

```

/*****
*
*          V O N O F F C . C
*
*-----*
* Task      : Demonstrates video display enable and disable *
*            on EGA and VGA cards.                          *
*-----*
* Author     : Michael Tischer                               *
* Developed on : 08/26/90                                     *
* Last update on : 02/18/92                                 *
*-----*
* (MICROSOFT C)                                             *
* Compilation : CL /AS vonoffc.c                           *
*-----*
* (BORLAND TURBO C)                                         *
* Compilation : Use the integrated development environment   *
*****/

#include <dos.h> /* Add include files */
#include <conio.h>
#include <stdio.h>

#ifdef __TURBOC__ /* Compiling with Turbo C? */
#define CLI()      disable()
#define STI()      enable()
#define outpw( p, w ) outport( p, w )
#define inp
#define outp( p, b ) outportb( p, b )
#define inp( p )    inportb( p )
#endif
#else /* No --> With Quick C or MSC */
#include <conio.h>
#define MK_FP(seg,ofs) ((void far *)\
((unsigned long)(seg) << 16) | (ofs))
#define CLI()      _disable()
#define STI()      _enable()
#endif

/*-- Constants -----*/

#define EV_STATC 0x3DA /* EGA/VGA color status register */
#define EV_STATM 0x3BA /* EGA/VGA mono status register */
#define EV_ATTR 0x3C0 /* EGA/VGA attribute controller */

/*****
*
* ScrOff : Disables the EGA/VGA screen.
*-----*
* Input   : None
*****/

void ScrOff( void )
{
    CLI(); /* Disable interrupts */
    inp( EV_STATC ); /* Reset color status reg */
    inp( EV_STATM ); /* Reset mono status reg */
    outp( EV_ATTR, 0x00 ); /* Mask bit 5 from access */
/* to CRT controller */
    STI(); /* Enable interrupts */
}

/*****
*
* ScrOn : Enables the EGA/VGA screen.
*-----*
* Input   : None
*****/

void ScrOn( void )
{
    CLI(); /* Disable interrupts */
    inp( EV_STATC ); /* Reset color status reg */
    inp( EV_STATM ); /* Reset mono status reg */
    outp( EV_ATTR, 0x20 ); /* Set bit 5 for access */
/* to CRT controller */
    STI(); /* Enable interrupts */
}

/*****/

```

```

* IsEgaVga : Determines whether an EGA or VGA card is installed.      *
**-----**
* Input      : None
* Output     : TRUE if EGA or VGA card, otherwise FALSE
*****/

int IsEgaVga( void )
{
    union REGS Regs;          /* Processor registers for interrupt call */

    Regs.x.ax = 0x1a00;        /* Function 1AH applies to VGA only */
    int86( 0x10, &Regs, &Regs );
    if ( Regs.h.al == 0x1a )    /* Is the function available? */
        return 1;
    else
    {
        Regs.h.ah = 0x12;      /* Call function 12H, */
        Regs.h.bl = 0x10;      /* sub-function 10H */
        int86(0x10, &Regs, &Regs); /* Call video BIOS */
        return ( Regs.h.bl != 0x10 );
    }
}

/*****
* Delay : BIOS induced time delay.
**-----**
* Input   : Delay in seconds
* Output  : None
*****/

void Delay( int pauslen )
{
    unsigned int tico_hi,          /* Time counter */
                tico_lo,
                ticks;
    union REGS inregs,            /* Processor registers */
                outregs;

    ticks = pauslen * 182 / 10;
    inregs.h.ah = 0;              /* Function 00H = Read counter */
    int86( 0x1a, &inregs, &outregs ); /* Get and store time */
    tico_hi = outregs.x.cx;
    tico_lo = outregs.x.dx;

    while ( ticks )              /* Repeat until ticks = 0 */
    {
        int86( 0x1a, &inregs, &outregs ); /* Get time */

        /*-- New tick occurred? -----*/

        if ( tico_hi != outregs.x.cx || tico_lo != outregs.x.dx )
        {
            tico_hi = outregs.x.cx; /* Store new counter value */
            tico_lo = outregs.x.dx;
            --ticks; /* Decrement number of remaining ticks */
        }
    }
}

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

void main( void )                /* Get a key */
{
    int i;                       /* Loop counter */

    for ( i=0; i<25; ++i )       /* Clear screen */
        printf( "\n" );

    printf( "VONOFFC - (c) 1992 by Michael Tischer\n\n" );
    if ( IsEgaVga() )            /* EGA or VGA card? */
    {
        printf( "ATTENTION: Screen will go black in five seconds.\n\n" );
        printf( "Press any key to enable screen again." );
        Delay( 5 );              /* Wait five seconds */
    }
}

```

```

while ( kbhit() )      /* Purge all keys from the keyboard buffer */
    getch();
Scroff();              /* Screen off */
getch();               /* Wait for a key */
Scron();               /* Screen on */
printf( "\n\n\nEnd program\n" );
}
else                   /* No --> No EGA or VGA */
    printf( "Warning: No EGA or VGA card found\n" );
}

```