

О ПРОГРАММНОЙ РЕАЛИЗАЦИИ ЭТАПА СОЗДАНИЯ МОДЕЛИ ДЛЯ СИСТЕМЫ САИС

Введение

Основное назначение системы автоматизации исследования автоматных сетей (САИС) является организация моделирования и проведение отдельного эксперимента или серии экспериментов над автоматными сетями [2] на ЭВМ.

Процесс моделирования эксперимента на ЭВМ весьма трудоемкая работа для исследователя. Поэтому возникает необходимость автоматизировать труд исследователя при создании модели эксперимента. В связи с этим в заказах пользователя системы САИС предусмотрен отдельный этап работы — этап создания модели эксперимента. Для осуществления этого этапа система САИС снабжена соответствующими программными и языковыми средствами.

Программная реализация этапа создания модели эксперимента является компонентой программного обеспечения системы САИС и называется СОЗМОД-ом.

Данная статья посвящена вопросам программной реализации компоненты СОЗМОД.

Назначение и состав компоненты СОЗМОД

СОЗМОД включает в себя следующие программные модули:

- управляющая программа;
- программа для создания простых структур;
- программа для создания сложных структур;
- программа для создания модели (эксперимента);
- программа связи с пользователем;
- программа обработки исключительных состояний.

Состав программных модулей компоненты СОЗМОД изображена на рис. 1.

Программы написаны на языке АССЕМБЛЕР и работают под управлением операционной системы ОС ЕС. Общий объем программ СОЗМОД-а около 4000 машинных команд. Загрузка программ СОЗМОД-а начинается с динамического вызова управляющей программы, осуществляемой главной управляющей программой системы САИС.

Компонента СОЗМОД предназначена для выполнения следующих функций:

- создание простой структуры;
- создание сложной структуры;
- создание модели.

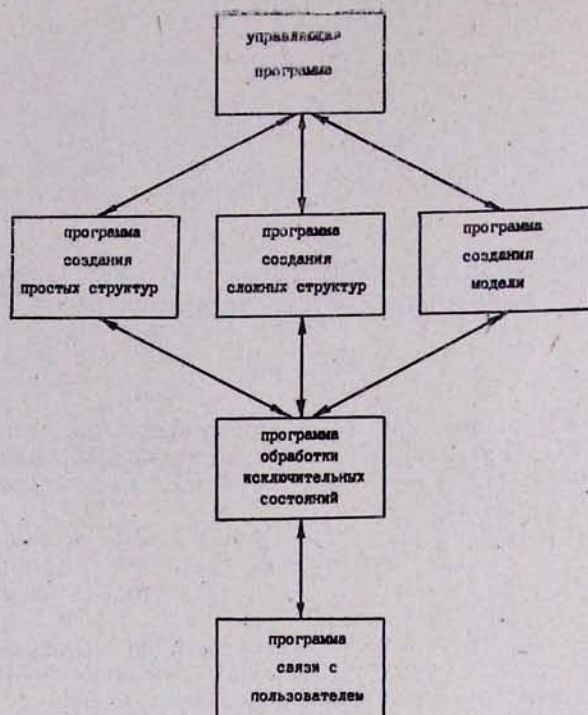


Рис. 1.

Входные и выходные данные

Входные данные СОЗМОД-а находятся в последовательном наборе данных операторов (ПНДОП) и в последовательном наборе данных «данные» (ПНДАН), которые создаются на предыдущих этапах работы системы САИС.

Из первого набора данных (ПНДОП) в управляющую программу поступают операторы СОЗМОД-а. СОЗМОД имеет три типа операторов:

- а) оператор «СТРУКТ», который указывает, что нужно создать простую структуру;
- б) оператор «СОБСТР», который указывает, что нужно создать сложную структуру;
- в) оператор «МОДЕЛЬ», который указывает, что нужно создать модель.

В операторах находятся имена создаваемой структуры или модели и некоторая информация, указывающая, например, что создаваемая структура (модель) должна записываться в постоянный или временный библиотечный набор данных.

Во втором наборе данных (ПНДАН) находятся те данные, которые необходимы для создания данной структуры или модели.

Для создания структур среди этих данных указываются:

- а) имена первой и второй структуры, которые составляют новую структуру;

- б) имя вершины, которая становится сложной;
- в) таблицы отождествления параметров этой вершины с параметрами вершин второй структуры и т. д.

Для создания модели среди данных указываются:

- а) имя структуры, над которой строится модель;
- б) первоначальные значения параметров некоторых вершин;
- в) список свободных вершин в структуре и имена соответствующих им программ или моделей и т. д.

Выходные данные компоненты СОЗМОД являются простая структура, сложная структура и модель.

Простая структура состоит из двух частей: СВ (содержащей сведения о вершинах) и совокупности полуопределенных коммутационных таблиц (ПКТ).

Каждая ПКТ определяет связи параметров данной вершины с параметрами других вершин данной структуры. Ей присваивается системное имя, и оно в виде отдельного раздела записывается в соответствующем библиотечном наборе данных (БНДПКТ).

Записи ПКТ заблокированы и имеют фиксированную длину. ПКТ состоит из двух частей. Первая часть ПКТ называется вводным. Ее первая запись содержит некоторую информацию о данной ПКТ (например, счетчик и т. д.).

Остальные записи имеют следующий формат:

поле 1	поле 2	поле 3
1	9	13 14

где:

- поле 1—содержит имя вводного параметра данной вершины;
- поле 2—содержит относительный адрес в области связи [2] выводного параметра вершины, с которой данный параметр связан;
- поле 3—длина вводного параметра данной вершины.

Вторая часть ПКТ называется выводным, и записи имеют тот же формат, что и записи вводной части, но следующего содержания:

- поле 1—содержит имя выводного параметра данной вершины;
- поле 2—содержит относительный адрес в области связи выводного параметра данной вершины;
- поле 3—длина выводного параметра данной вершины.

СВ простой структуры представляет из себя список всех вершин данной структуры с указанием для каждой вершины имени соответствующей ПКТ. СВ получает имя создаваемой простой структуры и в виде отдельного раздела записывается в библиотечном наборе данных структур.

Записи СВ простой структуры заблокированы и имеют фиксированную длину. СВ состоит из информационной части и части, описывающей вершины.

В разделе сначала расположена информационная часть, которая содержит имя простой структуры, размер области связи данной простой структуры, количество записей информационной части и т. д.

Вторая часть СВ простой структуры состоит из записей, имеющих следующий формат:

поле 1	поле 2	поле 3
1	9	17 28

где:

поле 1—содержит имя вершины простой структуры;
 поле 2—содержит системное имя ПКТ данной вершины.
 Поле 3 в простой структуре не содержит информацию. Оно резервировано для занесения необходимой информации при создании сложной структуры и модели (см. ниже).

Вершины простой структуры будем называть свободными.
 Из двух структур (простой или сложной) получаем новую сложную структуру, когда в некоторой свободной вершине первой структуры помещаем вторую.

СВ сложной структуры создается следующим образом. В библиотечном наборе данных структур в разделе с именем создаваемой сложной структуры сначала записывается СВ первой структуры, а потом СВ второй структуры. Относительный адрес СВ второй структуры помещается в запись свободной вершины первой структуры, в которую помещается вторая структура. Вершина, в которой помещена структура, называется сложной.

Запись сложной вершины в СВ имеет следующий формат:

поле 1	поле 2	поле 3	поле 4
1	9	17	25 28

где:

поле 1—содержит имя сложной вершины;
 поле 2—содержит системное имя ПКТ данной вершины;
 поле 3—содержит системное имя ОКТ данной вершины;
 поле 4—содержит относительный адрес СВ помещаемой структуры в разделе СВ созданной структуры.

Для сложной вершины создается определенная коммутационная таблица (ОКТ), которая в виде отдельного раздела записывается в соответствующем библиотечном наборе данных (БНДОКТ). Ей присваивается системное имя, и это имя тоже помещается в запись данной сложной вершины. ОКТ определяет закон соединения параметров данной вершины с параметрами структуры, помещаемой в эту вершину.

Каждая ОКТ создается на основе ПКТ данной сложной вершины и совокупности ПКТ тех вершин помещаемой структуры, параметры которых соединены с параметрами сложной вершины. Записи ОКТ по содержанию подобны записям ПКТ данной сложной вершины, но вместо имени параметра вставлен адрес отождествляемого параметра соответствующей вершины из соединяемой структуры.

Таким образом, сложная структура представляется в виде СВ и совокупности ПКТ и ОКТ (только для сложных вершин).

Модель создается на основе соответствующей структуры (простой или сложной), когда во всех оставшихся свободных вершинах этой структуры помещаются соответствующие программные модули (модели автоматов в ЭВМ [1]) или заранее построенные модели. При этом, для каждой такой вершины создается своя ОКТ.

Таким образом, в модели все вершины имеют свои ОКТ, которые должны использоваться для коммутации параметров на этапе «проведение эксперимента».

Краткое описание алгоритма

Компонента СОЗМОД работает следующим образом. В управляющую программу поступает очередной оператор из последовательного

набора данных операторов (ПНДОП). Потом управляющей программой определяется тип оператора. Если это оператор «СТРУКТ», то управляющей программой динамически вызывается и управление передается программе создания простой структуры. Этой программой в соответствии с данными из последовательного набора данных ПНДАН создается простая структура, которая записывается в соответствующих библиотечных наборах данных.

Подобным образом вызываются и функционируют программы создания сложной структуры и создания модели в случае поступления в управляющую программу соответственно операторов «СОБСТР» или «МОДЕЛЬ».

Все три упоминаемые программы, кроме своих основных функций «способны» обнаружить возникающие исключительные состояния, при которых динамически вызывается и управление передается программе обработки исключительных состояний.

После устранения исключительных состояний (в диалоговом или в автономном режиме), управление возвращается вызывающему модулю. После завершения работы любой рабочей программы, управление возвращается управляющей программе. После этого в управляющую программу поступает следующий оператор и начинается его обработка. Если это не оператор СОЗМОД-а, а оператор завершения работы этапа, то управление возвращается главной управляющей программе системы САИС.

Некоторые особенности

В системе САИС предусмотрены несколько библиотечных наборов данных для постоянного и временного хранения структур и моделей. По усмотрению пользователя структура или модель может храниться постоянно или в течении данного заказа пользователя. Это способствует рациональному использованию внешней памяти и машинного времени, затраченного на сжатие после удаления структур или моделей из библиотечных наборов данных.

Как уже было сказано, все ПКТ и ОКТ данной структуры или модели хранятся отдельно, в виде разделов в соответствующих библиотечных наборах данных. Это обусловлено следующими причинами. Если СВ и все ПКТ и ОКТ структуры записывались бы вместе в одном разделе библиотечного набора данных, предназначенный для структур, то создаваемая сложная структура из заранее созданных двух структур включала бы копии всех ПКТ и ОКТ этих двух структур.

При попытке с помощью этой структуры создать новую структуру, вышеупомянутые ПКТ и ОКТ записывались бы еще раз и т. д., что разумеется не целесообразно. Поэтому все ПКТ и ОКТ данной структуры хранятся отдельно в соответствующих библиотечных наборах данных.

Такая организация применяется и при создании модели, поскольку модель строится на основе некоторой структуры (создание модели начинается с перезаписи данной структуры в соответствующий раздел создаваемой модели).

Таким образом, хранение всех ПКТ и ОКТ данной структуры или модели способствует экономному использованию памяти (и внешней, и внутренней).

Здесь, естественно, возникает следующий вопрос: невозможно ли при создании СВ сложной структуры после перезаписи СВ первой структуры не записывать СВ второй структуры, а взамен этого, в записи соответственной сложной вершины СВ первой структуры записать, скажем, имя ссылаемой структуры? Ведь это тоже способствовать, скажем, имя ссылаемой структуры? Оказывается, что такая ло бы экономному использованию памяти. Оказывается, что такая организация в рамках применяемой формализации невозможно (при создании модели аналогичный вопрос решается положительно).

В системе САИС предусмотрена сервисная программа для удаления структур и моделей из соответствующих библиотечных наборов данных, а также сервисная программа для сжатия этих наборов данных. Но в связи с вышеупоминаемой спецификой организации структур и моделей, их удаление из библиотечных наборов данных может привести к следующему усложнению. При удалении данной структуры, СВ этой структуры можно удалять без повреждений других структур, что нельзя сказать по отношению к ПКТ и ОКТ этой структуры. Удаление всех ПКТ и ОКТ этой структуры привело бы к повреждениям всех других структур, в составе которых они участвуют. С другой стороны, при удалении структуры могут возникнуть такие ПКТ и ОКТ, которые не участвуют ни в какой другой структуре и их хранение не целесообразно. Заметим, что СВ модели с этой точки зрения проявляет себя аналогично ПКТ и ОКТ.

Чтобы обойти данную трудность, каждая ПКТ или ОКТ имеет счетчик, указывающий количество использования данного ПКТ или ОКТ во всех структурах и моделях, которые имеются в постоянных библиотечных наборах данных. Когда удаляется из библиотек некоторая структура или модель, где используется данная ПКТ или ОКТ, то из счетчика этого ПКТ или ОКТ вычитается единица. А когда в библиотеках добавляется новая структура или модель, то счетчикам ПКТ и ОКТ всех вершин тех структур или моделей, которые являются компонентами данной структуры или модели, добавляется единица. Из соответствующих библиотечных наборов данных удаляются разделы только тех СВ модели, ПКТ и ОКТ, счетчики которых приравняются нулю.

Для надежной работы в компоненте СОЗМОД предусмотрены программные средства, которые обнаруживают и обрабатывают некоторые исключительные состояния. Эти исключительные состояния могут появляться при создании модели эксперимента (переполнение в справочниках используемых библиотечных наборов данных, невозможность резервирования оперативной памяти требуемой размерности, исчерпаны системные имена для ПКТ и ОКТ и т. д.), которые разделяются на две группы:

- а) исключительные состояния первого рода;
- б) исключительные состояния второго рода.

Исключительные состояния первого рода появляются в основном в результате неправильного составления заказа пользователем. Примером такого исключительного состояния может стать то обстоятельство, когда при создании сложной структуры в библиотечных наборах данных отсутствует затребованная структура.

Исключительные состояния второго рода—это те состояния, которые появляются во время работы СОЗМОД-а при той или иной ситуации в условиях правильно составленного заказа. Примером такого исключительного состояния может стать: а) невозможность резервирования оперативной памяти необходимой размерности; б) исчерпаны системные имена для ПКТ или ОКТ и т. д.

Во всех рабочих модулях реализована возможность обнаружения возникающих исключительных состояний. Для устранения исключительных состояний предусмотрена отдельная программа, называемая программой обработки исключительных состояний. Она в зависимости от исключительных состояний их может устранять или в условиях вмешательства пользователя (в основном исключительные состояния первого рода) или без его вмешательства.

Таким образом, программа обработки исключительных состояний соответственно работает как в диалоговом, так и в автономном режимах. В диалоговом режиме с помощью программы связи с пользователем в программу обработки исключительных состояний поступают соответствующие указания от пользователя о выборе способа обработки (например, пропускается данный оператор, заказ и т. д.). Имеются некоторые исключительные состояния второго рода, которые также можно устранять в диалоговом режиме. Например, когда поступает сообщение о том, что невозможно резервировать оперативную память заранее фиксированной размерности, пользователь может соответствующим указанием уменьшать этот размер. А в основном исключительные состояния второго рода устраняются в автономном режиме. При автономном режиме способ обработки выбирается однозначно заранее фиксированным образом.

Ս. Մ. ՄՈՒՐԱԴՅԱՆ

„САИС“ ՀԱՄԱԿԱՐԳՈՒՄ ՄՈՒԵԼԻ ՍՏԵՂԾՄԱՆ ՓՈԼԻ ԾՐԱԳՐԱՅԻՆ ԻՐԱՅՄԱՆ ՄԱՍԻՆ

Ա. մ փ ռ փ ռ լ մ

Հոգիածում բերվում է էՀՄ-ի վրա փորձի մոդելի ստեղծման փուլի ծրագրային իրացման ընդհանուր նկարագրությունը, որը ավտոմատային ցանցերի ավտոմատացված հետազոտության համար նախատեսված «САИС» համակարգի ծրագրային ապահովման մի բաղկացուցիչ մասն է: Տրված է համակարգի բաղաչին լիզվի տարրերը, որոնք անհրաժեշտ են փորձի մոդելի ստեղծման համար:

ЛИТЕРАТУРА

- 1 Галстян Л. А. Об организации обработки программных моделей предметной области в системе САИС. В настоящем сборнике.
- 2 Мурадян С. М. Об организации проведения эксперимента над моделями в системе САИС. I Республиканская конференция «Перспективы применения микропрограммирования при проектировании ЭВМ и систем (Ереван, декабрь 1986). Тезисы докладов, ЕрНИИММ, Ереван, 1986, с. 34—35.