

СООБЩЕНИЯ
ОБЪЕДИНЕННОГО
ИНСТИТУТА
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ

ДУБНА



Ц8406

Д-691

408/2-79

В.И.Дорогов, М.Кунике

29/1-79

P10 - 11919

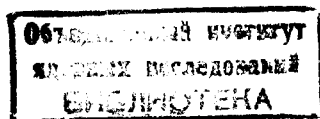
ПОДПРОГРАММЫ ДЛЯ ОБРАБОТКИ ФАЙЛОВ
НЕСТАНДАРТНОГО В ЕС ЭВМ ФОРМАТА (I)

1978

P10 - 11919

В.И. Дорогов, М.Кунике

ПОДПРОГРАММЫ ДЛЯ ОБРАБОТКИ ФАЙЛОВ
НЕСТАНДАРТНОГО В ЕС ЭВМ ФОРМАТА (I)



Дорогов В.И., Кунике М.

P10 - 11919

Подпрограммы для обработки файлов нестандартного в ЕС ЭВМ формата (1)

Приводятся описание и примеры использования фортрановских подпрограмм для чтения нестандартных файлов и преобразования информации нестандартного формата: BUFFIN, BYTINT, CONVRT, DECINT, INPUT, ZONA. Подпрограммы написаны на языке АССЕМБЛЕР и работают под управлением операционной системы ОС ЕС любой версии.

Работа выполнена в Лаборатории высоких энергий ОИЯИ.

Препринт Объединенного института ядерных исследований. Дубна 1978

Dorogov V.I., Kunike M.

P10 - 11919

Subroutines for Input from Non-Standard-Format Files in the OC/EC Operation System

Description and examples of FORTRAN subroutines for input from non-standard files and transformation of information with non-standard-format: BUFFIN, BYTINT, CONVRT, DECINT, INPUT, ZONA are proposed. Subroutines are written in ASSEMBLER language and are applied in the OC/EC operation system.

The investigation has been performed at the Laboratory on High Energy Physics, JINR.

Preprint of the Joint Institute for Nuclear Research.

Dubna 1978

Часто информация, полученная в физическом эксперименте, записывается на такие промежуточные носители информации, как магнитная лента или перфолента. Эта информация может быть частично обработана уже в ЭВМ, которая находится на линии с физической аппаратурой, или совсем не обработана /как, например, при выводе на перфоленту с различных перфораторов/. Наличие на линии различных ЭВМ и устройств вывода на перфоленту порождает разнообразные форматы такой информации.

В связи с развитием ЕС ЭВМ во все более широком масштабе внедряются ЭВМ единой серии для обработки физической информации, накопленной в экспериментах. Кроме того, в режиме пакетной обработки на ЕС ЭВМ поступает информация, записанная на других ЭВМ в формате, нестандартном для ЭВМ этой серии.

Программы обработки физической информации большей частью написаны на языке программирования FORTRAN-IV, что предъявляет определенные требования к структуре внешних файлов. Для чтения информации разного формата, нестандартного для языка FORTRAN-IV, требуется применение специальных подпрограмм чтения и преобразования.

Другой проблемой, которая возникает при обработке, является то, что FORTRAN-IV не дает таких возможностей, которые предоставляют языки низкого уровня, такие, как АССЕМБЛЕР /например, доступ к одному байту, что иногда необходимо/.

Ниже приводится описание 6 подпрограмм /BUFFIN ,
BYTINT ,CONVRT ,DECINT ,INPUT ,ZONA / для чтения
нестандартных файлов и преобразования информации не-
стандартного формата.

Для каждой программы приводятся примеры исполь-
зования. При этом предполагается, что подпрограммы
находятся в библиотеке загрузочных модулей USR1.PGMLIB.

Для файлов на магнитной ленте запись должна быть
9-дорожечной с плотностью 800 ВР1 /32 байт/мм/, для
файлов на перфоленте количество дорожек может быть
от 5 до 8.

ПОДПРОГРАММА ЧТЕНИЯ НЕФОРТРАНОВСКИХ ФАЙЛОВ BUFFIN

Среди операторов FORTRAN-IV отсутствует опера-
тор типа BUFFER IN, входящий в состав операторов
БЭСМ-FORTRAN, что приводит к невозможности чте-
ния "чужих", т.е. нефортрановских бесформатных файлов,
при использовании только стандартного набора операто-
ров ввода/вывода FORTRAN-IV.

Подпрограмма BUFFIN позволяет читать "чужие"
бесформатные файлы с магнитных лент и перфокарт.

Структура подпрограммы BUFFIN

Подпрограмма/подпрограмма-функция типа INTEGER*4.

Имя программы: BUFFIN

вызываемых программ нет

вызов 1) как подпрограммы:

CALL BUFFIN(<область ввода>, <длина ввода>)

2) как подпрограммы-функции:

BUFFIN (<область ввода>,<длина ввода>)

параметры файла чтения:

DDNAME = FTBUFFIN

язык программирования: АССЕМБЛЕР

операционная система: ОС/ЕС

Использование подпрограммы BUFFIN

BUFFIN можно вызывать, как подпрограмму или как
подпрограмму-функцию. Вызов подпрограммы-функции
BUFFIN позволяет программисту проверить условие до-
стижения конца файла /EOF/. Значение функции BUFFIN
равно 0, пока не достигнуто EOF, и становится рав-
ным 1, если выполнено условие EOF. При вызове под-
программы BUFFIN EOF остается незамеченным для
программиста, обнаружение EOF приводит лишь к тому,
что соответствующий CALL BUFFIN не вызывает переда-
чу данных в область ввода. При использовании вызова
функции идентификатор BUFFIN следует описать в вы-
зывающей программе, как INTEGER*4.

Все остальные, ниже описанные свойства функций
и подпрограммы идентичны.

Параметры подпрограммы. Параметр <об-
ласть ввода> задает начальный адрес, куда передается
информация с внешнего файла. Область ввода определяет-
ся любым идентификатором, описанным в вызывающей
программе. Длина ввода задается числом передаваемых
за один вызов байтов. Длину ввода можно указать кон-
стантой или переменной типа INTEGER* 4 .

Алгоритм работы подпрограммы. При каж-
дом вызове подпрограммы BUFFIN передаются в об-
ласть ввода последовательно куски информации с чужого
внешнего файла, по длине равные значению параметра
длины ввода. Структура файла на внешнем носителе при
этом не учитывается.

Описание внешнего файла. Подпрограмма
BUFFIN не обращается к внешнему файлу по номеру фай-
ла: она жестко связана с файлом, имя которого -FTBUFFIN.
Таким образом, для каждой фортрановской программы,
вызывающей BUFFIN; существует только один внешний
файл-FTBUFFIN. Этот файл следует описать оператором DD.

Для файла на МЛ в операторе DD должны быть за-
даны параметры UNIT, LABEL, DISP, VOL, DCB.

Для случая чтения файла с перфокарт перфокарты включаются во входной поток, а файл описывается параметром DATA.

Пример вызова подпрограммы-функции

```
// EXAMPLE JOB HUSER, NAME
//          EXEC FORSTCLD
// FORT. SYSIN DD *
      INTEGER* 2 B4WORD (3)
      INTEGER* 4 BUFFIN
10  IF(BUFFIN(B4WORD,6).EQ.1)GOTO 20
      .
      .
      .
      GOTO 10
20  STOP
      END
/*
// GO.SYSLIB DD
//          DD DSN=USR1.PGMLIB,DISP=SHR
// GO.FTBUFFIN DD UNIT=TAPE, LABEL=(1,NL),VOL=SER=BESM4T
//          DCB =(RECFM=U, BLKSIZE= 4000)
//
```

ПОДПРОГРАММА ЧТЕНИЯ НЕФОРТРАНОВСКИХ ФАЙЛОВ ZONA

Подпрограмма ZONA является упрощенным вариантом подпрограммы BUFFIN. Она отличается от BUFFIN

тем, что, во-первых, она применима только для чтения файлов на МЛ, во-вторых, при каждом вызове ее передается вся информация одной зоны /одного физического блока/ на МЛ в область ввода, а не по кускам определенной длины, как при использовании подпрограммы BUFFIN.

Структура подпрограммы ZONA

Подпрограмма/подпрограмма-функция типа INTEGER* 4
имя программы: ZONA
внутренних входов нет
вызываемых программ нет
вызов 1) как подпрограммы:
CALL ZONA(<область ввода >)
2) как подпрограммы-функции:
ZONA(<область ввода >)
параметры файла чтения:
DDNAME=FTZONA, DEVD=TA, RECFM=U
язык программирования: АССЕМБЛЕР
операционная система: ОС/ЕС.

Использование подпрограммы ZONA

ZONA можно вызывать, как подпрограмму или как подпрограмму-функцию. Вызов подпрограммы-функции ZONA позволяет программисту проверить условие конца файла чтения /смотри описание BUFFIN /.

Параметр области ввода задает начальный адрес, куда передается информация, содержащаяся в одной зоне. При задании области ввода следует учесть, что область ввода не должна быть короче, чем самая длинная зона, считываемая при помощи подпрограммы ZONA. Параметром области ввода может быть любой идентификатор, описанный в вызывающей программе.

Описание внешнего файла. Подпрограмма ZONA не обращается к внешнему файлу по номеру файла, она жестко связана с файлом, имя которого - FTZONA. Таким образом, для каждой фортрановской программы,

задает число символов кода программиста, отличных от стандартных кодов перфоленты. Оно может быть любым десятичным числом ≤ 256 .

Алгоритм работы подпрограммы. При каждом обращении к подпрограмме CONVRT происходит чтение информации с перфоленты до символов EOR, перекодировка и запись перекодированной информации в массив, описанный в параметре <область ввода>. Перекодирование информации осуществляется внутри программы с помощью таблицы перекодировки, в которой некоторые элементы исправлены с учетом кода программиста. Эти поправки осуществляются заменой соответствующих элементов таблицы кодами программиста. Эта информация вводится с перфокарт во входном потоке.

Описание внешнего файла. Конец записи на перфоленте должен быть отмечен двумя пробивками EOR. Подпрограмма считывает всю запись до EOR за одно обращение, причем сами символы EOR не входят в запись. При вводе перфоленты необходимо набрать комбинацию EOR на пульте управления перфоленточной станцией. Вводимые коды пользователя состоят из последовательных пар символов, пробитых на перфокартах. Первый - символ ДКОИ-12, второй - комбинация кода программиста, соответствующая этому символу. Эта информация включается во входной поток. Для файла данных в операторе DD достаточно указать параметр UNIT. Если длина записи > 8000 байтов, следует указать соответствующий параметр BLKSZISE в DCB.

Пример вызова подпрограммы

```
// EXAMPLE JOB HUSER, NAME
// FORTRAN EXEC FORTGCLD
// FORT.SYSIN DD *
  INTEGER* 2 DATA (1000)
  CALL CONVRT (DATA, 32)
  :
```

```

.
.
STOP
END
/*
// GO.SYSLIB DD
           DD DSN=USR1.PGMLIB, DISP=SHR
// GO.DATAIN DD UNIT=007
// GO.CARDIN DD
  <Таблица перекодировки>
/*
//
```

ПОДПРОГРАММА ЧТЕНИЯ ИНФОРМАЦИИ С ПЕРФОЛЕНТЫ INPUT

В некоторых задачах обработки информации на ЭВМ возникает необходимость считывания информации с перфоленты побайтно и интерпретации каждого байта как целого числа. FORTRAN-IV не дает такой возможности.

Подпрограмма INPUT предназначена для чтения информации с перфоленты по одному байту.

Структура подпрограммы INPUT

Подпрограмма
имя: INPUT
внутренних входов нет
вызываемых программ нет
вызов: CALL INPUT(<символ>)
параметры файла в DCB:
DDNAME=DATA, BLKSIZE=8000, RECFM=U
язык программирования: АССЕМБЛЕР
операционная система ОС ЕС.

Использование подпрограммы INPUT

INPUT можно вызывать, как подпрограмму, оператором FORTRAN-IV CALL.

Параметр подпрограммы. Параметр <символ> задает адрес памяти, в которую передается очередной байт за каждое обращение; он может быть любым идентификатором типа INTEGER*2.

Алгоритм работы подпрограммы. При первом обращении происходит чтение информации и обращение к первому байту считанной информации, при каждом следующем - обращение к очередному байту. Обход открытия файла осуществляется после анализа поля DCBOFLGS таблицы DSCB.

Описание внешнего файла. Конец записи на перфоленте должен быть отмечен двумя пробивками EOR. Символы EOR не входят в запись, эти символы должны быть набраны на перфоленточной станции. В DD-операторе необходимо указать параметр UNIT и, если длина записи больше 8000 байтов, - соответствующий параметр BLKSIZE в DCB.

Пример вызова подпрограммы

```
// EXAMPLE      JOB HUSER, NAME
// FORTRN       EXEC PROC=FORTGCLD
// FORT.SYSIN   DD *
                INIEGER * 2 NUMBER
                DO 1 I = 1,256
                CALL INPUT(NUMBER)
                WRITE (6,100) NUMBER
100  FORMAT(1X, I4)
```

```
1  CONTINUE
   STOP
   END
/*
// GO.SYSLIB   DD
               DD DSN=USR1.PGMLIB, DISP=SHR
// GO.DATA    DD UNIT=007
//
```

ПРОГРАММЫ ИНТЕРПРЕТАЦИИ БАЙТОВ BYTINT И DECINT

При обработке информации с экспериментальных установок часто необходимо интерпретировать данные нестандартным для языка FORTRAN-IV образом. Наиболее распространенными случаями являются интерпретация содержимого одного байта как двоичного представления целого числа и интерпретация содержимого группы байтов как двоично-десятичного представления /в виде тетрад/ целого числа. Язык FORTRAN-IV не имеет прямых средств для такого рода интерпретации, хотя она возможна, но очень громоздка. Наиболее простым методом является использование для этих целей специальных подпрограмм.

ПОДПРОГРАММА ИНТЕРПРЕТАЦИИ СОДЕРЖИМОГО ОДНОГО БАЙТА КАК ДВОИЧНОЕ ПРЕДСТАВЛЕНИЕ ЦЕЛОГО ЧИСЛА BYTINT

Структура подпрограммы BYTINT

Подпрограмма
имя программы: BYTINT
внутренних входов нет
вызываемых подпрограмм нет

вызов: CALL BYTINT(<адрес массива байтов>,
<адрес массива чисел>)

Язык программирования: АССЕМБЛЕР

Использование подпрограммы BYTINT

Первый параметр - адрес массива байтов - указывает на любой массив длиной в шесть байтов. Эти байты последовательно преобразуются в целые числа типа INTEGER*2 и являются первыми шестью элементами массива типа INTEGER*2, адрес которого указывается вторым параметром. Первый и второй параметры могут быть любыми идентификаторами, описанными в вызывающей программе.

ПОДПРОГРАММА ИНТЕРПРЕТАЦИИ СОДЕРЖИМОГО ГРУППЫ БАЙТОВ КАК ДВОИЧНО-ДЕСЯТИЧНОЕ ПРЕДСТАВЛЕНИЕ ЦЕЛОГО ЧИСЛА DECINT

Структура подпрограммы DECINT

Подпрограмма

имя программы: DECINT

внутренних входов нет

вызываемых программ нет

вызов: CALL DECINT (<адрес массива байтов>,
<целочисленная переменная>,
<адрес первой тетрады>,
<число тетрад>).

язык программирования: АССЕМБЛЕР

операционная система: ОС/ЕС

Использование подпрограммы DECINT

Первый параметр - адрес массива байтов - указывает на любой массив длиной в шесть байтов. Из этого массива указанием адреса первой тетрады и числа тетрад выбирается группа тетрад, которая интерпретируется

как двоично-десятичное представление числа. Это число преобразуется в целое типа INTEGER*4 и присваивается переменной типа INTEGER*4, указанной во втором параметре. В шестибайтовом массиве тетрады адресуются слева направо начиная с нуля. Таким образом, адрес первой тетрады может иметь значение от 0 до 11. Число тетрад не должно превышать 8. Адрес шестибайтового массива может быть любым определенным в вызывающей программе идентификатором. Вторым параметром, указывающим на целочисленную переменную, должен быть идентификатор переменной типа INTEGER*4. Адрес первой тетрады и число тетрад - третий и четвертый параметры - указываются либо константой, либо идентификатором переменной типа INTEGER*4.

Пример вызова подпрограмм BYTINT и DECINT:

```
// EXAMPLE JOB HUSER, NAME
//          EXEC FORSTCLD
// FORT.SYSIN DD *
                INTEGER*2 B4WORD(3)
                INTEGER*2 MINT(6)
                INTEGER*4 (NUMB, LENGTH)
                .
                .
                .
                CALL BYTINT(B4WORD,MINT)
                .
                .
                .
                LENGTH = 3
                CALL DECINT (B4WORD, NUMB,4, LENGTH)
                .
```

END

/*

// GO.SYSLIB DD

// DD DSN=USR1.PGMLIB,DISP=SHR

//

*Рукопись поступила в издательский отдел
28 сентября 1978 года.*