

```

/*****
*
*          D B L S P C C . C
*
*-----*
*      Task      : Display information about all active Double-
*                  Space drives.
*-----*
*      Author     : MICHAEL TISCHER
*      developed on : 09/23/1993
*      last update  : 04/14/1995
*****/

#include <dos.h>
#include <stdio.h>
#include <process.h>

/*-- type declarations -----*/

typedef unsigned char BYTE;
typedef unsigned int  BOOL;

/*-- constants -----*/

#define TRUE  (0==0)
#define FALSE (0==1)

/*****
* IsDoubleSpaceInstalled: Checks to see if DoubleSpace is installed and
*                        provides information about DoubleSpace
*-----*
* Input  :  pFirstDrive = contains the first device ID (65 = A:)
*            which is available to DoubleSpace
*            pNumDrives  = contains the number of device IDs
*                        reserved for DoubleSpace
*            pVerNum     = contains the internal DoubleSpace
*                        version number
*            pUpperMem   = contains the value TRUE if DoubleSpace
*                        is resident in upper memory
* Output : TRUE if DoubleSpace is installed; otherwise FALSE
*****/

BOOL IsDoubleSpaceInstalled( BYTE *pFirstDrive,
                             BYTE *pNumDrives,
                             BYTE *pVerNum,
                             BOOL *pUpperMem )
{
    union REGS Regs;

    Regs.x.ax = 0x4A11;          /* MUX code for DoubleSpace */
    Regs.x.bx = 0;               /* function number */
    int86( 0x2F, &Regs, &Regs ); /* call multiplexer */

    /*-- Fetch information from processor registers -----*/
    *pFirstDrive = Regs.h.cl;    /* number of the first DoubleSpace drive */
    *pNumDrives = Regs.h.ch;     /* number of DoubleSpace drives */
    *pVerNum = (Regs.x.dx & 0x7FFF); /* internal version number */
    *pUpperMem = (Regs.x.dx & 0x8000) == 0; /* in upper memory? */
    return (Regs.x.ax == 0);
}

/*****
* IsDoubleSpaceDrive: Determines whether a given drive
*                    is a DoubleSpace drive, and returns
*                    information about this drive
*-----*
* Input  :  pDR          = device ID of the drive to be checked
*            (0 = A:, 1 = B: etc)
*            pExchanged  = contains TRUE if it
*                        is a compressed drive which was
*                        exchanged with its host drive
*            pHostNo     = contains the device ID
*                        of the host drive if it is
*                        a DoubleSpace drive
*            pCvfvNo     = contains the number of the CVF file
*                        if it is a DoubleSpace
*                        drive
* Output  : TRUE if it is a DoubleSpace;
*****/

```

```

*               otherwise FALSE                                     *
******/
BOOL IsDoubleSpaceDrive( BYTE bDR,
                        BOOL *pExchanged,
                        BYTE *pHostDr,
                        BYTE *pCvfNo )
{
    BYTE bCvfNo, /* local variables, taken next */
          bHostDr;
    BOOL bExchanged,
          bIsDoubleSpace;

/*-- first assume an uncompressed, non-exchanged drive -----*/
    bHostDr = bDR;
    bExchanged = FALSE;
    bIsDoubleSpace = FALSE;
    bCvfNo = 0;

_asm
{
    mov     ax,4A11h /* call DoubleSpace Function */
    mov     bx,0001h /* 00001H */
    mov     dl,bDR
    int     2Fh
    or      ax,ax /* call successful? */
    jnz     idbende /* no, DoubleSpace not installed */

/*-- call was successful -----*/
    test    bl,80h /* compressed drive? */
    jz      idbHostDr /* no, possibly host drive */

/*-- compressed drive; now determine host drive -----*/
    mov     bIsDoubleSpace,TRUE
    mov     bCvfNo,bh /* note number of the CVF file */

    and     bl,7Fh /* filter out number of the host drive */
    mov     bHostDr,bl /* and note it */

    mov     dl,bl /* call Function 0001H with host */
    mov     ax,4A11h /* drive again */
    mov     bx,0001h
    int     2Fh

    and     bl,7Fh /* filter number of the host drive */
    cmp     bl,bDR /* is the host its own host? */
    mov     bExchanged,TRUE /* assume exchanged drive */
    je      idbend /* exchanged --> idb}e */

    mov     bExchanged,FALSE /* drive is not exchanged */
    mov     bHostDr,bl
    jmp     idbend

/*-- it is an uncompressed host drive -----*/
    idbHostDr:
    and     bl,7Fh /* filter host drive ID */
    cmp     bl,dl /* was the drive exchanged? */
    je      idbend /* no ---> idb}e previous note */

    mov     bExchanged,TRUE /* yes */
    mov     bHostDr,bl /* set true device ID */

idbend:
}; /* ASM */

*pHostDr = bHostDr; /* transfer results to variables of */
*pExchanged = bExchanged; /* the call routine */
*pCvfNo = bCvfNo;
return bIsDoubleSpace;
};

/*-- Variables for the main program -----*/
BYTE i, /* loop counter */
      vernum; /* DoubleSpace version no. */

```

```

BYTE  firstdrive,          /* first DoubleSpace drive */
      numdrive,           /* number of DoubleSpace drives */
      host,               /* receives host drive */
      cvfnr;              /* receives CVF number */
BOOL  isdbl,              /* DoubleSpace drive? */
      uppermem,           /* DoubleSpace in upper memory? */
      Exchanged;          /* exchanged with host drive? */

/*-----*/
/*--- M A I N   P R O G R A M ---*/
/*-----*/

void main( void )
{
    printf( "DBLSPCC.C - (c) 1993,94 by Michael Tischer\n" );
    isdbl = IsDoubleSpaceInstalled( &firstdrive, &numdrive,
                                    &vernum, &uppermem );

    if( !isdbl )
    {
        printf( "DoubleSpace is not installed!\n" );
        exit(1);
        /* quit program */
    };

    /*-- DoubleSpace is installed -----*/
    printf( "DoubleSpace version      : %d\n", vernum );
    printf( "First DoubleSpace drive : %c:\n", firstdrive );
    printf( "Reserved for DoubleSpace : %d drives\n", numdrive );
    printf( "DoubleSpace in upper memory : " );
    printf( "%s\n\n", uppermem ? "Yes" : "No" );

    /*-- Output DoubleSpace drives -----*/
    printf( "Compressed drive is actually CVF file\n" );
    printf( "-- -----\n" );
    for( i = 0; i < 26; i++ )
        /* Run through drives A: through Z: */
    {
        isdbl = IsDoubleSpaceDrive( i, &Exchanged, &host, &cvfno );
        if( isdbl || Exchanged )
        {
            printf( "%c: %s", 'A'+i, isdbl ? "yes" : "no" );

            if( Exchanged )
                printf( " %c: ", 'A'+host );
            else
                printf( " " );

            if( isdbl )
                /* output CVF number for DoubleSpace drives */
                printf( " DBLSPACE.%03d", cvfnr );
            printf( "\n" );
        }
    }
    printf( "\n" );
}

```