

А. А. Бирман

БИБЛИОТЕЧНЫЙ МОДУЛЬ WINDOWHANDLER

Библиотечный модуль WINDOWHANDLER реализует работу с окнами на алфавитно-цифровом дисплее. Модуль экспортирует объекты, позволяющие задать на экране дисплея прямоугольные области произвольного размера, – окна и осуществлять в них операции ввода-вывода.

Окно задается координатами на экране дисплея, его высотой и шириной. Процедура OPENWINDOW создает дескриптор окна и, если нужно (указано в параметрах), печатает шапку. Ссылка на дескриптор помещается в переменную типа WINDOW, указанную в параметрах процедуры.

Модуль предоставляет следующие возможности работы с окнами:

- очистить окно;
- очистить строку окна с указанной позиции;
- получить текущую позицию окна;
- установить нужную текущую позицию окна;
- вывести символ, строку, целое типа CARDINAL, начиная с текущей позиции окна;
- ввести символ, целое типа CARDINAL, начиная с текущей позиции окна;
- закрыть окно (уничтожить дескриптор).

Процедуры SETMARK и CLRD используют зависящие от типа дисплея команды управления. Для настройки модуля на тип дисплея, отличный от 15ИЭ0013, необходимо внести корректировку в эти процедуры.

Пользователь может задать несколько режимов вывода информации в окно. Для этого необходимо установить соответствующие признаки в слове состояния окна. В приложениях приведены тексты программ.

ПРИЛОЖЕНИЕ 1

```
DEFINITION MODULE WINDOWHANDLER; (*B, 1986, 02*)
EXPORT QUALIFIED
  WINDOW, WINDOWDESCRIPTOR, OPENWINDOW, WWRITEARRAY, STATE, SETSTATE,
  GETPOS, SETPOS, BLOCKDESCRIPTOR, CLRLINE, CLRWINDOW,
  SETMARK, NL, NCH, CLRD, CLOSEWINDOW, WREAD, WREADCARD,
  WWRITE, WWRITELINE, WWRITECARD, WWRITELN, DONE, NEXTCHAR;

CONST
  NL=24; NCH=80;

TYPE

  STATE=(OPEN, NEWLINE, OVER, PREDEL);
  SETSTATE=SET OF STATE; (* СЛОВО СОСТОЯНИЯ ОКНА *)
  (* OPEN - ОКНО ОТКРЫТО;
  NEWLINE- ПО ДОСТИЖЕНИЕМ МАРКЕРОМ ГРАНИЦЫ ОКНА,
  ВЫВОД ИНФОРМАЦИИ ПРОДОЛЖИТЬСЯ НА
  ОЧЕРЕДНОЙ СТРОКЕ;
  OVER - БИТ УСТАНАВЛИВАЕТСЯ, ЕСЛИ НА ГРАНИЦЕ
  ОКНА МАРКЕР 'ЗАЛИП';
  PREDEL - ПРИ ПЕРЕХОДЕ НА НОВУЮ СТРОКУ,
  ЕЕ СОДЕРЖИМОЕ СТИРАЕТСЯ.
  *)

  BLOCKDESCRIPTOR=RECORD X,Y,W,H: CARDINAL END;

  WINDOW=POINTER TO WINDOWDESCRIPTOR;

  WINDOWDESCRIPTOR =
    RECORD
      WSTATE: SETSTATE; (* СЛОВО СОСТОЯНИЯ ОКНА *)
      CPOS, CLINE: CARDINAL; (* ТЕКУЩИЕ КООРДИНАТЫ ОКНА *)
      BLK: BLOCKDESCRIPTOR; (* КООРДИНАТЫ И РАЗМЕР ОКНА *)
      HEAD: ARRAY [0..NCH-1] OF CHAR; (* НАЗВАНИЕ ШАПКИ ОКНА *)
    END;
```

```

VAR DONE:BOOLEAN; (* DONE='ВВЕДЕНО ЧИСЛО' *)
NEXTCHAR:CHAR; (* ПЕРВАЯ НЕ ЦИФРА ПРИ ВВОДЕ ЦЕЛЫХ *)

PROCEDURE SETMARK ( LINE,POS:CARDINAL );
(*МАРКЕР НА LINE-ЛИНИЮ И POS-ПОЗИЦИЮ ЭКРАНА
( LINE-[0..23], POS-[0..79] ) *)

PROCEDURE OPENWINDOW ( VAR W:WINDOW; POS,LINE,WIDTH,HIGH:CARDINAL;
NAME:ARRAY OF CHAR;
VAR DONE:BOOLEAN; HEAD:BOOLEAN);
(* POS-АБСОЛЮТНАЯ КООРДИНАТА КРАЙНЕЙ ЛЕВОЙ ПОЗИЦИИ,
LIN-АБС.КООРДИНАТА ВЕРХНЕЙ СТРОКИ,
WIDTH-ШИРИНА ОКНА,
HIGH-ЧИСЛО ЛИНИЙ В ОКНЕ,ВКЛЮЧАЯ ШАПКУ,
NAME-ПОСТОЯННАЯ ШАПКА,
DONE='ОКНО ОТКРЫТО',
HEAD='НУЖНО ПЕЧАТАТЬ ШАПКАУ'*)

PROCEDURE CLOSEWINDOW ( VAR W:WINDOW );

PROCEDURE SETPOS( W:WINDOW; LINE,POS:CARDINAL );
(*УСТАНОВИТЬ ТЕКУЩУЮ ПОЗИЦИЮ ОКНА-(LINE,POS) *)

PROCEDURE GETPOS( W:WINDOW; VAR LINE,POS:CARDINAL );
(*ПОЛУЧИТЬ ТЕКУЩИЕ КООРДИНАТЫ ОКНА *)

PROCEDURE CLRLINE( W:WINDOW; LINE,POS:CARDINAL );
(*ОЧИСТИТЬ В ОКНЕ СТРОКУ LINE С ПОЗИЦИИ POS *)

PROCEDURE CLRWINDOW ( W:WINDOW ); (* ОЧИСТИТЬ ОКНО *)

PROCEDURE WWRITE ( WIN:WINDOW; CH:CHAR );
(* ЗАПИСАТЬ СИМВОЛ В ТЕКУЩЕЙ ПОЗИЦИИ ОКНА *)

PROCEDURE WWRITELINE ( WIN:WINDOW; STRING:ARRAY OF CHAR );
(* ЗАПИСАТЬ СТРОКУ С ТЕКУЩЕЙ ПОЗИЦИИ ОКНА *)

PROCEDURE CLRD ( LINE:CARDINAL );
(* ОЧИСТИТЬ ДИСПЛЕЙ НАЧИНАЯ СО СТРОКИ LINE *)

PROCEDURE WWITELN ( WIN:WINDOW );
(* ПЕРЕЙТИ НА ДРУГУЮ СТРОКУ ОКНА *)

PROCEDURE WREAD ( WIN:WINDOW; VAR CH:CHAR );
(* УСТАНОВИТЬ МАРКЕР В ТЕКУЩУЮ ПОЗИЦИЮ ОКНА
И ПРИНЯТЬ СИМВОЛ,ЗАТЕМ ЕГО ВЫСВЕТИТЬ *)

PROCEDURE WREADCARD ( WIN:WINDOW; VAR C:CARDINAL );
(* МАРКЕР УСТАНОВЛИВАЕТСЯ В
ТЕКУЩУЮ ПОЗИЦИЮ ОКНА И ВВОДИТСЯ ЦЕЛОЕ*)

PROCEDURE WWRITECARD ( WIN:WINDOW; C,N:CARDINAL );
(* ВЫВЕСТИ ЦЕЛОЕ-'C' С ТЕКУЩЕЙ ПОЗИЦИИ ОКНА С 'N' ЗНАКАМИ *)

END WINDOWHANDLER.

```

```

IMPLEMENTATION MODULE WINDOWHANDLER; (*T- B,1986,05 *)

FROM TTIO IMPORT WPRTSTRINH, WPRT, READ, SETMODE;
FROM STORAGE IMPORT ALLOCATE, DEALLOCATE;
(*CONST NL=24; NCH=80; *)

PROCEDURE SETMARK(LINE, POS: CARDINAL);
BEGIN
  WPRT(33C); WPRT(131C);
  WPRT(CHAR(40B+LINE)); WPRT(CHAR(40B+POS));
END SETMARK;

PROCEDURE OPENWINDOW(VAR WIN: WINDOW; X, Y, WI, HI: CARDINAL;
                     NAME: ARRAY OF CHAR;
                     VAR DONE: BOOLEAN;
                     KAP: BOOLEAN);

  VAR I: CARDINAL;
BEGIN
  NEW(WIN);
  IF (X > NCH-1) OR (Y > NL-1) THEN DONE := FALSE; RETURN END;
  WITH WIN^ DO
    BLK.X := X;
    IF (KAP) THEN BLK.Y := Y+2 (*ШАПКА ОКНА*)
    ELSE BLK.Y := Y END;
    IF X+WI <= NCH-1 THEN BLK.W := WI ELSE BLK.W := NCH-1-X END;
    IF Y+HI <= NL-1 THEN BLK.H := HI ELSE BLK.H := NL-1-Y END;
    IF KAP THEN I := 0;
      WHILE (I <= HIGH(NAME)) AND (NAME[I] # 0C) AND (I <= WI) DO
        HEAD[I] := NAME[I]; INC(I)
      END;
    (*ЗАПИШЕМ ЗАГОЛОВOK ОКНА*)
    SETMARK(Y, X); I := 0;
    WHILE (I <= HIGH(NAME)) AND (NAME[I] # 0C) & (I <= WI) DO
      WPRT(NAME[I]); INC(I)
    END;
    SETMARK(Y+1, X); I := 0;
    WHILE (I <= HIGH(NAME)) & (NAME[I] # 0C) & (I <= WI) DO
      WPRT(' '); INC(I)
    END;
  END; (*IF*)
  CPOS := 0; CLINE := 0;
  DONE := TRUE;
  WSTATE := SETSTATEWOPEN, NEWLINE, PREDELW;
END; (*WITH*)
END OPENWINDOW;

PROCEDURE CLOSEWINDOW(VAR WIN: WINDOW);
BEGIN
  DISPOSE(WIN)
END CLOSEWINDOW;

PROCEDURE SETPOS(WIN: WINDOW; LINE, POS: CARDINAL);
VAR L, P: CARDINAL;
BEGIN
  IF WIN = NIL THEN RETURN END;
  WITH WIN^ DO
    IF OPEN IN WSTATE THEN

```

```

IF LINE>BLK.H THEN L:=BLK.H ELSE L:=LINE END;
IF POS>BLK.W THEN P:=BLK.W ELSE P:=POS END;
SETMARK(BLK.Y+L,BLK.X+P);
CLINE:=L;CPOS:=P
END
END(*WITH*)
END SETPOS;

PROCEDURE GETPOS( WIN:WINDOW; VAR LINE,POS:CARDINAL);
BEGIN IF WIN=NIL THEN RETURN END;
WITH WIN^ DO
  IF OPEN IN WSTATE THEN LINE:=CLINE; POS:=CPOS END
END
END GETPOS;

PROCEDURE CLRLINE( WIN:WINDOW; LINE,POS:CARDINAL);
VAR I:CARDINAL;
BEGIN IF WIN=NIL THEN RETURN END;
WITH WIN^ DO
  IF OPEN IN WSTATE THEN
    IF LINE> BLK.H THEN RETURN END;
    IF POS> BLK.W-1 THEN RETURN END;
    SETMARK(BLK.Y+LINE,BLK.X+POS);
    FOR I:=POS TO BLK.W-1 DO WPUTE(' ') END;
  END
END(*WITH*)
END CLRLINE;

PROCEDURE CLRWINDOW( WIN:WINDOW);
VAR I:CARDINAL;
BEGIN IF WIN=NIL THEN RETURN END;
WITH WIN^ DO
  IF OPEN IN WSTATE THEN
    FOR I:=0 TO BLK.H-1 DO CLRLINE(WIN,I,0) END;
    CPOS:=0;CLINE:=0
  END
END(*WITH*)
END CLRWINDOW;

PROCEDURE WWRITE( WIN:WINDOW; CH:CHAR);
CONST CR=150;LF=120;
VAR CARRY:BOOLEAN;
BEGIN
IF WIN=NIL THEN RETURN END;
WITH WIN^ DO
  IF OPEN IN WSTATE THEN
    IF CH=CR THEN CPOS:=0; EXCL(WSTATE,OVER); RETURN
    ELIF CH=LF THEN CLINE:=(CLINE+1)MOD(BLK.H);
      IF PREDEL IN WSTATE THEN CLRLINE(WIN,CLINE,0) END;
      RETURN
    ELSE CARRY:=(CPOS+1)DIV(BLK.W)>0;
      IF NOT(OVER IN WSTATE) THEN
        SETMARK(BLK.Y+CLINE,BLK.X+CPOS); WPUTE(CH);
        IF NOT CARRY THEN CPOS:=(CPOS+1)MOD(BLK.W) END;
        IF CARRY THEN
          IF (NEWLINE IN WSTATE) THEN
            WPUTE(7C); CLINE:=(CLINE+1)MOD(BLK.H); CPOS:=0;
            IF PREDEL IN WSTATE THEN CLRLINE(WIN,CLINE,0) END;

```

```

        ELSE
            INCL (WSTATE,OVER)
        END
    END
END
END
END(*IF*)
END(*WITH*)
END WWRITE;

PROCEDURE WWRITELN ( WIN:WINDOW );
BEGIN IF WIN=NIL THEN RETURN END;
    WWRITE (WIN,15C);WWRITE (WIN,12C)
END WWRITELN;

PROCEDURE WWRITELINE ( WIN:WINDOW; LIN:ARRAY OF CHAR );
BEGIN
    WWRITEARRAY (WIN,LIN)
END WWRITELINE;

PROCEDURE WWRITEARRAY ( WIN:WINDOW; VAR LIN:ARRAY OF CHAR );
    VAR I:CARDINAL;
BEGIN I:=0;
    WHILE (I<=HIGH(LIN))&(LINC(I)≠0C) DO WWRITE (WIN,LINC(I));INC(I) END
END WWRITEARRAY;

PROCEDURE CLRD ( LIN:CARDINAL );
    VAR I,J:CARDINAL;
BEGIN
    SETMARK (LIN,0);
    WPATE (33C);WPATE (112C)
END CLRD;

PROCEDURE WREAD ( WIN:WINDOW; VAR CH:CHAR );
BEGIN
    IF WIN=NIL THEN RETURN END;
    WITH WIN^ DO
        SETMARK (BLK.Y+CLINE,BLK.X+CPOS)
    END;(*BNTX*)
    READ (CH);
    WWRITE (WIN,CH)
END WREAD;

PROCEDURE WWRITECARD ( WIN:WINDOW; X,N: CARDINAL );
    VAR I:CARDINAL;
        A:ARRAY [0..6] OF CARDINAL;
BEGIN I:=0;
    REPEAT A(I):=X MOD 10; X:=X DIV 10;INC(I)
    UNTIL X=0;
    WHILE N>I DO DEC(N); WWRITE (WIN,' ') END;
    REPEAT DEC(I); WWRITE (WIN,CHAR (A(I)+60B)) UNTIL I = 0
END WWRITECARD;

PROCEDURE WREADCARD ( WIN:WINDOW ;VAR X:CARDINAL);
VAR CH:CHAR;X0:CARDINAL;
BEGIN
    IF WIN=NIL THEN RETURN END;
    IF OPEN IN WIN^.WSTATE THEN
        LOOP WREAD (WIN,CH);
            IF CH)' ' THEN

```

```
IF (CH)='0') & (CH<='9') THEN
  X0:=CARDINAL(CH)-60B; WREAD(WIN,CH);
  WHILE (CH)='0') & (CH<='9') DO
    X0:=10*X0+(CARDINAL(CH)-60B); WREAD(WIN,CH)
  END; (*WHILE*)
  X:=X0;
  DONE:=TRUE;
ELSE DONE:=FALSE;
END;
NEXTCHAR:=CH; EXIT
END (*IF*)
END (*LOOP*)
END (*IF*)
END WREADCARD;

BEGIN
SETMODE(0, TRUE)
END WINDOWHANDLER.
```
