

```

'*****
'*
'-----
'* Task : Demonstrates direct access to video RAM.
'-----
'* Author : Michael Tischer
'* Developed on : 05/06/91
'* Last update : 02/05/92
'*****
DECLARE SUB InitDPrint ()
DECLARE SUB Demo ()
DECLARE SUB DPrint (Column%, ScRow%, DColr%, StrOut AS STRING)

CONST NORMAL = &H7 'Define character attributes
CONST HIINT = &HF 'on monochrome video card
CONST INVERSE = &H70
CONST UNDERSCORED = &H1
CONST BLINKING = &H80

CONST BLACK = &H0 'Color attributes on color video card
CONST BLUE = &H1
CONST GREEN = &H2
CONST CYAN = &H3
CONST RED = &H4
CONST VIOLET = &H5
CONST BROWN = &H6
CONST LGHTGRAY = &H7
CONST DARKGRAY = &H8
CONST LGHTBLUE = &H9
CONST LGHTGREEN = &HA
CONST LGHTCYAN = &HB
CONST LGHTRED = &HC
CONST LGHTVIOLET = &HD
CONST YELLOW = &HE
CONST WHITE = &HF

DIM SHARED VSeg AS LONG 'Segment address of video RAM

CALL InitDPrint 'Initialize DPrint information
CALL Demo 'Demonstrate DPrint
END

'*****
'* Demo : Demonstrates DPrint routine.
'* Input : None
'* Output : None
'*****
SUB Demo

DIM Column AS INTEGER 'Display column
DIM ScRow AS INTEGER 'Display row
DIM DColr AS INTEGER 'Display attribute

RANDOMIZE TIMER 'Initialize random generator

IF VSeg = &HB800 THEN 'Color adapter connected?
CLS 'Clear screen
CALL DPrint(22, 0, WHITE, " DVIB - (c) 1988, 92 by Michael Tischer ")
DO
Column = INT(76 * RND) 'Select random columns
ScRow = INT(22 * RND) + 1 'Select random rows
DColr = INT(14 * RND) + 1 'Select random color
CALL DPrint(Column, ScRow, DColr, "ûûû") 'Display block
LOOP UNTIL INKEY$ <> " " 'Repeat until user presses a key
ELSE 'Monochrome adapter connected
CLS 'Clear screen
CALL DPrint(22, 0, INVERSE, " DVIB - (c) 1988, 92 by Michael Tischer ")
DO
Column = INT(76 * RND) 'Select random column
ScRow = INT(22 * RND) + 1 'Select random row
SELECT CASE INT(4 * RND) 'Select random character attribute
CASE 0
DColr = NORMAL
CASE 1
DColr = HIINT

```

```

CASE 2
DClr = INVERSE
CASE 3
DClr = BLINKING OR INVERSE          'For maximum visibility
END SELECT
CALL DPrint(Column, ScRow, DClr, "ÛÛ")          'Display block
LOOP UNTIL INKEY$ <> " "          'Repeat until user presses a key
END IF
END SUB

'*****
'* DPrint : Writes a string directly to video RAM. *
'* Input : - Column : The display column *
'*          - ScRow : The display row *
'*          - DClr : Character color (attribute) *
'*          - StrOut : The string to be displayed *
'* Output : None *
'*****
SUB DPrint (Column%, ScRow%, DClr%, StrOut AS STRING)

DIM Offset AS INTEGER          'Offset address of char. should be poked
DIM Counter AS INTEGER          'Loop counter

DEF SEG = &H40          'Segment address of BIOS variable range
Offset = PEEK(&H4E) + PEEK(&H4F) * 256          'Get starting address of page
Offset = Offset + ScRow% * 160 + Column% * 2          'Offset address: 1st char.
DEF SEG = VSeg          'Segment address of video RAM
FOR Counter = 1 TO LEN(StrOut)          'Execute string
POKE Offset, ASC(MID$(StrOut, Counter, 1))          'ASCII code in video RAM
POKE Offset + 1, DClr%          'Color in video RAM
Offset = Offset + 2          'Set offset to next character
NEXT
END SUB

'*****
'* InitDPrint : Gets the segment address for DPrint. *
'* Input : None *
'* Output : The segment address of video RAM through the VSeg *
'*          global variable *
'*****
SUB InitDPrint

DEF SEG = &H40          'Segment address: BIOS variable register
IF PEEK(&H63) + PEEK(&H64) * 256 = &H3B4 THEN          'Monochrome adapter?
VSeg = &HB000          'Video RAM at 8000:0000
ELSE          'Color adapter?
VSeg = &HB800          'Video RAM at B800:0000
END IF
END SUB

```