И Н С Т И Т У Т

ЯДЕРНОЙ ФИЗИКИ СОАН СССР

ПРЕПРИНТ И·Я Ф 78 - 29

В.И.Кононов, Д.В.Пестриков, Б.Н.Сухина

СИСТЕМА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ЭКСПЕРИМЕНТОВ ПО ЭЛЕКТРОННОМУ ОХЛАЖДЕНИЮ

Новосибирск

1978

В.И.Кононов, Д.В.Пестриков, Б.Н.Сухина

СИСТЕМА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ЭКСПЕРИМЕНТОВ ПО ЭЛЕКТРОННОМУ ОХЛАЖДЕНИЮ

А Н Н О Т А Ц И Я

В работе описана операционная система, разработанная для автоматизации экспериментов по электронному охлаждению. Система позволяет с помощью довольно простого специализированного языка описать в виде программ проведение практически любого эксперимента.

Использование системы управления комплексом НАП-М, организованной на базе ЭВМ, было одним из факторов, способствовавших ускорению проведения экспериментов по электронному охлаждению[1]. Система управления включает в себя набор аппаратуры передачи и сбора данных, аппаратуры преобразования код-аналог, аналог-код и систему программного обеспечения, организующую работу комплекса при реализации того или иного эксперимента. Окончательный комплект этой аппаратуры, используемый на НАП-М, в основном, сформировался к 1975 году[2] и существенного изменения не претерпел. Настоящая работа посвящена описанию операционной системы, используемой для автоматизации экспериментов по электронному охлаждению.

Управление комплексом организовано на базе вычислительной машины ОДРА-1325 с объемом памяти 32 К 24-разрядных слов. В качестве внешней памяти используются магнитные барабаны (256 К) и магнитные ленты. Система дополнена графическим дисплеем и графопостроителем. Имеется возможность вывода информации на широкую печать (DZM-180). Взаимодействие оператора с ЭВМ осуществляется посредством расположенного в пультовой накопителя текстового дисплея - ВИДЕОТОН-340.

Основные принципы построения системы остались прежними[3] - управление посредством программ, написанных на специализированном языке и работающих под контролем интерпретатора системы. Однако, сам язык управления и интерпретатор претерпели значительные изменения.

Все многообразие работ, выполняемых ЭВМ при управлении комплексом, может быть достигнуто с использованием в управляющих программах небольшого количества элементарных операций, которые, в основном, сводятся к обмену данными с внешними устройствами, выполнению логических операций (таких как проверка состояния ключей, преобразование кодов, организация передачи управления и др.) и выполнению несложных вычислений. Если для программирования первых двух видов работ наиболее удобным является язык, приближенный к машинному, когда в операторах фигурируют не сами величины, а их адреса в памяти, то для программирования вычислений более удобным является язык, приближенный к языку ФОРТРАН. Поэтому, язык управления должен совмещать в себе оба эти свойства. Кроме того, иногда для описания функции не реали-

зованных в интерпретаторе системы может быть полезно включение в управляющие программы сегментов, написанных в машинных командах. Реализация перечисленных свойств позволила создать весьма эффективный и достаточно простой язык управления, не требующий мощных трансляторов (таких как для языков ФОРТРАН, ПЛАН и др.).

Программа переводящая текстовую информацию в команды интерпретатору была изменена. Дело в том, что минимальной единицей информации, с которой может работать интерпретатор системы, является строка управления, определяющая вид работы и параметры интерпретатору. Поэтому, в настоящее время используется программа целиком переводящая принятую с дисплея информацию в строки управления. Это позволило избежать ряда ограничений при написании программ, присущих старой системе /3/.

В интерпретатор системы, кроме очевидных изменений, связанных с изменениями в языке, были добавлены новые функции, позволяющие ему одновременно контролировать работу нескольких (в настоящее время до 9) управляющих программ. При этом, выполнение запущенных программ производится либо по приоритету, либо по приходу внешнего прерывания, в качестве которого может служить сигнал с кнопки или внешнего устройства. Возможность одновременного управления работой нескольких программ позволяет также производить запуск программы из программы (с возвратом в вызывающую программу). Это используется в качестве одного из приемов сегментации программ в данной системе. Помимо этого, имеется возможность включения в программы ранее написанных сегментов с помощью компиллятора системы (см.ниже).

Такая организация позволила доверить часть функций системы программам, написанным на языке управления, освободив тем самым, значительную часть оперативной памяти для нужд эксперимента и достигнуть автономии при автоматизации новых экспериментов. В частности, такими программами удалось заменить большую часть "сервисных" директив (функций интерпретатора реализующих ту или иную работу комплекса). Описанные изменения привели к сокращению объема памяти, занятой интерпретатором до 3 К, а остальная память ЭВМ, в зависимости от ситуации, может использоваться для размещения управляющих программ, либо результатов измерений.

Использование разработанной системы программного обеспече-

ния позволило в значительной мере автоматизировать проведение экспериментов по электронному охлаждению. В настоящее время ЭВМ выполняет широкий круг работ от подготовки пучка к эксперименту и оптимизации режимов накопителя до ведения пультовых журналов. Обработка результатов измерений может производиться непосредственно во время эксперимента с помощью программ, написанных на языке системы, что существенно ускоряет получение результатов. Обычно, при проведении цикла измерений производится автоматический запуск от 3 до 6 программ, останавливающихся в ожидании сигналов с кнопок и исполняющих ту или иную работу в данном эксперименте. При этом, работа оператора по управлению комплексом сводится к нажатию ограниченного числа кнопок, запускающих ту или иную группу программ.

I. ЯЗЫК УПРАВЛЕНИЯ

I.I. Язык управления НАП-М предназначен для описания последовательности действий интерпретатора системы при реализации тех или иных процессов управления. Минимальной единицей информации, с которой работает интерпретатор является строка управления, текст которой вводится с пульта текстового дисплея. Элементы строки управления разделяются символами:

$$- \quad + \quad * \quad / \quad \sqcup \quad = \quad (\quad) \quad ? \quad '$$

и могут относиться либо к названию директивы интерпретатора, либо к названию величины (переменной или постоянной). Текст строки управления, содержащий название директивы (оно должно находиться в начале строки) интерпретируется как обращение к одной из подпрограмм системы. В этом случае остальные элементы строки рассматриваются как параметры этой подпрограммы. В качестве параметров директивы могут использоваться как отдельные величины, так и арифметические выражения: последовательности величин соединенных знаками арифметических действий

$$- \quad + \quad * \quad /$$

и знаком присвоения $=$. Параметры директивы отделяются неарифметическими знаками:

$$\sqcup \quad (\quad) \quad ' \quad ?$$

Вычисления арифметических выражений и операции занесения производятся последовательно слева направо, пока не встретится величина, оканчивающаяся неарифметическим знаком. При этом, в качестве параметра директивы будет использован результат последнего занесения. Например, строка

SD 1 I-1*32+2=IC 32 10

эквивалентна

SD 1 IC 32 10 ,

где IC = 32*(I-1)+2 ; строка

SD 1 10=I-1*32+2=IC 32 10

эквивалентна

SD 1 IC 32 10 ,

где IC = 32*(I-1)+2 , I=10 .

После вычисления арифметических выражений, определяющих параметры, управление передается подпрограмме системы, реализующей требуемую директиву.

I.2. Величины фигурирующие в строке могут относиться как к постоянным, значение которых определяется при трансляции строки, так и к переменным. В языке НАП-М могут использоваться постоянные четырех видов:

1. реальные десятичные числа

   I.5 .05 - I25,3 и т.д.

2. целые десятичные числа

   I I00 - I25 и т.д.

3. целые восьмеричные числа

   #I0 #I00 #-I00 и т.д.

4. текстовые постоянные

   :TEXT :FILE :NAME и т.д.

Правила написания констант различных типов ясны из приведенных примеров. Признак текстовой константы: после приема заменяется знаком ␣ (пробел).

При трансляции каждой величине отводится по две ячейки памяти. При этом численные величины хранятся в виде чисел с фиксированной запятой, т.е. в первой ячейки хранится целая часть числа, а во второй дробная. Выбранное представление чисел связано с тем, что при управлении установкой наиболее часто встречающимися величинами являются целые: номера устройств, ключей, коды, которые требуется выслать либо принять с тех или иных устройств и т.д.

Правило написания переменных такое же как в языке Фортран:

X ABC PRIZN и т.д.

I.3. Имеется несколько ограничений на написание текста строки управления:

1. текст не должен превышать 60 символов;
2. в тексте строки не должно встречаться подряд два разделительных символа (за исключением знака минус перед числом);
3. название численных постоянных не должно содержать буквенных знаков, а названия восьмеричных чисел знаков 8 и 9;
4. название переменной не должно содержать более I2 символов.

Нарушение любого из этих пунктов рассматривается транслятором системы, как грамматическая ошибка и помечается значком на экране дисплея после текста строки, затем транслятор требует новый текст строки взамен ошибочного.

I.4. Группа строк управления, следующая за директивой интерпретатора

PRØG :NAME

образуют программу. Окончание приема программы сигнализируется директивой *END*. Строки следующие между директивами

PRØG :NAME и END

не исполняются, а заносятся массивами по I6 слов на программный файл барабанов в качестве текста программы под названием *NAME*.

В начале программного файла помещается каталог программ, находящихся в данный момент на барабане. Первое слово каталога содержит число программ, далее группами по 4 слова располагает-

ся информация о программах

I,2 слова – название программы,

3 – слово – адрес начала программы на барабане

4 слово – количество строк в программе

Одновременно на программном файле может находиться до I50 программ. По желанию оператора текст программы может быть записан на магнитную ленту.

I.5. В интерпретаторе НАП-М реализовано 39 директив, которые делятся на исполняемые и неисполняемые.

Неисполняемых директив две

директива окончания приема текста программы

END

директива, требующая выделения программе COMMON блока

GIV PRIOR :NAM LEN ,

где PRIOR – приоритет программы, :NAM – название требуемого блока памяти, LEN – длина блока в словах.

Если в программе директива GIV отсутствует, в качестве COMMON блока будут использованы первые 4096 слов COMMON блока интерпретатора.

Если две программы требуют блок памяти под одним названием, то отведенный блок памяти будет общим для обеих программ (т.что становится возможным обмен данными между программами). В этом случае COMMON блок помещается сразу за первой требующей программой.

Отметим, что COMMON блок программы управления не пересекается с массивом содержащим значения переменных и постоянных программы. Он служит специально для организации работы с массивами данных, которые при управлении установкой, поступают с внешних устройств в виде массивов целых чисел, занимающих последовательно ячейки памяти программы.

Полный список исполняемых директив приведен в Приложении I. Ниже мы обсудим лишь некоторые из них, наиболее часто встречающиеся при программировании на языке управления, или при работе на установке.

Директива READ

READ X Y – прочитай значения X и Y с пульта текстового дисплея. Число параметров произвольно.

Директива WRTE – выведи на экран текстового дисплея информацию. Например, после исполнения строки

WRTE :X X :Y Y

на экране появится надпись

X (значение X) Y (значение Y)

курсор будет переведен в начало следующей строки. Число параметров произвольно. Имеется модификация. Если в качестве первого спецсимвола используется ' ,вывод информации не сопровождается переводом строки:

WRTE ' :X X :Y Y

Директива DS – чтение с барабанов. Строка

DS A1 A2 L NF

вызывает чтение массива длиной L , с файла NF на барабанах начиная с ячейки A2 в ячейку памяти A1 . Если в программе использована директива GIV , адрес A1 отсчитывается от начала заказанного блока. Директива имеет 4 параметра.

Директива SD – запись массива из памяти на барабан. Имеет те же особенности, что и директива DS .

Директива RVP – обратись к оконечному устройству.

RVP NU AP L AINU,

здесь NU – номер внешнего устройства (число от I до I5), в зависимости от номера NU производится чтение, либо запись массива длиной L , начиная с адреса на устройстве AINU в память ,начиная с ячейки AP . Если директива имеет 5 параметров, действие ее сводится к подготовке работы с устройством, а передача производится после исполнения ближайшей директивы RVP с четырьмя параметрами.

Директива GOTO – безусловный переход. Строка

GOTO X

вызывает передачу управления строке программы с номером X . В качестве параметра может быть как постоянная, так и переменная величина:

GØTØ 5 , GØTØ X .

Если в качестве первого спецсимвола использован знак ' , переход понимается как относительный, то-есть управление передается на X -ую строку отсчитывая от строки GØTØ 'X .

Директива IF - перейди по знаку. Строка

IF X MN

вызывает передачу управления строке MN программы, если $X \geqslant 0$; если $X < 0$ выполняется следующая строка программы. Строка

IF X M1 M2

вызывает передачу управления строке M1, если целая часть положительна, и строке M2 если она равна нулю. При $X < 0$ управление передается на следующую строку. Если первый спецсимвол ' , переходы понимаются как относительные.

Директива SUP - остановись в ожидании сигнала с кнопки. Строка

SUP NPINT NSTR :NAME

вызывает остановку программы NAME. По приходу сигнала с кнопки NPINT, производится запуск программы с названием NAME со строки NSTR . Если последний параметр опущен, останавливается программа, исполняющая директиву SUP .

Директива M - занеси в память команду. Первые четыре параметра этой директивы рассматриваются, как элементы команды языка PLAN.

M F X AP XMØD

Директива формирует команды и помещает их в блок памяти программы, начиная с первой ячейки. Выполнение сформированного сегмента программы инициируется строкой

M #72 0 0

вызывающей возврат в работающую программу. При этом, в первые

три аккомулятора помещается информация:

X0 - адрес возврата;
X1 - адрес начала блока памяти программы;
X2 - адрес начала таблицы адресов библиотеки интерпретатора

и управление передается первой команде сформированного сегмента.

Имеются еще две модификации, облегчающие программирование с директивами M . Если спецсимвол перед адресной частью команды ' , то к адресу будет добавлено начало блока памяти программы. Строка

M F X'VAPRØG XMØD

транслируется в две команды PLANa

LDX XMØD 'VAPRØG'
F X 0(XMØD)

т.е. позволяет выполнять операции с участием непосредственно переменных программы.

Директива PRØG - вводи программу с пульта текстового дисплея. После получения директивы

PRØG :NAME

интерпретатор все последующие строки заносит на барабан (в конец программого файла) до тех пор, пока не появится директива END, сигнализирующая о конце приема программы. По приему директивы END данные о программе NAME заносятся в каталог, находящийся в начале программного файла.

Директива RUN - запусти программу. Строка

RUN :NAME M1

вызывает запуск программы NAME со строки M1 , Если программы в памяти нет, то интерпретатор попытается найти ее на барабане и оттранслировать. Если программы нет ни в памяти, ни на барабанах, на экране дисплея высвечивается надпись LØAD . Если строка RUN :NAME M1 выполняется какой-либо программой, то после окончания работы NAME управление

передается строке вызвавшей программы, следующей за

RUN :NAME M1

Запуск с первой строки указывать необязательно:

RUN :NAME эквивалентно RUN :NAME 1

Наличие лишних параметров не влияет на выполнение директивы.

Директива LIST - выведи текст программы на экран текстового дисплея. Вывод текста производится блоками по I4 строк.

II. ФУНКЦИИ СИСТЕМЫ, РЕАЛИЗОВАННЫЕ В ВИДЕ ПРОГРАММ УПРАВЛЕНИЯ

Функции системы, не требующиеся непосредственно при работе программ управления реализованы в виде программ, написанных на языке НАП-М. К таким функциям относятся:

I. редакция текстов программ;
2. компилляция программ;
3. работа с магнитными лентами;
4. вывод текстов программ на широкую печать.

2.I. Редакция программы осуществляется программой ALT.

После запуска программы

RUN :ALT

строкой

AL :NAME1 :NAME2

указывается название редактируемой программы NAME1 и выходной программы NAME2 . Дальнейший процесс редакции выполняется аналогично, описанному в работе /2/.

2.2. Компилляция программ осуществляется программой PILE . Входными параметрами программы PILE являются название master -программы и название программы, образующейся в результате компилляции.

Компилляция происходит следующим образом. Программа PILE переписывает master-сегмент в конец программного файла и производит просмотр строк полученного текста. Если строка содержит директиву RUN :NAME , проверяется не встречалось ли обращение к программе NAME ранее. Если обращение появляется впервые, программа PILE переписывает текст программы NAME в конец имеющегося текста, а название NAME заносится в список собираемых программ. Для организации возврата в master сегмент в конце текста программы NAME компиллятором помещается строка

GØTØ RETn

где n - номер программы NAME в списке собираемых программ. Исходная строка master - сегмента

RUN :NAME M1 1=X1 5=Y

заменяется строкой

GØTØ JNAME MM=RETn :NAME M1 1=X1 5=Y

список параметров исходной

строки, в которой JNAME получается сложением номера строки текста, в которой начинается NAME, с M1 , MM - номер строки следующей за обращением к программе NAME . Программа PILE производит просмотр всех строк полученного текста. Поэтому, если какая-либо из уже собранных подпрограмм NAME содержит обращение к NAME1 , программа NAME1 будет включена в сборку тем же способом что и NAME . Всего в сборку может быть включено I2 различных подпрограмм.

2.3. Работа с магнитными лентами осуществляется с помощью группы программ, которые позволяют:

I. прикрепить к программе названную магнитную ленту или ленту SCRATCH TAPE ;

2. записать или считать с магнитной ленты текст программы; управляющую таблицу; группу блоков данных, содержащих результаты измерения;

3. Вывести на экран текстового дисплея либо на широкую печать названия блоков информации, хранящейся на ленте;

4. проводить редакцию лент;

5. Открепить от программы прикрепленные ранее ленты.

Одновременно система может оперировать с I5 лентами. Полный список программ, организующих работу с магнитными лентами, приведен в Приложении II.

2.4. Текст любой программы, находящейся на программном файле, может быть выведен на широкую печать DZM - I80. Для этого используется программа LIST , входным параметром которой указывается название выводимой программы. Текст программ выводится с нумерацией строк.

Ш. ОСОБЕННОСТИ РАБОТЫ ПРОГРАММЫ

3.I. Ввиду большого объема памяти (~23 K) в существующей системе возможен одновременный запуск нескольких программ (в настоящее время до девяти). При этом, запущенные программы могут работать либо как независимые, выполняя отдельные задачи, либо часть программ могут работать совместно, запуская друг друга по мере необходимости. В этом случае обмен данными между программами осуществляется через COMMON- блок, общий для всей группы. Запуск программ из master - программы организуется директивой RUN и понимается системой как прерывание работы master - программы. Поэтому, возврат в master - программу производится по остановке запущенной программы, на строку следующую за директивой RUN . Количество прерванных программ, в настоящее время, допускается до шести. Так как вообще программы хранятся на барабанах в текстовом виде, то при первом обращении к программе происходит ее трансляция и размещение её в оттранслированном виде в памяти, при последующих обращениях программа не транслируется.

В каждый данный момент работает последняя из запущенных программ, при этом остальные программы находятся в состоянии ожидания запуска. Очередность запуска определяется несколькими факторами. Если работающая программа заканчивает работу, запускается последняя из прерванных программ. Другим фактором, вызывающим запуск программы может быть появление сигнала внешнего прерывания (пинта) с номером, которого ожидает данная программа, У такого способа запуска программы максимальный приоритет, т.е. прерывается работа любой программы и запускается требуемая. По приходу прерывания происходит запоминание номера прерванной программы, который позволяет восстановить информацию о состоянии программы при последующем ее продолжении работы. Передача управления при запуске программы пин-

том происходит после выполнения строки,в которой произошло прерывание.

Возможность одновременного нахождения нескольких программ в памяти с запуском от внешнего устройства позволяет организовать, например, непрерывное слежение за режимом установки (либо отдельных элементов). В этом случае следующая программа будет запускаться автоматически по окончании работы более приоритетных программ.

3.2. При запуске программы начинается последовательное выполнение строк управления. Как было описано выше, очередная строка управления просматривается интерпретатором слева направо и производится выполнение всех арифметических действий и занесение результатов.по адресу параметра следующего за знаком = . При этом, интерпретатор формирует новую строку управления с сокращенным числом параметров, в которой арифметические выражения исходной строки заменены результатами и помещены по адресам правых частей выражений. После этого управление передается той части интерпретатора, которая отвечает за исполнение данной директивы. Если директива имеет нулевой номер, то исполняется следующая строка программы. Такая организация выполнения строки управления позволяет совместить выполнение промежуточных арифметических действий с исполнением требований директивы, что сокращает текст программ .

По выполнению директивы управление передается в ту часть интерпретатора, которая следит за исполнением программ. Производится контроль - выполнена ли данная программа и не требуется ли запуск какой-либо другой программы.

Приложение I

ДИРЕКТИВЫ СИСТЕМЫ УПРАВЛЕНИЯ

I. Директивы работы с таблицей:

LK – занеси код в таблицу,

LK IN IC NC KØD NC1 KØD1 ...

IN – начальная строка таблицы
IC – конечная строка таблицы
NC, NC1 – номера ЦАПов
KØD, KØD1 – коды, которые нужно заслать в указанные ЦАПы.
LØ – измени код в ЦАПе с определенным шагом

LØ IN IC NC DKØD NC1 DKØD1 ...

DKØD, DKØD1 – шаг изменения кода в соответствующих номерах ЦАПов. Остальные параметры те же, что и у директивы *LK*. Значение кода для заданного ЦАПа изменяется по формуле:

$KØD(i)=KØD(IN)+(i-IN)*DKØD$, где i – текущая строка в пределах с *IN* по *IC*.

ØN – "включи" ЦАПы

ØN n m ...

n, m – номера ЦАПов, которые нужно сделать доступными для ЭВМ.

ØFF – "выключи" ЦАПы

ØFF n m ...

Действует аналогично директиве *ØN*.
Если директивы *ØN* или *ØFF* без параметров то "включаются" или "выключаются" все ЦАПы.

2. Директивы работы с барабанами.

DS – перенеси с барабана в память

DS m n l f

m – номер ячейки в оперативной памяти
n – номер ячейки на барабане
l – длина пересылки (в ячейках)
f – номер файла на барабане.

SD – перенеси из памяти на барабан. Параметры те же, что и у директивы *DS*.

3. Директивы ввода, вывода

READ – введи данные или текст

READ X Y Z ...

X, Y, Z – переменные, на место которых вводятся данные или текст. Прием информации заканчивается знаком *NL* независимо от ожидаемого числа параметров.

WRTE – выведи значение переменных

WRTE X Y :AA

X, Y, :AA – переменные. Причем для переменной *:AA* выведется на видеотон ее собственное название, т.е. *AA*. Если в качестве первого спецсимвола используется *'(WRTE'X Y)*, то после высвечивания переменных строка не переводится.

ØUI – выведи из оперативной памяти на видеотон числа.

ØUI m l dm

m – адрес ячейки, с которой нужно вывести
l – длина вывода (в ячейках)
dm – шаг, с которым изменяется адрес *m*, может быть равно либо I, либо 2.

ØUR – выведи в знаковом виде

ØUR m l

l – число знаков.

4. Директивы передачи в памяти

LET – присвой переменной значение ячейки из памяти

LET X1 M1 X2 M2

X1, X2 – переменные в рабочей программе
M1, M2 – ячейки в памяти
CLE – занеси целое число в ячейку памяти

CLE m a l

m – число, которое нужно занести
a – начальный адрес в памяти
l – длина заполняемого массива

5. Директивы перехода

GØ – безусловный переход

GØ m – перейди на строку программы с номером m . Если в качестве первого символа используется ' , то происходит переход относительно исполняемой строки и принимаемой за 0№ . В зависимости от знака m , в этом случае, переход может быть осуществлен выше или ниже исполняемой строки.

IF – условный переход

IF A n1 n2 – перейди на строку n1 , если целая часть A>0 , на строку n2 , если целая часть A=0 . При A<0 управление передается следующей строке. Если первый спецсимвол ' , то переход происходит относительно исполняемой строки с учетом знака при n1 и n2. Если директива содержит два параметра IF A n , то проверка на целую часть A=0 не делается.

6. Программные директивы.

PRØG – прими программу с видеотона

PRØG :NAME

NAME – название программы

После этой директивы система становится на прием текста организуемой программы. Заканчивается прием директивой END . Текст полученной программы заносится на программный файл барабанов под названием NAME .

RUN – защусти программу в работу

RUN :NAME m

Запускается программа под названием: NAME со строки m. Если m отсутствует, то запуск происходит с I-ой строки.

RE – смени приоритет у программы

RE :NAME P

P – приоритет (число)

SUP – останови программу

SUP NPINT NSTR :NAME

вызывает остановку программы NAME

NPINT – номер кнопки пинга запускающего программу.

NSTR – номер строки, с которой следует начать работу по приходу сигнала с кнопки NPINT. Если последний параметр опущен, останавливается программа, исполняющая директиву SUP .

END – конец ввода программы, коррекции программы.

7. Прочие директивы

DIS – подготовь ячейку для высвечивания на графический дисплей.

DIS X Y ADR

X – координата на оси X

Y – координата на оси Y

ADR – адрес в памяти, куда поместить подготовленную информацию.

M – занеси в память команду.

Параметры этой директивы рассматриваются как элементы команды языка PLAN.

M F X AP XMØD

F – код команды

X – аккумулятор

AP – адресная часть команды

XMØD – модификатор

Директива готовит команды и помещает их в блок памяти программы с первой ячейки. Выполнение набранного сегмента из команд PLANa начинается после набора строки: M#72 0 0

При исполнении директивы M в первые три аккумулятора заносится следующая информация:

X0 – адрес возврата в программу после работы плановского сегмента

X1 – адрес начала блока памяти программы

X2 – адрес начала таблицы адресов библиотеки интерпретатора.

С помощью данной директивы, и используя X2 , можно обратиться в плановском сегменте к любой из подпрограмм, содержащихся в библиотеке интерпретатора.

Имеются две модификации данной директивы облегчающие программирование.

Если перед адресной частью команды стоит спецсимвол ', то к адресу будет добавлено начало блока памяти программы (т.е. содержимое X1)

M F X 'AP XMØD

Если строка имеет вид:

M F X(VAR XMØD

то она транслируется в две команды:

LDX XMØD 'VAR'
F X 0(XMØD)

т.е. можно работать с переменными программы.

RVP - обратись к внешнему устройству.

RVP NU AP L AINU

NU - номер внешнего устройства (число от I до I5). В зависимости от номера NU производится чтение либо запись.

AP - начальный адрес для приема или передачи информации.

L - длина передачи в ячейках

AINU - адрес в устройстве, с которого начинается работа. Если директива имеет 5 параметров (значение 5-го параметра безразлично), то производится лишь подготовка работы с устройством, а исполнение происходит при ближайшем обращении к директиве RVP, но с 4-мя параметрами.

PR - распечатайсписок программ, имеющихся на программном файле

DE - уничтожь программу на программном файле

DE :NAME

Стирается программа под названием NAME

LIST - выведи на видеотон текст программы.

LIST :NAME - выводится программа под названием NAME.

PRS - распечатай список программ, находящихся в оперативной памяти.

Высвечивается следующая информация о программах:

NP S
NAME APAR L NV NS M PINT ASTR NCØM ACØM LCØM P

NP - число программ, находящихся в памяти
S - информация о номерах остановленных программ
NAME - название программы
APAR - адрес начала размещения программы в памяти
L - длина программы с учетом заказанного
NV - число переменных, используемых в программе

NS - число строк в программе
M - номер исполняемой строки
PINT - № пинта, ожидаемой программы
ASTR - адрес начала адресов строк
NCØM - название CØMMØN
ACØM - адрес CØMMØN
LCØM - длина CØMMØN
P - приоритет программы

DES - уничтожить программу в памяти

DES :NAME

NAME - название стираемой программы.

Если директива без параметра, то уничтожаются все находящиеся в памяти программы.

GIV - определи приоритет программе и отведи CØMMØN

GIV P :NCØ LC

P - приоритет (любое число)
NCØ - название CØMMØN
LC - длина CØMMØN

K - дешифровка кодов, считанных с внешних устройств

K NU AD AK L

NU - номер устройства, определяет своим номером структуру дешифровки

AD - начальный адрес, откуда берется код для работы

AK - начальный адрес, куда помещается обработанная информация.

L - длина обрабатываемого массива.

8. Арифметические директивы

ABS X a - возьми модуль числа X, положи в a
SØRT X a - корень квадратный
LN X a - натуральный логарифм
LG X a - десятичный логарифм
EXP X a - экспонента
SIN X a - синус
SH X a - гиперболический синус
ACØS X a - арккосинус.

Приложение П.

Рабочие программы, написанные для системы управления НАП-М можно разделить на три группы по выполняемым функциям обслуживания системы.

I. Программы для организации и редакции рабочих программ, а также для котроля и их уничтожения.

Программа *FILE* служит для компилляции программ. При запуске программа требует два параметра: название *master* - программы и название программы, которая будет собрана. Вновь полученная программа помещается в конец программного файла и счетчик программ увеличивается на единицу.

Программа *ALT* - для редакции текста программ. Программа запрашивает названия редактируемой и отредактированной программ. Оповещение о запросе - программа всегда сообщает высвечиванием по видеотон шести точек (.....). Затем запрашивается информация об исправляемых строках. Имеются следующие возможности:

а) если сразу набираются строки управления без указания исправляемых номеров строк, то текст помещается перед началом программы;

б) если указывается номер строки и число исправляемых строк равно нулю, то текст выставляется перед указанной строкой;

в) если указан номер и число строк, то вновь набираемые строки будут вставлены на это место;

г) если за указанием номера и числа строк текст не последует, то эти строки будут просто уничтожены.

Конец редакции происходит по приему директивы

Программа *PR* - для опроса числа программ, находящихся на программном файле и вывода названий этих программ на видеотон. При запуске требует два параметра - номер файла (2-ой или I2-ый) и номер программы, с которого выводить на видеотон названия программ.

Программа *DEAE* - для уничтожения программ на программном файле. При запуске запрашиваются поочередно номера программ (в порядке их расположения), которые нужно уничтожить.

Заканчивается прием списка номеров *0 н* номером. В результате работы уничтожаются названные программы, список сжимается, счетчик программ соответственно уменьшается.

Программа *LIST* - для вывода на широкую печать (*DZM*-I80) текста программ с параллельным высвечиванием строк на видсотон. Запрашивается программой один параметр - название программы.

2. Программы организующие работу с магнитными лентами.

Прежде чем работать с магнитными лентами необходимо подключить магнитофоны к машине, прикрепить нужные ленты к системе с помощью программы *MTCR* . Если будет обращение из какой-либо программы к неприкрепленной ленте, произойдет ошибка и машина остановится. После окончания работы с лентами необходимо открепить *MT* от системы с помощью программы *MTUNL* и отключить магнитофоны от машины (подключение и отключение магнитофонов производит оператор в машинном зале).

Программа *MTCR* - для прикрепления магнитных лент (*MT*) к системе. Запрашивается программой три параметра - мода прикрепления (*0* - для *SCRATCH TAPE* , либо *1* - для ранее названных лент), название прикрепляемой ленты (из 7 символов)и номер ленты (от I до I5).

Программа *MTUNL* - для закрытия и открепления *MT* от системы. Запрашивается один параметр - номер *MT*.

Программа *REP* - для считывания программ с *MT* на программный файл барабанов. Запрашивается два параметра - название программы и номер ленты.

Программа *WRP* - для записи программ на *MT*. Требуется два параметра - название программы и номер ленты, на которую будет производиться запись.

Программа *RET* - для считывания управляющих таблиц с и записи их на барабаны. Запрашивается также два параметра - дата из шести цифр, под которой записана управляющая таблица на ленту и номер ленты.

Программа *WRT* - для записи управляющих таблиц на *MT* Параметрами являются дата из шести цифр и номер ленты.

Программа *MTRD* – для чтения блоков данных с *MT* на барабаны, Три параметра – дата, как и для таблиц и номера вариантов с какого и по какой.

Программа *MTWRDA* – для записи блоков данных на *MT*. Три параметра, такие же как и для программы *MTRD*.

Программа *MTLIS* – для вывода на видеотон и на широкую печать (*DZM*-180) всех названий блоков данных, содержащихся на данной ленте. Если блоком данных является программа, то печатается название программы и число строк в ней, если управляющая таблица, то слово "*NAPTABL*" и дата записи, если варианты экспериментов то слово "*NAPDATA*", дата записи и число вариантов. Запрашивается один параметр – номер прикрепления *MT*.

Программы *MTDU, MTED, EDTP, EDA* используются для автоматической записи новых программ и редактирования магнитных лент. Для работы программ требуются две ленты – одна с редактируемым содержанием и вторая – "чистая", на которую будет производиться перепись.

Программа *DUMP* – записывает на ленту состояние программного файла и используемую управляющую таблицу с указанием даты записи.

Программа *LØAD* – восстанавливает состояние программного файла и управляющей таблицы.

Литература

1. Г.И.Будкер, Я.С.Дербенев и др. А.Э., 40, вып.I, 1976.
2. Ю.А.Болванов, В.И.Кононов и др. ПТЭ, № 4, стр.40, 1976.
3. Д.В.Пестриков, Б.Н.Сухина. Препринт ИЯФ 75–22, Новосибирск, 1975.

Работа поступила – 28 марта 1978 г.

Ответственный за выпуск – С.Г.ПОПОВ
Подписано к печати 3.IУ-1978 г. МН 02746
Усл. I,5 печ.л., I,2 учетно-изд.л.
Тираж 200 экз. Бесплатно
Заказ № 29.

Отпечатано на ротапринте ИЯФ СО АН СССР