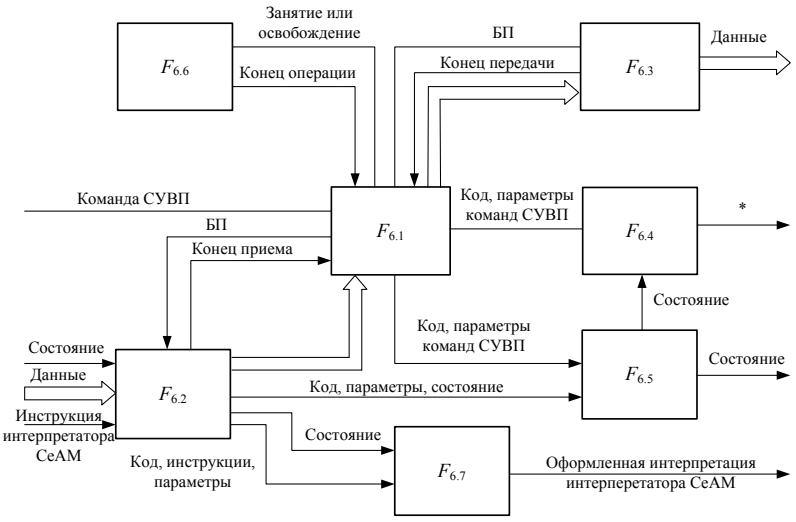


Рис. 3 Диаграммное макроопределение F_6
(управление и обмен данными с интерпретатором СеАМ)

Определим процесс физической реализации функциональной структуры СУВП как процесс построения физической структуры в заданном конструктивном базисе с учетом заданных функциональных требований. В конструктивном плане такой процесс сводится к решению задачи покрытия функциональной структуры определенным базовым набором физических структур.

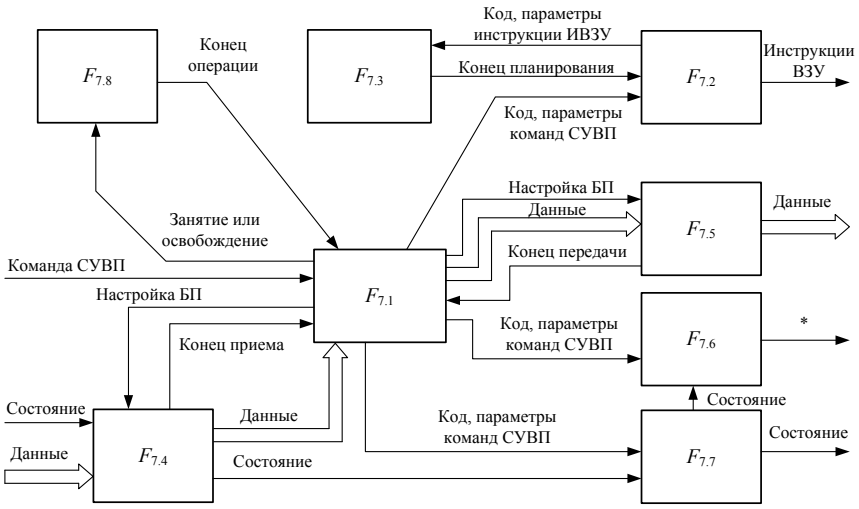


Рис. 4 Диаграммное макроопределение F_7 (управление и обмен данными с ВЗУ)

Для перехода к физической структуре СУВП необходимо выделить функционально полное множество физических элементов, из которых формируется структура, и множество связей между ними. Множество типов аппаратных структур задается следующим образом: P – процессорный блок; M – блок памяти; I – интерфейсное оборудование. Осуществим переход от функциональной структуры СУВП к физической с учетом использования многопортовой БП, для которой возможно параллельное выполнение операций обмена информацией с центральной ЭВМ в режиме прямого доступа к памяти и ВЗУ. На рис. 5 представлена физическая структура реализации модулей, составляющих к уровню F_0 (управление ВЗУ), построенная при учете темпоральных операций в сценариях, описанных цепочками L_1–L_6, и условий перехода от темпоральных сценариев к пространственным.

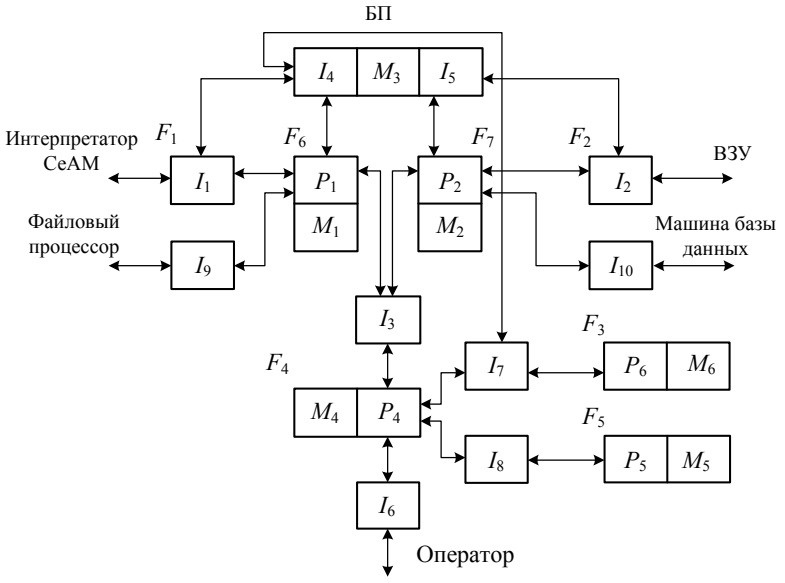


Рис. 5 Физическая реализация интеллектуальной системы управления ВЗУ

Область истинности (отношение развертывания) предиката D для данной структуры формируется модулем СеАМ, выполняющим следующие правила обновлений:

$M_{suvp} = [\alpha_{start}](\{D(F_1, I_1) \leftarrow \text{true}, D(F_2, I_2) \leftarrow \text{true}, D(F_3, P_6) \leftarrow \text{true},$
$D(F_4, P_4) \leftarrow \text{true}, D(F_5, P_5) \leftarrow \text{true}, D(F_6, P_1) \leftarrow \text{true},$
$D(F_7, P_2) \leftarrow \text{true}\} \vee R^E),$

по завершении выполнения которых функциональные модули размещаются на конструктивных модулях интеллектуальной СУВП. Здесь α_{start} – признак готовности системы к загрузке. Выполнение данного модуля СеАМ в реальной системе сопровождается загрузкой соответствующего программного обеспечения в основную память процессорных модулей, составляющих СУВП. Рисунок 6 иллюстрирует построение СУВП на базе типовой мультимикропроцессорной системы с общедоступной мультипроцессорной шиной. Здесь и далее аббревиатура *LM* обозначает локальную основную память, а *IP* – выделенный интерфейсный модуль. Пунктиром обозначен процессорный элемент P_0 с локальной памятью, включение которого в структуру необходимо при централизованном управлении работой СУВП; при децентрализованном управлении необходимость в выделенном процессоре отсутствует. На рис. 7 приведена структура СУВП на основе коммутатора, реализованная в инфраструктуре локальной сети.

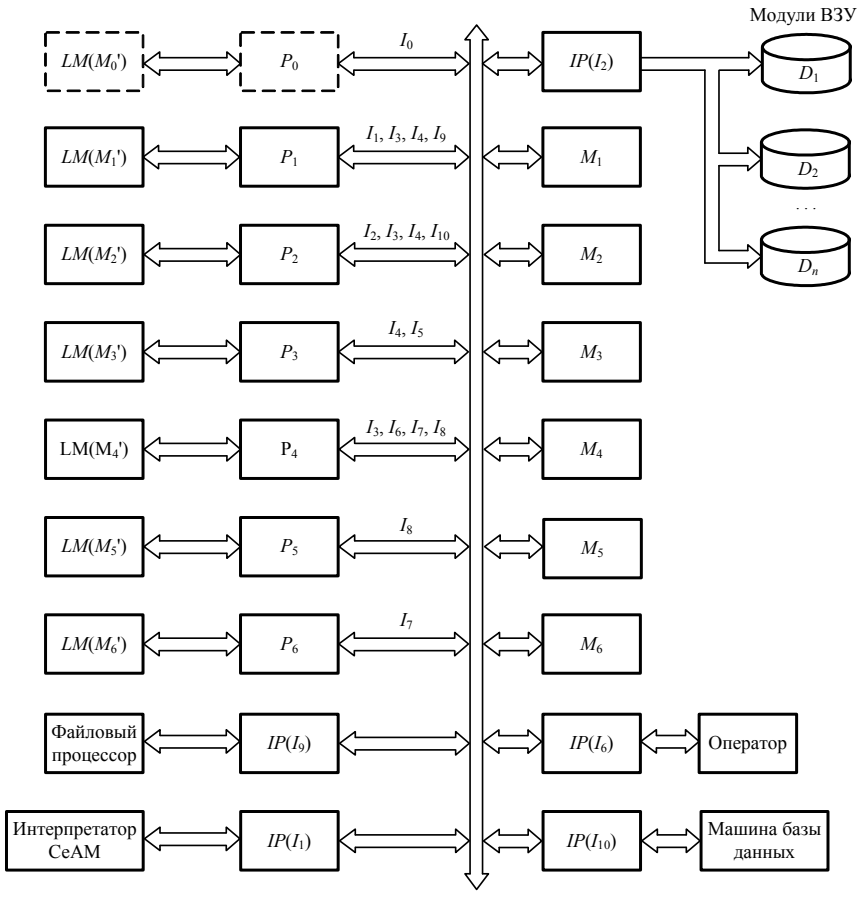


Рис. 6 СУВП на базе мультимикропроцессорной системы

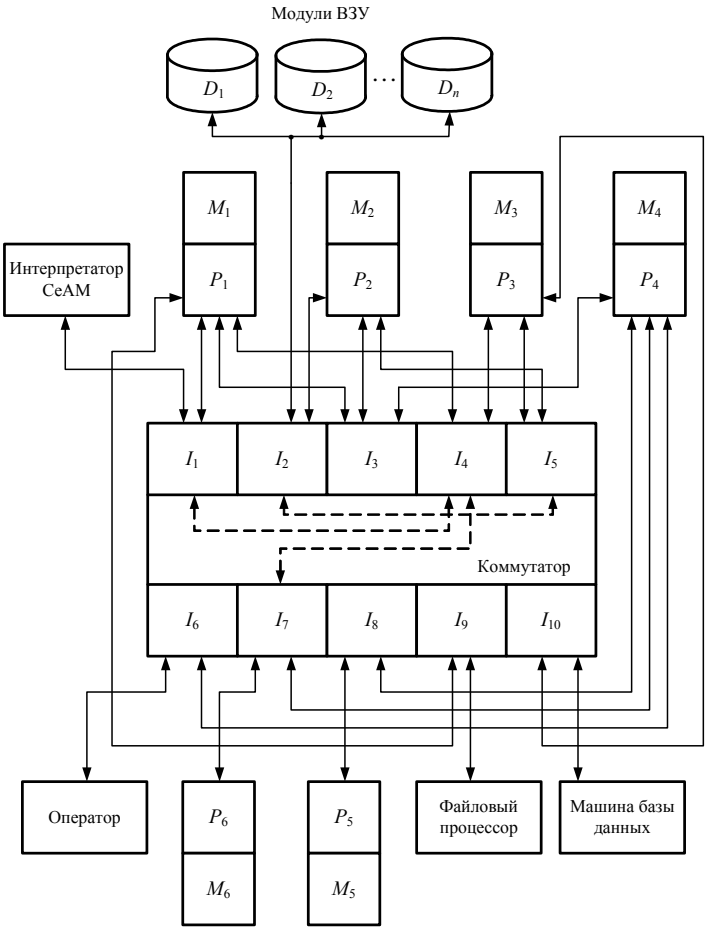


Рис. 7 Структура мультимикропроцессорной системы управления ВЗУ на основе коммутатора

3 Функционально-структурная реализация машин баз данных и знаний в составе интеллектуальных систем управления внешней памятью ЭВМ и сетей

В рамках исследований в области технической имитации интеллекта большое значение придается разработке систем управления базами знаний (БЗ) как одного из важнейших инструментальных средств новой информационной технологии. К основным функциям управления БЗ относятся [7]: поддержка пользовательского интерфейса; пополнение и редактирование БЗ; логический вывод; поддержка целостности и непротиворечивости знаний; реорганизация БЗ. Отметим, что системы управления БЗ (СУБЗ) могут быть встроены в информационно-управляющую систему, например в систему управления внешней памятью ЭВМ, с целью ее интеллектуализации. БЗ и соответственно СУБЗ можно классифицировать по ряду признаков, основным из которых является модель представления знаний. В настоящее время используются основные модели представления знаний на основе логики предикатов, систем продукций, семантических сетей, фреймов. В данной работе в качестве модели представления знаний используются сети абстрактных ма-

шин (СеАМ). В зависимости от выбранной модели существенно различается не только представление, но и алгоритмы обработки знаний, а также аппаратные средства их реализации.

Любую операцию отдельного устройства СУВП или его блока можно описать определенной последовательностью действий, тип и характер реализации которых задается аппаратно-программной структурой СУВП. Структура программного обеспечения СУВП определяется с помощью отображения полученных алгоритмов на многоуровневую древовидную структуру физической среды (ФС). Рассмотрим физическую реализацию машины базы данных (МБД) на основе ее дерева функций:

I уровень:

F_0 – управление реляционной базой данных.

II уровень:

F_1 – организация сопряжения с интерпретатором СеАМ;

F_2 – управление и обмен данными с интерпретатором;

F_3 – организация сопряжения с файловым процессором (ФП);

F_4 – управление и обмен данными с ФП;

F_5 – ведение и поддержка справочника отношений базы данных; управление буфером данных;

F_6 – интерпретация запросов;

F_7 – контроль и диагностика;

F_8 – реализация операций реляционной алгебры и агрегатных функций.

III уровень:

– функция F_6:

$F_{6.1}$ – анализ инструкций интерпретатора СеАМ;

$F_{6.2}$ – трансляция инструкций интерпретатора СеАМ в последовательность команд МБД;

$F_{6.3}$ – поддержка мультипрограммного режима работы МБД;

$F_{6.4}$ – планирование и диспетчеризация команд МБД;

$F_{6.5}$ – анализ результатов выполнения команд;

$F_{6.6}$ – формирование информации о состоянии МБД, изменение схемы базы данных;

$F_{6.7}$ – сбор и анализ трасс-обращений к базе данных.

На рис. 8 приведено диаграммное макроопределение DM_0, представляющее функцию F_0. Здесь данные 1 – программа реализации инструкции интерпретатора СеАМ; данные 2 – справочник отношений в базе данных; данные 3 – результаты выполнения отдельных реляционных операций и отношения базы данных. Звездочками отмечены результаты выполнения инструкций МБД.

На рис. 9 приведена аппаратная структура МБД в предположении использования многопортовой буферной памяти, причем появление дополнительных блоков интерфейсного оборудования объясняется характером взаимодействия процессорных блоков.

Область истинности (отношение развертывания) предиката D для данной структуры формируется модулем СеАМ, выполняющим следующие правила обновлений:

$M_{\text{mbd}} = [\alpha_{\text{start}}](\{D(F_1, I_1) \leftarrow \text{true}, D(F_2, P_1) \leftarrow \text{true}, D(F_3, I_2) \leftarrow \text{true},$
$D(F_4, P_2) \leftarrow \text{true}, D(F_5, P_5) \leftarrow \text{true}, D(F_6, P_4) \leftarrow \text{true},$
$D(F_7, P_4) \leftarrow \text{true}, D(F_8, P_6) \leftarrow \text{true}\} \vee R^E),$

по завершении выполнения которых функциональные модули размещаются на конструктивных модулях МБД. Здесь, как и ранее, α_{start} – признак готовности системы к загрузке. Выполнение данного модуля СеАМ в реальной системе сопровождается загрузкой соответствующего программного обеспечения в основную память процессорных модулей, составляющих МБД.

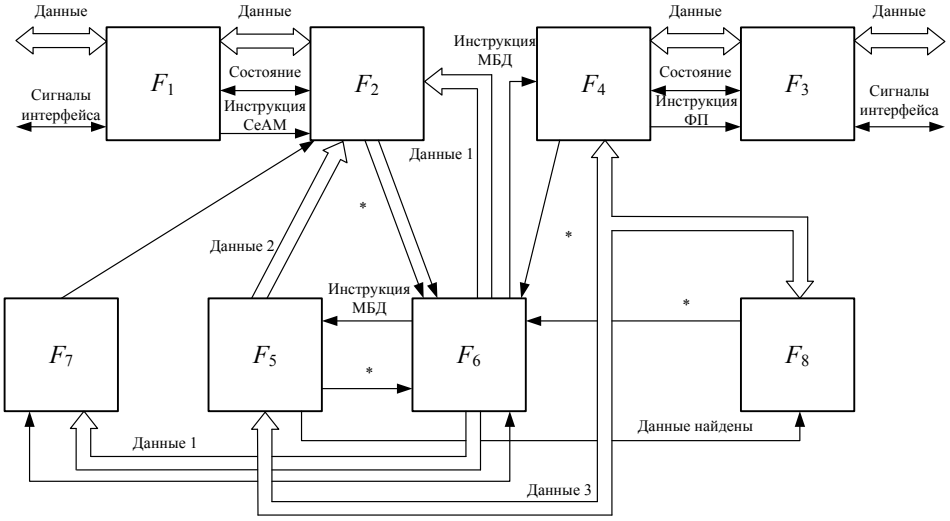


Рис. 8 Диаграммное макроопределение F_0

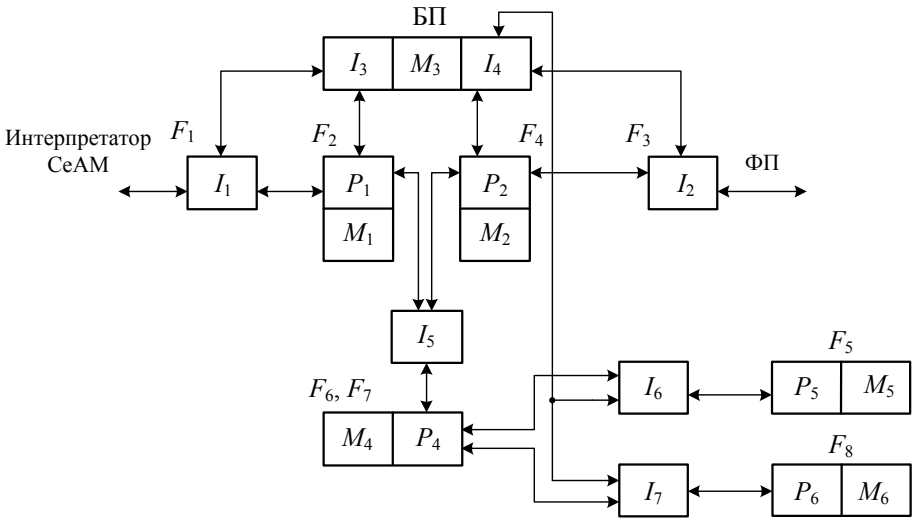


Рис. 9 Аппаратная структура МБД с использованием буферной памяти

Рисунок 10 иллюстрирует построение МБД на базе типовой мультимикропроцессорной системы с общедоступной мультипроцессорной шиной, а на рис. 11 приведена структура МБД на основе коммутатора, реализуемая в среде локальной сети.

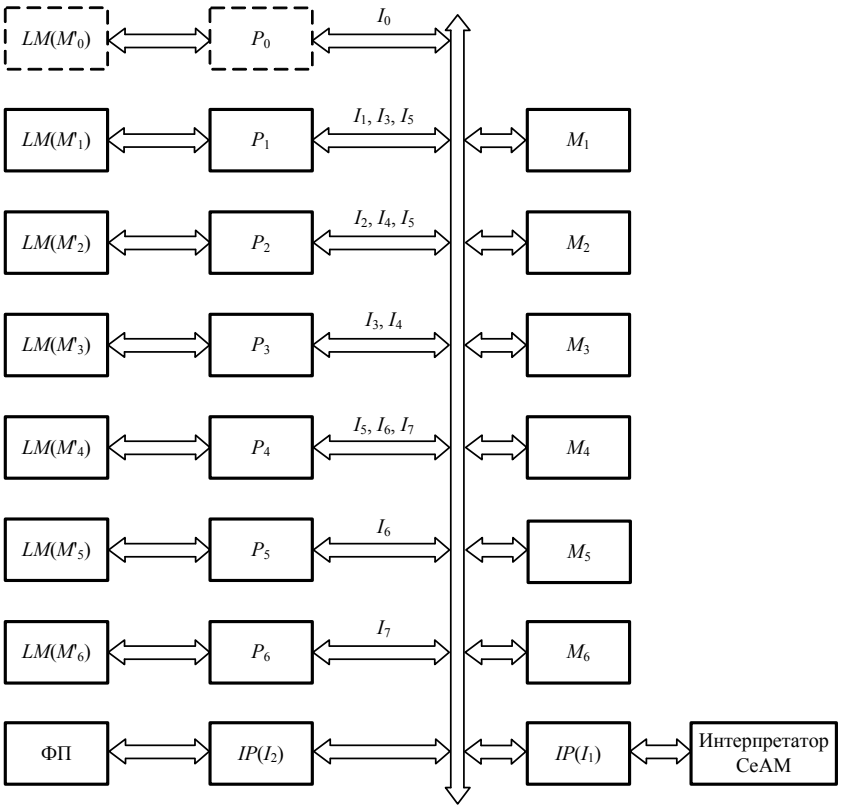


Рис. 10 СУВП на базе типовой мультимикропроцессорной системы

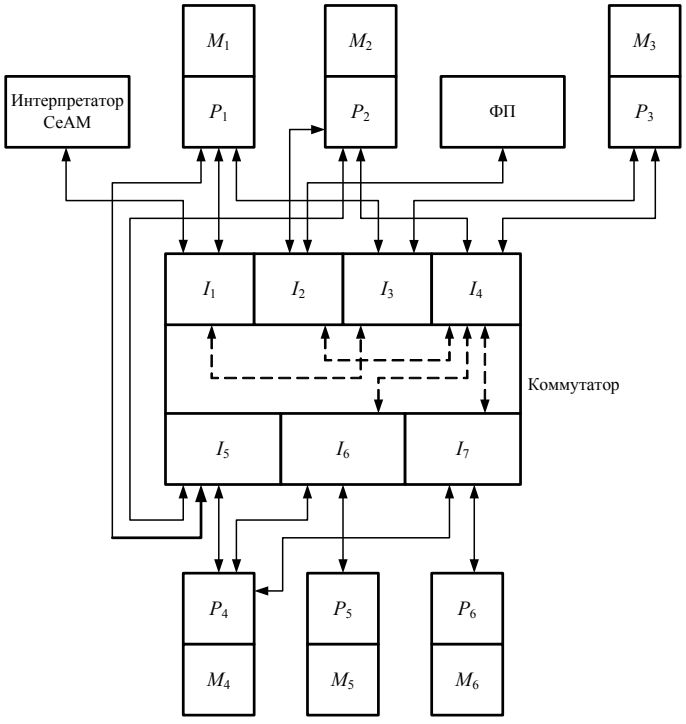


Рис. 11 Структура мультимикропроцессорной МБД на основе коммутатора

Заключение

Предложенные методы предназначены для построения аппаратно-программных средств для реализации механизмов логического вывода, управления базой знаний, обеспечения интеллектуального интерфейса с пользователем. В отличие от работы [8], предложенная здесь реализация предполагает сетевую реализацию и развитое самоуправление в иерархической памяти. Кроме того, предполагается, что предложенные методы, основанные на механизмах исполняемых спецификаций сетей абстрактных машин и активных семантических сетей, предложенных в работах автора [1–6], позволят сделать шаг к преодолению семантического разрыва между инфраструктурой классической мультипроцессорной системы и алгоритмическими структурами, обеспечивающими внутреннюю и внешнюю интеллектуализацию СУВП.

Список литературы

1. **Зинкин, С. А.** Сети абстрактных машин высших порядков в проектировании систем и сетей хранения и обработки данных (базовый формализм и его расширения) / С. А. Зинкин // Известия высших учебных заведений. Поволжский регион. Технические науки. – 2007. – № 3. – С. 13–22.
2. **Зинкин, С. А.** Сети абстрактных машин высших порядков в проектировании систем и сетей хранения и обработки данных (механизмы интерпретации и варианты использования) / С. А. Зинкин // Известия высших учебных заведений. Поволжский регион. Технические науки. – 2007. – № 4. – С. 37–50.
3. **Зверев, С. Л.** Интеллектуализация систем внешней и массовой памяти ЭВМ / С. Л. Зверев, С. А. Зинкин, Б. М. Раков // Вопросы радиоэлектроники. – 1990. – Вып. 13. – С. 43–58. – (Электронная вычислительная техника).
4. **Зинкин, С. А.** Алгебра сценариев для спецификации операционной семантики активных сетей хранения и обработки данных / С. А. Зинкин // Известия высших учебных заведений. Поволжский регион. Технические науки. – 2004. – № 2. – С. 96–107.
5. **Зинкин, С. А.** Самомодифицируемые сценарные модели функционирования систем и сетей хранения и обработки данных (базовый формализм и темпоральные операции) / С. А. Зинкин // Известия высших учебных заведений. Поволжский регион. Технические науки. – 2007. – № 1. – С. 3–12.
6. **Зинкин, С. А.** Самомодифицируемые сценарные модели функционирования систем и сетей хранения и обработки данных (реализация и свойства сценарных моделей) / С. А. Зинкин // Известия высших учебных заведений. Поволжский регион. Технические науки. – 2007. – № 2. – С. 13–21.
7. Представление и использование знаний / под ред. Х. Уэно, М. Исидзука. – М. : Мир, 1989. – 220 с.
8. **Мото-ока, Т.** Японский проект вычислительных систем пятого поколения / Т. Мото-ока // Высокоскоростные вычисления. Архитектура, производительность, прикладные алгоритмы и программы суперЭВМ / под ред. Я. Ковалика. – М. : Радио и связь. – 1988. – С. 90–105.