

Command line for CP/M

This program writes a initial command line into the on-disk CP/M. This command is always executed on both warm and cold boots. The program will also erase a previous command and display the current command. All parameters are passed on the command line.

Run the program with no command parameters for a HELP message and the exact command syntax.

This program was written in TURBO Pascal.
The disk parameters are specifically for the VT180

This program is nort for MSX! Study is as an example how to handle a commandline and how to modify a sector on disk.

Scanned, ocr'ed and converted to PDF by HansO, 2002

```
program Startup;
const VERSION = 'V1.2';
{$U+
Copyright (C) 1986 John D. Scott
Author: John Scott
```

Date: 9-Mar-86

Purpose: This program writes a command to be executed on system boot and all warm boots into track 0, sector 1 of the target disk. This is useful for developing application disks. Command can have a max length of 15 characters.

Command string must be null terminated when

written to disk,

History:

4-Oct-86 JDS01 Changed message when startup command is removed to indicate such. Changed No-command to output help message. If command is '?' then display any command currently stored.

}

```
const { CP/M BIOS Function Codes }
  SelDsk = 8; { parameter is drive number 0..15 }
  SetTrk = 9; { parameter is track number }
  SetSec = 10; { parameter is sector number }
  SetDMA = 11; { parameter is address of sector buffer }
  READ = 12;
  WRITE = 13;
```

```
type
  str132 = string[132];
```

```
var
  SectorBuf: array[0..127] of byte;
  Param: string[32];
  DrvChar: char;
  Drive: integer;
  I: integer;
```

```
begin
{ If no command line, output version number and HELP message}
```

```
if ParamCount = 0 then
  begin
    ClrScr;
    writeln('Startup Command Installer. Copyright (C) 1986 John D.
Scott');
    writeln(' Version ',version);
    writeln;
    writeln('Command format:');
    writeln;
    writeln(' Startup <disk> <command>');
    writeln;
    writeln(' where');
    writeln(' <disk> is a drive specifier, ie, B');
```

```

        writeln('          <command> is the command to use');
        writeln('          MAXIMUM of 15 Characters');
        writeln;
        writeln('          If <command> is "7" then any command currently');
        writeln('          installed is displayed.');
```

If there is no new command, then any startup
command');

```

        writeln('          currently installed is removed.');
```

CAUTION: THIS HAS ONLY BEEN TESTED ON DEC VT180s');

```

        exit;
    end;

Param := ParamStr(1);
DrvChar := Ucase(Param[1]);
Drive := ord(DrvChar) - ord('A');
if NOT (Drive in [0..15]) then
    begin
        writeln(Param, ' is not a valid drive designator. Must be in range
A..P.');
```

exit;

```

    end;
Param := '';
if ParamCount >= 2 then
    begin
        for I := 2 to ParamCount do Param := Param + ParamStr(I) + ' ';
        for I := 1 to length(Param) do Param[I] := Ucase(Param[I]);
        Param[length(Param)] := char(0);
        if length(Param) > 16 then
            begin
                writeln(length(Param), ' character command is too long. MAXIMUM
of 15 Characters.');
```

exit;

```

            end;
    end;
BIOS(SelDsk, Drive);
BIOS(SetTrk, 0) ;
BIOS(SetSec, 1) ;
BIOS(SetDMA, Addr(SectorBuf));
if BIOS(READ) <> 0 then
    begin
        writeln('Unrecoverable error occured while reading drive
', DrvChar, '. Sorry. ' ) ;
        exit;
    end;
if Param[1] = '?' then
    begin
        Param[0] := char(SectorBuf[7]);
        if length(Param) = 0 then
            begin
                writeln('No command is currently installed on drive ',
DrvChar, '.');
```

exit;

```

            end;
        for I := 1 to SectorBuf[7] do
```

```

    Param[I] := char(SectorBuf[7+I]);
    writeln('Command installed on drive ',DrvChar,' is: ',Param);
    exit;
end;
if length(Param) > 0 then
    begin
        { Store the passed command }
        SectorBuf[7] := length(Param);
        for I := 1 to length(Param) do
            SectorBuf[7+I] := ord(Param[I]);
        end
    end
else
    begin
        { Length = 0, blank command area }
        for I := 0 to 16 do
            SectorBuf[7+I] := 0;
        end;
    end;
if BIOS(WRITE) <> 0 then
    begin
        writeln('Unrecoverable error occured while writing drive
',DrvChar, '. Sorry. ');
        exit;
    end;
if length(Param) > 0 then
    writeln('Startup command "',Param,'" successfully written to drive
',DrvChar, '. ');
else
    writeln('Startup command successfully removed from drive
',DrvChar, '. ');
end.

```