



Title: Porting Legacy MS-DOS Motion Applications to Windows NT
Products(s): MCAPI
Keywords: DOS, MS-DOS, WIN32, C, C++, console programs
ID#: TN1013
Date: August 18, 1998

Summary

When porting legacy DOS applications to Windows NT it is no longer possible to use DOS interface libraries to communicate with the motion controller. Access to peripherals, including the motion controller, requires an NT device driver.

More Information

Under Windows NT (or more specifically the WIN32 API), it is rather easy to convert 16-bit DOS programs into 32-bit console programs. Console programs are similar to DOS programs in that they do not have a Windows graphical user interface and they run in a "DOS Box". The advantage of console programs is that they may call functions within Windows Dynamic Link Libraries (DLLs). Thus by replacing calls to PMC's DOS Interface Libraries with calls to PMC's Motion Control API (MCAPI) for Windows NT it is possible to preserve the bulk of original DOS program.

To create a Console Program you will need a 32-bit C/C++ compiler (other 32-bit language programs may also support Console Program creation, consult the product documentation for details). You will need to include the standard Windows header file **WINDOWS.H** as well as the **MCAPI.H** header file within your program. Finally, you will need to link your project to the **MCAPI32.LIB** library file (Borland C/C++ users should see *TechNote TN1011* for information regarding LIB file usage with Borland's C/C++).

A simple display position sample:

```
#include <windows.h>
#include <stdio.h>
#include "mcapi.h"

void main( int argc, char* argv[], char* envp[] )
{
    HCTRLR hCtrlr;
    double posit = 0.0;

    if ((hCtrlr = MCOpen( 0, MC_OPEN_BINARY, "" )) > 0) {
        MCGetPositionEx( hCtrlr, 1, &posit );
        printf( "Position = %lf\n", posit );
        MCClose( hCtrlr );
    }
}
```

Console programs may also be run on Windows 95 / 98 as both support the WIN32 API. They are an ideal way to create simple motion test programs and utilities.