



ОБЪЕДИНЕННЫЙ  
ИНСТИТУТ  
ЯДЕРНЫХ  
ИССЛЕДОВАНИЙ  
ДУБНА

P10-85-416

Г.Балука, И.М.Саламатин

РЕАЛИЗАЦИЯ МЕТОДА ОПИСАНИЯ СТРУКТУРЫ  
ОБОРУДОВАНИЯ КАМАК  
ПРИ МОДУЛЬНОЙ ОРГАНИЗАЦИИ ПРОГРАММ  
НА ЯЗЫКЕ ПАСКАЛЬ

Направлено на XIX школу по АНИ  
/Новосибирск, 1985/

1985

1. Для ЭВМ типа "Электроника-60", ДВК, СМ-3, СМ-1300, СМ-4 и др., преимущественно используемых в задачах автоматизации научных исследований/1/, в настоящее время имеются базовые средства для создания систем автоматизации экспериментов /САЭ/: штатные операционные системы включают удобные языки программирования с развитыми структурами данных - Паскаль, СИ, Модула-2 и др.; промышленность выпускает совместимые контроллеры крейтов КАМАК /например, К-16, для ЭВМ СМ-3, типа 106А, 106Б и др./; расширяется ассортимент функциональных блоков КАМАК. Между тем сроки создания САЭ еще велики и составляют 1-3 года/1/. Существенного сокращения сроков создания программного обеспечения /ПО/ САЭ можно ожидать при переходе к программированию ПО САЭ на языках высокого уровня, обеспечении инвариантности программ для КАМАК относительно изменений способа размещения блоков КАМАК в крейтах, способа распайки поля коммутации контроллера крейта и др., а также при модульной организации ПО КАМАК.

В данной работе описываются разработанные средства, обеспечивающие эти свойства при программировании указанного оборудования на языке Паскаль в ОС RT-11.

2. В работе<sup>/2/</sup> предложен новый метод программирования оборудования КАМАК - метод описания структуры оборудования /ОО/. Этот метод заключается в том, что регистры оборудования обозначаются идентификаторами, состав регистров и способ доступа к ним /т.е. структура оборудования/ описываются в виде структуры данных, эта структура отображается на адресное пространство, выделенное оборудованию, после чего идентификаторы регистров могут использоваться в арифметических и логических выражениях.

Для использования данного метода на языке Паскаль не требуются дополнительные разработки. Однако в ПО САЭ могут быть выделены компоненты, необходимые во многих системах автоматизации. Реализация таких компонентов в общем виде, единообразное их использование на основе модульной организации ПО КАМАК, разработка программных модулей, комплексируемых без изменения в различных САЭ позволяют сократить сроки и затраты на создание ПО САЭ. С этой целью разработаны описываемые ниже две группы программ. Одна из них предназначена для использования в проблемно-ориентированной модульной системе программирования ДКС<sup>/3/</sup>, вторая - для иного способа комплексирования компонентов в систему /например, с помощью штатного редактора связей/ и для отладочных программ. Описываемый набор программ обладает

определенной полнотой и взаимной самосогласованностью и может быть применен также в других системах программирования САЭ, использующих метод ОСО для программирования оборудования КАМАК.

3. Модульная организация ПО КАМАК обслуживается программой динамического распределения памяти /ДРП/ и загрузчиком перемещаемых программ, подробно описанными в работе<sup>/4/</sup>. Программа ДРП объединяется с управляющей программой САЭ редактором связей. Этой программе выделяется ресурс оперативной памяти ~10К слов /поле ДРП/ для размещения программных модулей /ПМ/. Реализованы операции:

LM - поиск программного модуля на поле ДРП /или загрузка модуля в случае его отсутствия/;

FIX - загрузка и фиксация ПМ на поле ДРП;

UNFIX - отмена фиксации;

ST - передача параметров и управления /запуск/ ПМ и др.

Структура ПМ принята идентичной структуре процедур на языке Паскаль. Такие процедуры независимо транслируются, обрабатываются редактором связей и в перемещаемом формате загрузки заносятся в формате .REL в специальную библиотеку<sup>/5/</sup> или штатную файловую систему. Обращение к ПМ, декларированному инструкцией

PROCEDURE MODNAM (DPI:TYPE1;...; DPN:TYPE2),

реализуется парой инструкций:

LM(MODNAM);

ST(API,...,APN);

где API,...,APN - список фактических параметров, соответствующий списку формальных параметров DPI,...,DPN.

Выделенный программе ДРП участок памяти используется как "кольцевой" буфер, в котором место для новых ПМ, потребовавшихся в процессе обслуживания эксперимента, выделяется программой ДРП по дисциплине FIFO<sup>/4/</sup>.

Использование предложенного в работе<sup>/2/</sup> метода ОСО программирования ПМ для оборудования КАМАК позволило разделить во времени создание программных модулей и конкретизацию способа /адреса/ размещения соответствующих блоков в крейтах, номеров крейтов и т.д. Это, в свою очередь, позволило выделить работу по конкретизации конфигурации используемого оборудования КАМАК в отдельную задачу и разработать для ее решения программу CCON, не зависящую от методики эксперимента.

4. Программа CCON, SAV предназначена для описания конфигурации оборудования КАМАК, используемого в конкретной САЭ. Эта программа работает в режиме диалога, который ведется по следующим правилам:

- от пользователя требуются ответы типа "да", "нет" и значения параметров, названия /назначение/ которых программа сообщает, используя систему понятий экспериментатора;

- программа подсказывает диапазон возможных значений /выделено круглыми скобками/ и сообщает значение, которое будет принято по умолчанию /выделено квадратными скобками/;

- при ошибках пользователя сообщается диагноз и предлагается повторить ответ;

- согласие пользователя на использование значения по умолчанию /обычно - ранее введенного/ сообщается символом возврата каретки;

- все команды пользователя односимвольные и не требуют использования символа возврата каретки;

- сообщенный пользователем символ команды "разворачивается" программой в текст команды на естественном языке и визуализируется.

Принятый способ ведения диалога в сочетании с разработанной системой подсказок практически исключает потребность в предварительном обучении пользователей способу работы с программой.

Введена группа операций, позволяющая редактировать старый файл описания конфигурации или создать новый.

Для каждого функционального блока программой запрашиваются следующие значения:

- название блока /<6 символов/;

- адрес блока: номер крейта /1÷4/ и номер станции /1÷23/;

- идентификатор версии блока /A÷H/;

- режим работы: активный /блок генерирует запросы прерывания/ или пассивный.

В случае, когда способ распайки на поле коммутации контроллера соответствует принятому в нашей организации, программа вычисляет и заносит в файл базовые адреса пространств, выделенных контроллеру и функциональному блоку, адрес вектора прерывания, код маски прерывания. В противном случае /нестандартная распайка/ программа запрашивает эти значения у пользователя.

Файл содержит записи длиной 1210 байтов. Для описания пассивного блока используется одна запись, для описания активного блока - две записи. Структура описаний активных и пассивных блоков приведена ниже. Разряды последнего байта описания пассивного блока (PASBLK.M) имеют следующее назначение:

Разряды	Содержание
0 - 3	- код версии блока /один символ из интервала A÷H/;
4 - 6	- код ширины блока /ширина = код ширины + 1/;
7	- код режима блока: 0 - пассивный, 1 - активный /т.е. работающий в режиме запросов прерывания ЭВМ/.

По окончании работы программа представляет содержание файла в виде схематического рисунка крейта, на котором для каждого блока приводятся вычисленные и сообщенные пользователем значения.

```

NAME =ARRAY [1..6] OF CHAR;
CRTREG=RECORD CSR,DMR,DHR :INTEGER END;
BLKREG=RECORD A0,A1,A2,...,A15 :INTEGER END;
VECTOR=RECORD ADR: ... (*АДРЕС ВХОДА В ПРОГРАММУ*);
PRI: INTEGER; (*ПРИОРИТЕТ ПРОЦЕССОРА*) END;
CAMAC =^PASBLK;
(*-----*)
PASBLK=RECORD
  BN :NAME; (*НАЗВАНИЕ БЛОКА*);
  R :^CRTREG;
  N :^BLKREG;
  M :CHAR; (*КОД МОДЕЛИ БЛОКА*);
  U :CHAR; (*КОД ПОЛЬЗОВАТЕЛЯ БЛОКА*);
END;
ACTBLK=RECORD
  C :PASBLK;
  PN :NAME; (*НАЗВАНИЕ ПРОГРАММЫ*);
  V :^VECTOR; (*АДРЕС ВЕКТОРА*);
  P :CHAR; (*ПРИОРИТЕТ*);
  I :CHAR; (*МАСКА ПРЕРЫВАНИЙ*);
  L :INTEGER; (*ДЛИНА ПАКЕТА ДОПОЛНИТЕЛЬНЫХ ПАРАМЕТРОВ*);
END;

```

Структура описаний пассивных и активных блоков КАМАК.

Результат работы данной программы - файл описания конфигурации - используется описываемыми ниже процедурами.

5. Процедура REACAM (FILNAM:NAME;VAR CC:CAMAC) выполняет чтение файла FILNAM /описание конфигурации/ в динамическую область оперативной памяти. При благополучном завершении операции указателю CC присваивается значение, равное адресу начала этой информации в памяти.

6. Процедура CDCHN(CC:CAMAC;BLKNAM:NAME; VAR BLKPTR:CAMAC) выполняет поиск в оперативной памяти описания функционального блока с названием BLKNAM. Эта процедура использует указатель CC начала поля, содержащего описание всех блоков. Найденный адрес описания блока BLKNAM присваивается указателю BLKPTR. После этого указатель BLKPTR может многократно использоваться в прикладных /функциональных/ программных модулях, работающих с данным блоком. Поскольку поиск описания блока выполняется, как правило, на этапе подготовки ПО к работе и не требует обращения к ВЗУ, эта операция практически не влияет на временные характеристики САЭ. Процедуры REACAM и CDCHN используются совместно.

7. Процедура CDLNK (CC:CAMAC; ACTBLK, PRGNAM:NAME;...) декларирует экспериментальный канал, работающий в режиме прерывания.

Обязательными параметрами данной процедуры являются: указатель на динамическую область памяти CC, значение которому присваивает процедура REACAM; название активного блока; название программы, обрабатывающей запросы прерывания.

Процедура CDLNK подключает вектор прерывания, выделенный блоку ACTBLK, к диспетчеру прерываний и дополняет таблицы диспетчера прерываний значениями, необходимыми для правильной работы программы обработки прерывания PRGNAM. Программа обработки прерываний PRGNAM может быть выполнена в виде процедуры на языке Паскаль. Эти процедуры, как и другие программные модули проблемно-ориентированной системы ДКС<sup>3/</sup>, перед использованием раздельно транслируются, обрабатываются редактором связей и в формате .REL заносятся в специальную библиотеку<sup>5/</sup> или файловую систему RT-11.

При декларации процедур обработки прерываний могут быть использованы формальные параметры. Для того чтобы при возникновении прерывания процедуре были переданы фактические параметры, они должны быть указаны в обращении к процедуре CDLNK. Место для этих параметров обозначено выше в конце списка параметров процедуры CDLNK многоточием. Количество параметров может быть любым, их типы определены правилами языка Паскаль. Например, пусть процедура обработки прерывания декларирована следующим образом:

```

TYPE
  P = ^ARRAY[1..1024] OF INTEGER;
PROCEDURE ADCCONVERTER (A:P;LENGTH:INTEGER;FREESPACE:INTEGER)
и выполняет регистрацию данных в массив A. В конкретной системе фактические параметры описаны следующими идентификаторами:
VAR
  ARRPT:P;
  L,F:INTEGER;
  MODELS:CAMAC;
...
Тогда, если данная процедура зарегистрирована в файловой системе со спецификацией ADCCON.REL, а блок имеет название VKPOO4, то декларация канала может быть выполнена инструкцией
CDLNK(MODELS,'VKPOO4','ADCCON',ARRPT,L,F).

```

Контроль соответствия количества и типов параметров в строках декларации процедуры обработки прерывания и декларации активного канала осуществляет программист. Процедура CDLNK, помимо указанных действий, загружает в оперативную память процедуру PRGNAM, настраивает ее по месту загрузки и фиксирует ее на этом месте, исключая его из автоматического распределения на срок вплоть до отмены фиксации. Скоростные характеристики программ обработки прерываний, доступные в данной реализации, удовлетворительны и подробно описаны в работе<sup>2/</sup>.

8. Процедура CAMSEEK (FILNAM,BLKNAME;VAR BLKPTR:CAMAC) отыскивает в файле описания конфигурации оборудования FILNAM описание функционального блока КАМАК BLKNAME, переписывает это описание в динамическую память и присваивает его адрес указателю BLKPTR. После этого указатель BLKPTR может использоваться прикладными /функциональными/ программными модулями, работающими с данным блоком. Данная процедура загружает из файла FILNAM в память описание одного блока КАМАК, в отличие от процедуры REACAM, загружающей в память полное описание конфигурации оборудования. Процедура CAMSEEK предназначена в основном для отладочных задач.

9. Процедура ACTCHN(CC:CAMAC;ACTBLK:NAME;PROCEDURE PRGADR;...) заполняет вектор прерывания и формирует таблицы диспетчера прерываний. Первые два параметра описаны выше. PRGADR - идентификатор процедуры обработки прерывания. Данная процедура используется в том случае, когда процедура обработки прерывания PRGADR объединяется с управляющей программой редактором связей, в отличие от процедуры CDLNK, предназначенной для использования в системах с динамическим распределением памяти для программ обработки прерываний.

10. Описанный набор программ обеспечивает возможность:

- разделить во времени процессы программирования оборудования КАМАК и спецификации конфигурации оборудования КАМАК для всей САЭ;
- запрограммировать оборудование КАМАК /включая и обработку внешних запросов прерывания/ средствами языка высокого уровня Паскаль;
- использовать модульную организацию для ПО КАМАК;
- выполнять сборку ПО для оборудования КАМАК двумя способами: динамически в процессе работы ПО САЭ или с помощью редактора связей перед использованием ПО САЭ.

Скоростные характеристики описанной реализации для ПО на языке Паскаль для ЭВМ PDP-11/20 можно представить следующими цифрами: время выполнения одиночной КАМАК-функции 10-25 мкс; время организации обработки прерывания /холостой цикл/ 130 мкс; время поиска ПМ на поле DRP  $480+k \cdot 40$  мкс /где k - номер позиции, занимаемой ПМ на рабочем поле/; время загрузки ПМ на поле DRP  $0,1 \pm 5$  с в зависимости от типа ВЗУ и др.условий/6/.

#### ЛИТЕРАТУРА

1. Велихов Е.П., Выставкин А.Н. УСИМ, 1984, № 4, с.3.
2. Балука Г., Саламатин И.М. ОИЯИ, 10-84-573, Дубна, 1984.
3. Балука Г., Саламатин И.М. ОИЯИ, P10-85-281, Дубна, 1985.

4. Балука Г. ОИЯИ, P10-84-789, Дубна, 1984.
5. Балука Г., Саламатин И.М., Хрыкин А.С. ОИЯИ, 10-12546, Дубна, 1979.
6. Балука Г. ОИЯИ, 10-84-463, Дубна, 1984.

Рукопись поступила в издательский отдел  
31 мая 1985 года.