



ОБЪЕДИНЕННЫЙ
ИНСТИТУТ
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ
ДУБНА

4626/2-80

22/9-80

11-80-320

Ф.Булла

МЕТОДИКА ПРОГРАММИРОВАНИЯ
ДЛЯ ВИДЕОТЕРМИНАЛА,
СОСТОЯЩЕГО ИЗ ГРАФИЧЕСКОГО ДИСПЛЕЯ
ГД-71 И ЭВМ KRS-4201
И СВЯЗАННОГО С ЭВМ ЕС-1040

Направлено в "Wissenschaftliche Zeitschrift der TU Dresden"

1980

Программирование для графического видеоконкомплекса ГД-71/ KRS-4201 осуществляется на основе Графической Интерактивной Системы /ГИС/. При использовании комплекса как графического периферийного устройства базовой ЭВМ ЕС-1040 программист вынужден иметь дело с двумя синхронно работающими программами, одна из которых /на малой ЭВМ/, обеспечивает работу дисплея, другая /ЕС-1040/ выполняет решение проблемной задачи и осуществляет связь с малой ЭВМ.

Предлагаемый метод дает возможность полного переноса процесса программирования на базовую ЭВМ.

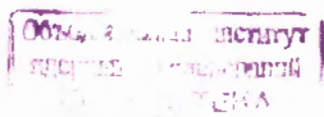
Подпрограммы, работающие в программе пользователя, содержат в себе необходимые для малой ЭВМ программы в форме загрузочных модулей. Постоянно работающий на малой ЭВМ специальный загрузчик принимает такой модуль и организует его работу.

Преимущества предлагаемого метода очевидны: он, прежде всего, повышает вычислительный комфорт, так как все функции видеоконкомплекса, включая интерактивную работу, выполняются "проблемной" программой; кроме того, этот метод существенно экономит память малой ЭВМ, так как там теперь находится загрузочный модуль.

1. МЕТОД "ЭКСПОРТ"

Метод "Экспорт" /EXTERNE PROGRAMMORGANISATION DES TERMINALS/ основывается на том, что работа малой ЭВМ в прикладных задачах организована через программу в ЕС ЭВМ. Эта программа передает в малую ЭВМ только ту свою часть, которая должна выполняться в данный момент /рис.1/. Для малой ЭВМ принципиально необходима программа, обеспечивающая работу видеоконкомплекса, которая вместе с операционной системой формирует дисплейный файл, организует обработку прерываний от периферийных устройств дисплея и делает возможной связь малой ЭВМ с базовой ЕС ЭВМ.

Программа для малой ЭВМ состоит из постоянной части - импортирующей программы /рис.2/, выполняющей роль специализированного загрузчика, и переменной части, которая содержит переданный из базовой ЭВМ материал, предназначенный для представления на экране.



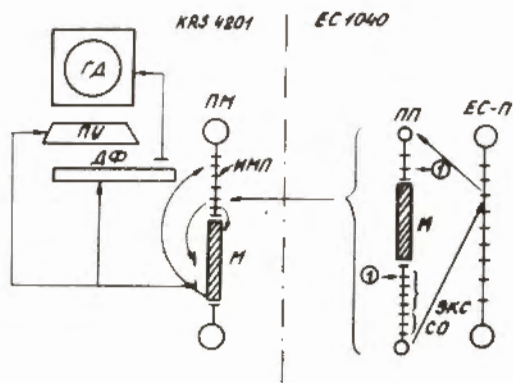


Рис. 1. Основная схема метода "Экспорт": ГД - графический дисплей, ПУ - периферийные устройства дисплея, ДФ - дисплейный файл, ПМ - программа, работающая в малой ЭВМ, ЕС-П - программа, работающая в ЭВМ ЕС-1040, ПП - подпрограмма в ЕС-П, ЭКС - экспортирующая программа, ИМП - импортирующая программа, М - передаваемый загрузочный модуль, СО - специальные операции.

Программное использование графического терминала через ЕС ЭВМ реализуется с помощью подпрограмм, вызываемых на языке ФОРТРАН. Каждая подпрограмма /рис. 3/ состоит, в свою очередь, из экспортирующей части, переменной передаваемой части и части, служащей для специальных прикладных операций, выполняемых ЕС ЭВМ. Передаваемая часть записана в форме загрузочного модуля для малой ЭВМ.

Во время выполнения такой программы экспортирующая часть передает загрузочный модуль в малую ЭВМ. Постоянно работающая импортирующая часть малой ЭВМ принимает загрузочный модуль и передает его в определенную область памяти. Потом управление передается от импортирующей программы на начальный адрес переменной части подпрограммы, которая перед этим была передана малой ЭВМ в форме загрузочного модуля. После выполнения переменной части цикл возвращается к импортирующей подпрограмме, которая готова к приему следующего загрузочного модуля.

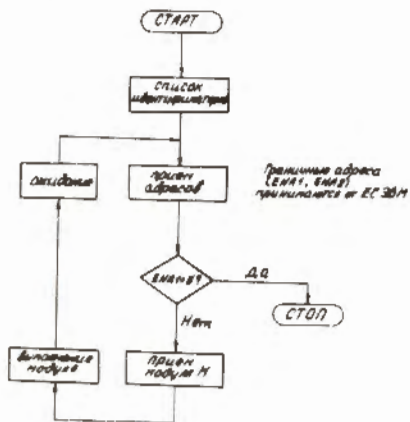


Рис. 2. Импортирующая программа.

Каждая переменная часть имеет одинаковые начальные адреса в области загрузки в малой ЭВМ и одинаковые пусковые адреса. Так как загрузочные модули имеют различную длину, то экспортирующая программа перед передачей сообщает конечный адрес области загрузки импортирующей программе.

Процесс визуализации сводится к последовательному выполнению переданных из ЕС ЭВМ в малую ЭВМ переменных частей подпрограмм.

2. ГЕНЕРАЦИЯ ПОДПРОГРАММ

Разработана библиотека подпрограмм ЕС ЭВМ, которые реализуют все основные операции графического терминала. Эти подпрограммы, как правило, написаны в форме FORTRAN-SUBROUTINE /рис. 4/. Загрузочный модуль для малой ЭВМ представлен в виде последовательности чисел типа INTEGER, размещаемой после оператора DATA в поле начальных значений. Загрузочные модули разрабатываются с использованием малой ЭВМ. Процесс разработки модулей для библиотеки сводится к следующим операциям. Вначале формируется требуемое изображение на экране дисплея под управлением системы ГИС на языке малой ЭВМ. Массивы параметров, описывающих созданное изображение, размещаются по стандартному начальному адресу. С помощью специальной программы массивы передаются из малой ЭВМ в базовую ЕС ЭВМ и перфорируются в необходимом формате /INTEGER/. Таким образом создается библиотека наиболее употребительных изображений. Эту работу выполняет квалифицированный программист, хорошо знакомый с языками ЭВМ KRS-4201, освобождая пользователей от рутинной работы. Полученные перфокарты могут быть включены в колоду перфокарт с программой пользователя и расположены после оператора DATA. Допустимы и другие стандартные способы включения полученных модулей в прикладную программу ЕС ЭВМ. При выполнении модуля используется поле значений параметров, которые задаются программистом и передаются в малую ЭВМ вслед за модулем. Вводя параметры в малую ЭВМ, можно варьировать работу модуля.

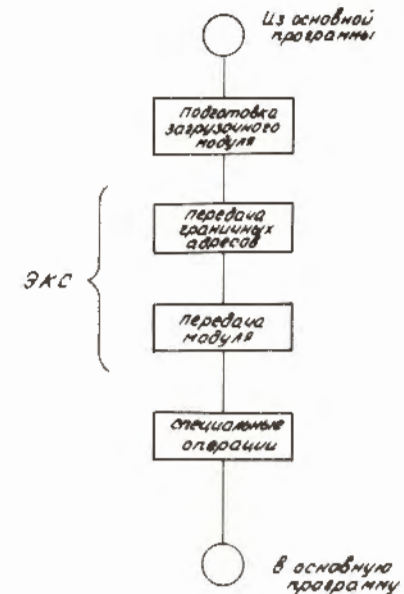


Рис. 3. Подпрограмма для ЕС ЭВМ.


```

C      BEISPIELPROGRAMM  EXPORTO4  NETZGITTER
C
      INTEGER*2  GH(4),GV(4),IPX(2),IPY(2)
      DATA  GH/100,200,900,200/
      DATA  GV/100,200,100,800/
C  ES 1040 - KRS 4201  OPEN
      CALL FDOPEN
      N=1
      IPX(1)=GH(1)
      IPY(1)=GH(2)
      IPX(2)=GH(3)
      IPY(2)=GH(4)
      DO 1  I=1,4
      CALL LINZUG(2,IPX,IPY,N)
      IPY(1)=IPY(1)+200
1  IPY(2)=IPY(2)+200
      IPX(1)=GV(1)
      IPY(1)=GV(2)
      IPX(2)=GV(3)
      IPY(2)=GV(4)
      DO 2  I=1,5
      CALL LINZUG(2,IPX,IPY,N)
      IPX(1)=IPX(1)+200
2  IPX(2)=IPX(2)+200
      CALL KOPEND
      CALL FDCLOS
C  ES 1040 - KRS 4201  CLOSE
      END

```

Рис. 5. Пример: "прямоугольная сетка".

храняет все функции видеоконтекста и его периферии. При его использовании освобождается часть оперативной памяти малой ЭВМ, что позволяет увеличить длину дисплейного файла. Кроме того, применение "Экспорта" позволяет легко перейти к другим графическим системам, как PLOT, ОКР, созданным для графопостроителя ЕС-7054 и известной системы GD3 /ЦЕРН/.

```

C      BEISPIELPROGRAMM  EXPORTO7  IMPULSLINIEN
C
      INTEGER*2  IPFX(16),IP1Y(16),IP2Y(16)
      DATA  IPFX/100,240,240,360,360,440,440,560,560,
      *      640,640,760,760,900/
      DATA  IP1Y/560,560,680,680,560,560,680,680,560,
      *      560,680,680,560,560/
      DATA  IP2Y/440,440,320,320,440,440,320,320,440,
      *      440,320,320,440,440/
      CALL FDOPEN
      N=1
      CALL LINZUG(14,IPFX,IP1Y,N)
      CALL LINZUG(14,IPFX,IP2Y,N)
      CALL FOKFUT(1,31,1)
      CALL KOPEND
      CALL FDCLOS
      END
C
C      BEISPIELPROGRAMM  EXPORTO8  SINUSKURVEN
C
      INTEGER*2  IPKTX(1024),IPKTY(1024)
      CALL FDOPEN
      DO 2  NF=1,2
      IX=0
      P=6.2832/1023
      DO 1  I=1,1024
      IPKTX(I)=IX
      X=P*IX
      Y=SIN(NF*X)
      IPKTY(I)=200*Y+500
1  IX=IX+1
      NI=NF
2  CALL LINZUG(1024,IPKTX,IPKTY,NI)
      DO 3  NF=1,2
3  CALL FOKFUT(1,31,NF)
      CALL KOPEND
      CALL FDCLOS
      END

```

Рис. 6. Пример: "меандр и синусоида".

```

C      BEISPIELPROGRAMM  EXPORT09  MENUE-ELEMENTE
C
INTEGER*2 IPK1X(5),IPK1Y(5),IPK2X(4),IPK2Y(4),IPK3X(4)
INTEGER*2 IPK3Y(4),IPK4X(4),IPK4Y(4),IPK5X(4),IPK5Y(4)
DATA IPK1X/100,150,150,100,100/
DATA IPK1Y/900,900,950,950,900/
DATA IPK2X/100,150,125,100/
DATA IPK2Y/700,700,750,700/
DATA IPK3X/100,100,150,150/
DATA IPK3Y/500,550,550,500/
DATA IPK4X/100,150,100,150/
DATA IPK4Y/300,300,350,350/
DATA IPK5X/100,150,150,125/
DATA IPK5Y/100,100,150,150/
N=1
C  ES 1040 - KRS 4201  OPEN
  CALL FDOPEN
100 CALL FUTAST(NPT)
  M=NPT
  GOTO(1,2,3,4,5,6),M
  1 NAIT=N
  CALL LINZUG(5,IPK1X,IPK1Y,NAIT)
  N=NAIT
  CALL POKPUT(0,31,N-1)
  GOTO 100
  2 NAIT=N
  CALL LINZUG(4,IPK2X,IPK2Y,NAIT)
  N=NAIT
  CALL POKPUT(0,31,N-1)
  GOTO 100
  3 NAIT=N
  CALL LINZUG(4,IPK3X,IPK3Y,NAIT)
  N=NAIT
  CALL POKPUT(0,31,N-1)
  GOTO 100
  4 NAIT=N
  CALL LINZUG(4,IPK4X,IPK4Y,NAIT)
  N=NAIT
  CALL POKPUT(0,31,N-1)
  GOTO 100
  5 NAIT=N
  CALL LINZUG(4,IPK5X,IPK5Y,NAIT)
  N=NAIT
  CALL POKPUT(0,31,N-1)
  GOTO 100
  6 CONTINUE
  CALL KOPEND
C  ES 1040 - KRS 4201  CLOSE
  CALL PDCLOS
  END

```

Рис. 7. Пример: различные элементы "меню".

ЛИТЕРАТУРА

1. Monjau D., Ortleb R. Dokumentation Digital - Geometrischer Arbeitsplatz GD 71 - KRS 4201 - ESER Seite Kleinrechner TU, Dresden 1978.
2. Grossmann R., Monjau D. Dokumentation Terminal - Basiszugriffsmethode TEBAM TU Dresden, 1979.
3. ОКР. Общая программа для черчения - Дигиграф D3G.1. Прага, 1976.
4. Zeichensoftware PLOT im OS/ES 4.1 ROBOTRON - Dokumentation. Dresden, 1978.
5. Miller R. GD3 CERN COMPUTER CENTRE, Program Library, 1976.

Рукопись поступила в издательский отдел
9 июня 1980 года.