

# Windows 3.1 SETUP.INF Complete Reference

Of course. My apologies for the previous incomplete response. Here is the full, high-quality translation and restructuring of the provided text into English, formatted for DokuWiki.

**\*\*\* Installation and Configuration of Microsoft Windows 3.1** This guide provides a complete overview of the installation and setup process for Microsoft Windows 3.1. ---

**\*\*\*\* ===== About the Windows Setup Program =====** The Windows Setup program is located on Disk #1 of the installation set. Its purpose is to guide you through the installation process, which is divided into two main phases: - Phase 1: DOS-Based Setup - Setup evaluates your computer's hardware and displays its findings for you to verify. - It then copies the essential Windows files to your hard drive. - Phase 2: Windows-Based Setup - Setup loads the Windows graphical environment for the first time. - It installs the remaining system files and drivers. - You are prompted to configure system settings, install printer drivers, and set up applications. Note: In this documentation, `WINDOWS` refers to the main installation directory you choose, and `SYSTEM` refers to the subdirectory within it where the core operating system files are stored. ---

**\*\*\*\* ===== New Features in Windows 3.1 Setup =====** The Setup program for Windows 3.1 includes several significant improvements: - Compatibility Verification - Setup checks for and warns about device drivers known to be incompatible with either the Setup process or Windows 3.1 itself. It can automatically update drivers from previous Windows 3.x installations. - Automated Hardware Detection - Whether you choose Express or Custom Setup, the program automatically detects most hardware (display, keyboard, mouse). You rarely need to manually select devices from a list. - Startup Group - Setup automatically creates a "Startup" group in Program Manager. Any application placed in this group will launch automatically when Windows starts. - Automated Installation - You can fully automate the installation process by creating a script file. This is ideal for system administrators deploying Windows on multiple machines. ---

**\*\*\*\* ===== Executing the Setup Program =====** The Setup program serves two primary purposes, which require different execution methods:

Purpose	How to Execute
Initial Installation or Upgrade	Run `setup` from the MS-DOS command prompt, either from the floppy disks or a network drive.
Modify Existing Installation	Run Setup from within Windows via the Program Manager to change your system configuration or add/remove components.

Procedure for a standard installation from floppy disks: - Insert Windows 3.1 Disk #1 into your floppy drive. - Switch to the corresponding drive letter (e.g., `A:`). - At the command prompt, type `setup` and press Enter. - Follow the on-screen instructions. Press the F1 key at any time for help.

Critical Note: Do not use the MS-DOS `copy` command to transfer files from the installation disks. The files are compressed and must be processed by the Setup program, which decompresses and renames them correctly. ---

**\*\*\*\* ===== Setup Command-Line Parameters =====** You can customize the Setup program's behavior using various switches when launching it from the command prompt.

Parameter	Purpose
/i	Bypasses automatic hardware detection. You must manually verify and correct settings on the System Information screen.
/n	Installs a shared copy of Windows from a network server.
/a	Performs an Administrative Setup, copying all files from the disks to a network server and marking them as read-only.
/b	Configures Windows for use with a monochrome display.
/t	Scans the drive for software that may conflict with Setup or Windows 3.1 (for maintenance).
/h:[filename]	Runs Batch Mode Setup for an automated installation. The `[filename]` specifies a system settings file containing the user's configuration.
/o:[filename]	Specifies an alternate `SETUP.INF` file.
/s:[path]	Specifies the path to the directory containing the Windows installation files.

| Tip: To see a

**list of all available switches, type `setup /?` at the command prompt. --- ##### =====**

**Express Setup vs. Custom Setup ===== At the beginning of the installation, you must choose between two setup types:**

- Express Setup (Recommended) - The fastest and simplest method, ideal for most users.
- You are only required to provide basic information, such as your printer model and the port it's connected to.
- Setup handles everything else automatically.
- Custom Setup - Designed for advanced users who require precise control over the installation.
- You are asked to verify and potentially modify the detected hardware and software configuration.
- Allows you to select/deselect specific components, choose the installation directory, configure the network, and specify which applications to set up.

**--- ##### ===== Custom Installation Strategies ===== The following decision trees help you plan how to install and customize Windows.**

- For a Single Computer - To customize the \*installation process\*: **Choose Custom Setup at the first prompt.**
- To customize Windows \*after\* installation: **Use Control Panel, Program Manager, and the Windows Setup program within Windows.**
- For Multiple Computers (Network/Automated Deployment) - To deploy on a network: **-> Use `setup /a` to copy all files to a network server. -> Then, use `setup /n` on each workstation to install a shared copy.**
- To automate installation for multiple users: **-> Create custom system settings files and use the `setup /h` switch for a hands-off installation.**
- To customize available hardware options: **-> Edit device entries in the `SETUP.INF` file and printer entries in `CONTROL.INF`.**
- To customize the installed software environment: **-> Modify the relevant `.INF` files to install custom applications, add custom PIFs, or change which Windows components are installed by default.**
- To create a standardized user environment: **-> Set up shared Program Manager groups or edit the `WIN.SRC` and `SYSTEM.SRC` initialization files before installation.**

**===== Technical details =====**

**===== Introduction =====**

**SETUP.INF is the master configuration file that controls every aspect of Windows 3.1 installation. This comprehensive reference documents all sections, keys, and installation processes based on the original Windows Resource Kit documentation.**

**===== Installation Process Overview =====**

**The Windows installation follows this precise sequence:**

- Phase 1: Initialization - Read [setup] section for help file location - Process [data] section for system requirements and defaults - Initialize disk definitions from [disks] and [oemdisks] - Load dialog text from [dialog] section
- Phase 2: File Copying - Copy core Windows files from [windows] section - Copy system files from [windows.system] section - Copy 386-specific files from [windows.system.386] - Install display drivers from [display] section - Install keyboard drivers from keyboard sections - Install mouse drivers from pointing device sections - Copy network components if selected - Install fonts based on display resolution
- Phase 3: Configuration - Update INI files based on [ini.upd.patches] and [ini.upd.31] - Create Program Manager groups - Run post-installation programs from [run] section
- Phase 4: Finalization - Process [blowaway] section to mark installation end - Clean up temporary files

**===== General Installation Sections =====**

**===== [setup] Section =====**

**Defines the Setup help file location. Format: `<code> [setup] help=filename.hlp </code>`**

Parameter	Value	Description
help	setup.hlp	Specifies the help file for Setup program

**Example: `<code> [setup] help=setup.hlp </code>`**

**===== [run] Section =====**

**Specifies programs to execute after Setup completion. Format: `<code> [run] "program_name" </code>`**

Parameter	Description
program_name	Executable to run after installation

**Example: `<code> [run] "control" ; Runs Control Panel after Setup </code>`**

**===== [dialog] Section =====**

**Contains all text strings displayed in Setup dialog boxes. Complete Standard Entries:**

- `<code> [dialog] caption = "Windows Setup" exit = "Exit Windows Setup" title = "Installing Windows 3.1" options = "In addition to installing Windows 3.1, you can:" printwait = "Please wait while Setup configures your printer(s)..."`



**286grabber, logo code, VDD, 386grabber, ega.sys, logo data, optional section** </code>  
Complete Field Definitions: | **Field** | **Description** | |----|-----| | **keyname** | **Profile name for driver references in other SETUP.INF sections** | | **driver** | **Filename of the display device driver** | | **description** | **Driver description in quotes (appears in Change System Settings dialog)** | | **resolution** | **Aspect ratio as three comma-separated numbers in quotes** | | **286grabber** | **Filename of the 286 grabber for standard mode data exchange** | | **logo code** | **Filename for Microsoft Windows logo code** | | **VDD** | **Filename for the Virtual Display Driver** | | **386grabber** | **Filename of the 386 grabber for 386 enhanced mode data exchange** | | **ega.sys** | **Filename for data determining EGA.SYS driver installation need** | | **logo data** | **Filename for appropriate Microsoft logo bitmap** | | **optional section** | **Name of section with additional display driver information** | Important: **Do not change description strings as they are used for upgrade decisions.** Example Entry for 8514/a Display: <code> 8514= 2:8514.driv, "8514/a", "100,120,120", 3:vgacolor.2gr, 2:vgalogo.lgo, 2:vdd8514.386, 3:vgadib.3gr, , 2:vgalogo.rle, 8514 </code> === **Optional Display Sections** === Available optional sections for display drivers: <code> [v7vga], [vgamono], [8514], [8514s], [tiga1], [tiga2], [v7b], [v7c], [v7d], [v7t], [v7e], [xgasm], [xgalg], [xgalo] </code> Optional Section Format: <code> file, destination, ini file, section, OldEntry, NewEntry </code> Optional Section Field Definitions: | **Field** | **Description** | |----|-----| | **file** | **Optional file to copy (can be null)** | | **destination** | **Directory for file copy (0: = WINDOWS, 0:system = SYSTEM)** | | **ini file** | **.INI file to modify (WIN.INI or SYSTEM.INI)** | | **section** | **Section in .INI file to modify** | | **OldEntry** | **Entry to remove from .INI file (in quotes, optional)** | | **NewEntry** | **New entry to add to .INI file (in quotes)** | Example Optional Section: <code> [8514] , , system.ini, 8514.DRV, "dpi=", "dpi=120" </code> === **[.3gr] Font Sections** === Define fonts for 386 grabbers that support text/graphics copying from non-Windows applications. Available .3gr Sections: <code> [vga.3gr], [vgalo.3gr], [v7vga.3gr], [ega.3gr], [vgadib.3gr], [plasma.3gr], [herc.3gr], [tiga.3gr] </code> Example for Video 7 VGA: <code> [v7vga.3gr] 2:CGA40WOA.FON,2:CGA40850.FON 2:CGA80WOA.FON,2:CGA80850.FON 2:EGA40WOA.FON,5:EGA40850.FON 2:EGA80WOA.FON,2:EGA80850.FON </code> Font Selection Rules: \* **Code page 437 (US default): Copies \*WOA.FON files \* Other code pages: Copies \*850.FON files** ===== **Keyboard and Code Page Sections** ===== **[keyboard.drivers] Section** === Maps keyboard driver filenames to short names. Complete Standard Entries: <code> [keyboard.drivers] kbd= 2:keyboard.driv kbdhp= 2:kbdhp.driv </code> === **[keyboard.types] Section** === Creates keyboard short names used in **[machine] section**. Format: <code> short\_name = "description", DLL\_filename </code> Field Definitions: | **Field** | **Description** | |----|-----| | **short\_name** | **Abbreviated descriptor for [machine] section** | | **description** | **Description in quotes for Change System Settings dialog** | | **DLL\_filename** | **Supporting DLL filename (can be null)** | Complete Standard Entries: <code> [keyboard.types] t3s0alat= "All AT type keyboards (84-86 keys)",nodll t1s2at&t= "AT&T '301' keyboard",nodll t1s4at&t= "AT&T '302' keyboard",nodll t4s0enha= "Enhanced 101 or 102 key US and Non US keyboards",nodll t3s0hp1= "Hewlett-Packard Vectra keyboard (DIN)",nodll t4s40oliv= "Olivetti 101/102 A keyboard",nodll t1s0oliv= "Olivetti 83 key keyboard",nodll t3s10oliv= "Olivetti 86 Key keyboard",nodll t2s1oliv= "Olivetti M24 102 key keyboard",usadll t1s42oliv= "PC-XT 83 key keyboard",nodll t1s0pcxt= "PC/XT -Type keyboard (84 keys)",nodll </code> === **[keyboard.tables] Section** === Maps DLL short names to disk locations for international language support. Complete Standard Entries: <code> [keyboard.tables] beldll= 2:kdbde.dll , "Belgian" bridll= 2:kbduk.dll , "British" cafdll= 2:kbdfc.dll , "Canadian Multilingual" dandll= 2:kbdde.dll , "Danish" dutdll= 2:kbdne.dll , "Dutch" findll= 2:kbdfi.dll , "Finnish" fredll= 2:kbdfr.dll , "French" candll= 2:kbdca.dll , "French Canadian" gerdll= 2:kbdgr.dll , "German" icedll=

2:kbdic.dll , "Icelandic" itadll= 2:kbdit.dll , "Italian" latdll= 2:kbdla.dll , "Latin American" nordll= 2:kbdno.dll , "Norwegian" pordll= 2:kbdpo.dll , "Portuguese" spadll= 3:kbdsp.dll , "Spanish" swedll= 2:kbdsw.dll , "Swedish" swfdll= 2:kbdsf.dll , "Swiss French" swgdll= 2:kbdsg.dll , "Swiss German" nodll= , "US" usadll= 2:kbdus.dll , "US" usddll= 2:kbdsv.dll , "US-Dvorak" usxdll= 2:kbdusx.dll, "US-International" </code> === [codepages] Section === Defines code page support for international systems. Format: <code> codepage\_number = xlat\_table, OEMfont, WOAfont, "description" </code> Field Definitions: | Field | Description | |----|-----| | codepage\_number | Table number for keyboard character mapping | | xlat\_table | Translation table filename for OEM font association | | OEMfont | OEM font filename with disk number | | WOAfont | Windows grabber font filename with disk number | | description | Country description in quotes | Complete Standard Entries: <code> [codepages] 863= 1:xlat863.bin, 2:vga863.fon, 2:app850.fon, "Canadian-French (863)" 861= 2:xlat861.bin, 2:vga861.fon, 2:app850.fon, "Icelandic (861)" 865= 2:xlat865.bin, 2:vga865.fon, 2:app850.fon, "Nordic (865)" 850= 2:xlat850.bin, 2:vga850.fon, 2:app850.fon, "Multi-Lingual (850)" 860= 3:xlat860.bin, 2:vga860.fon, 2:app850.fon, "Portuguese (860)" 437= , , 2:dosapp.fon, "English (437)" </code> ===== Mouse Driver Sections ===== [pointing.device] Section === Table of information for supported pointing devices. Format: <code> keyname = mouse\_driver, mouse\_descriptor, VMD, optional\_section </code> Field Definitions: | Field | Description | |----|-----| | keyname | Profile name | | mouse\_driver | Mouse driver filename with disk number | | mouse\_descriptor | Driver description in quotes for Setup dialog | | VMD | Virtual Mouse Device support filename with disk number | | optional\_section | Section with additional mouse driver information | Special VMD Values: \* <code>x:\*vmd</code> - File built into WIN386.EXE (not separate file) \* <code>x:</code> - Refers to disk where WIN386.EXE is located Complete Standard Entries: <code> [pointing.device] hpmouse= 2:hpmouse.driv, "HP Mouse (HP-HIL)", x:\*vmd lmouse= 3:lmouse.driv, "Logitech", 2:lvmd.386, lmouse ps2mouse= 2:mouse.driv, "Microsoft, or IBM PS/2", x:\*vmd genius1= 2:mscmouse.driv, "Genius serial mouse on COM1", 1:mscvmd.386 genius2= 2:m3bc2.driv, "Genius serial mouse on COM2", 1:mscvmd.386 msmouse2= 2:m3bc2.driv, "Mouse Systems serial or bus mouse", 1:mscvmd.386 msmouse1= 2:m3bc2.driv, "Mouse Systems serial mouse on COM2", 1:mscvmd.386 nomouse= 2:nomouse.driv, "No mouse or other pointing device", x:\*vmd kbdmouse= 2:kbdmouse.driv, "Olivetti/AT&T Keyboard Mouse", x:\*vmd </code> === [dos.mouse.driver] Section === Maps Windows mouse drivers to MS-DOS mouse drivers. Complete Standard Entries: <code> [dos.mouse.drivers] mouse.sys= 4:mouse.SYS, "MS DOS Mouse driver .SYS ver 7.XX" mouse.com= 4:mouse.com, "MS DOS Mouse driver .COM ver 7.XX" mousehp.sys= 3:mousehp.SYS, "HP DOS Mouse driver .SYS ver 7.XX" mousehp.com= 3:mousehp.com, "HP DOS Mouse driver .COM ver 7.XX" </code> === [lmouse] Section === Support data for Logitech MS-DOS mouse driver. Complete Standard Entry: <code> [lmouse] 2:lmouse.com,0:,,, </code> ===== Network Installation Sections ===== [network] Section === Associates network keynames with installation files. Format: <code> network\_keyname = driver, description, HelpFile, optional\_file, WininiSectName, SysiniSectName, VDD, ... </code> Field Definitions: | Field | Description | |----|-----| | network\_keyname | Profile for specific network driver | | driver | Network driver filename with disk number | | description | Driver description in quotes for Setup dialog | | HelpFile | Associated help filename | | optional\_file | Optional file to install | | WininiSectName | WIN.INI section to add/modify | | SysiniSectName | SYSTEM.INI section to add/modify | | VDD | Virtual Display Device support filenames | Complete Standard Entries: <code> [network] nonet= , "No Network Installed" 3open= , "3Com 3+Open" 3share= , "3Com 3+Share" ,,msnet\_stuff lantastic= , "Artisoft LANtastic" banyan= , "Banyan Vines" dlr= , "IBM OS/2 LAN Server" pclp= , "IBM PC LAN Program" lanman= , "Microsoft LAN Manager" msnet= , "Microsoft Network (or 100% compatible)",

**,,msnet\_stuff novell= ,“Novell Netware” pathworks= ,“DEC Pathworks” 10net= ,“TCS 10Net”** `=== [Network_Version] Sections === Provide version-specific network information. Available Version Sections: [10net.versions] ; TCS 10Net [3open.versions] ; 3Com 3+Open [banyan.versions] ; Banyan Vines [dlr.versions] ; IBM OS/2 LAN Server [lanman.versions] ; Microsoft LAN Manager [lantastic.versions] ; Artisoft LANtastic [novell.versions] ; Novell Netware [pathworks.versions] ; DEC Pathworks [pclp.versions] ; IBM PC LAN Program Format: version_number = “description”, optional_files, optional_sections Example for LAN Manager: [lanman.versions] xx000000=“versions 1.X”, , lmbasic 00020000=“version 2.00 Basic”, , lmbasic 01020000=“version 2.00 Enhanced”, , lmenh20 00021000=“version 2.10 Basic”, , lmbasic21 01021000=“version 2.10 Enhanced”, lanman21.drv, lmenh21 nodrv,lmenh21`

**</code> === [Network\_Specific] Sections === Define INI file modifications for specific networks. Available Network\_Specific Sections: `[10net41], [10net41a], [10net42], [10net42def], [10net50] [3open1] [ban4], [ban41], [ban41def] [dec4], [dec41] [dlr12], [dlr12csd], [dlr13] [lant1], [lant4], [lant4def] [lmbasic], [lmbasic21], [lmenh20], [lmenh21], [lmenh21nodrv] [msnet_stuff] [novell301], [novell321] [old_pclp], [pclp132] [unsupported_net]`** `Format: [network_keyname] ini-keyname = section, entry_keyname, value` Example for Novell NetWare 3.01: `[novell301] warning=2 driver=2:netware.drv vxd=x:*vnetbios,2:vnetware.386,2:vipx.386 sysdir=2:netware.hlp,2:nwpopup.exe windir=2:ipx.obj,2:ipxodi.com,2:netx.com,2:lsl.com windir=2:tbmi2.com`

**==== System Fonts Sections ===== [sysfonts] Section === System fonts by display resolution. Complete Standard Entries: `[sysfonts] 1:egasys.fon,“EGA (640×350) resolution System Font”, “133,96,72” 1:vgasys.fon,“VGA (640×480) resolution System Font”, “100,96,96” 1:8514sys.fon,“8514/a (1024×768) resolution System Font”, “100,120,120” 1:egasys.fon,“AT&T (640×400) resolution System Font”, “120,96,80”`**

**</code> === [fixedfonts] Section === Fixed-width system fonts by display resolution. Complete Standard Entries: `[fixedfonts] 2:egafix.fon,“EGA (640×350) resolution Fixed System Font”, “133,96,72” 1:vgafix.fon,“VGA (640×480) resolution Fixed System Font”, “100,96,96” 1:8514fix.fon,“8514/a (1024×768) resolution Fixed System Font”, “100,120,120” 2:egafix.fon,“AT&T (640×400) resolution Fixed System Font”, “120,96,80”`**

**</code> === [oemfonts] Section === Terminal fonts by display resolution. Complete Standard Entries: `[oemfonts] 1:egaoem.fon,“EGA (640×350) resolution Terminal Font (USA/Europe)”, “133,96,72”,1 1:vgaoem.fon,“VGA (640×480) resolution Terminal Font (USA/Europe)”, “100,96,96”,1 1:8514oem.fon,“8514/a (1024×768) resolution Terminal Font (USA/Europe)”, “100,120,120”,1 1:egaoem.fon,“AT&T (640×400) resolution Terminal Font (USA/Europe)”, “120,96,80”,1`**

**</code> ===== Program Manager Groups Sections ===== [new.groups] Section === Used when upgrading from Windows 3.0 to 3.1. Format: `section = Progman_group, Min/Max`** Field Definitions: | Field | Description | |---|-----| | section | Name of [group#] section with group definition | | Progman\_group | Profile name identifying group in Program Manager | | Min/Max | 1=maximize group, nothing=minimize group | Complete Standard Entries: `[new.groups] group7=Main, 1 group1=Startup group2=Accessories group6=Games`

**</code> === [progman.groups] Section === Used for new Windows 3.1 installations. Complete Standard Entries: `[progman.groups] group3=Main, 1 group4=Accessories group5=Games group1=Startup`**

**</code> === [group#] Sections === Define program items within Program Manager groups. Format: `“description”, filename.exe, icon_filename, icon_number, profile`** Field Definitions: | Field | Description | |---|-----| | description | Text below icon in Program Manager (in quotes) | | filename.exe | Executable filename with extension | | icon\_filename

| **Icon extraction file (blank = first icon from filename.exe) | | icon\_number | Icon offset number (for 3rd icon, specify 2) | | profile | If null, always added; otherwise only if installed via Add/Remove Files | Complete Group Definitions: Main Group ([group3] for [progman.groups]):** `[group3] "File Manager", WINFILE.EXE "Control Panel", CONTROL.EXE "Print Manager", PRINTMAN.EXE "Clipboard Viewer", CLIPBRD.EXE "MS-DOS Prompt", DOSPRMPT.PIF, PROGMAN.EXE, 9 "Windows Setup", SETUP.EXE "PIF Editor", PIFEDIT.EXE "Read Me", README.WRI,, readme` Accessories Group ([group4] for [progman.groups]): `[group4] "Write", WRITE.EXE,,, write "Paintbrush", PBRUSH.EXE,,, pbrush "Terminal", TERMINAL.EXE,,, terminal "Notepad", NOTEPAD.EXE,,, notepad "Recorder", RECORDER.EXE,,, recorder "Cardfile", CARDFILE.EXE,,, cardfile "Calendar", CALENDAR.EXE,,, calendar "Calculator", CALC.EXE,,, calc "Clock", CLOCK.EXE,,, clock "Object Packager", PACKAGER.EXE,,, packager "Character Map", CHARMAP.EXE,,, charmap "Media Player", MPLAYER.EXE,,, mplayer "Sound Recorder", SOUNDREC.EXE,,, soundrec` Games Group ([group5] for [progman.groups]): `[group5] "Solitaire", SOL.EXE,,, sol "Minesweeper", WINMINE.EXE,,, winmine` Additional Groups for [new.groups]: `[group1] ; For [progman.groups] and [new.groups] [group2] ; For [new.groups] "Object Packager", PACKAGER.EXE,,, packager "Character Map", CHARMAP.EXE,,, charmap "Media Player", MPLAYER.EXE,,, mplayer "Sound Recorder", SOUNDREC.EXE,,, soundrec "PIF Editor" "Write", WRITE.EXE "Control Panel" [group6] ; For [new.groups] "Solitaire", SOL.EXE,,, sol "Minesweeper", WINMINE.EXE,,, winmine [group7] ; For [new.groups] "PIF Editor", PIFEDIT.EXE "Read Me" "Read Me", README.WRI,, readme "Tutorial"` **==== Fonts Sections ====** **==== [fonts] Section ====** Raster and vector screen fonts based on display resolution. Format: `filename, "description", "resolution"` Example VGA Entries: `[fonts] (other entries) 6:SSERIFE.FON, "MS Sans Serif 8,10,12,14,18,24 (VGA res)", "100,96,96" (other entries) 6:COURE.FON, "Courier 10,12,15 (VGA res)", "100,96,96" (other entries) 6:SERIFE.FON, "MS Serif 8,10,12,14,18,24 (VGA res)", "100,96,96" (other entries) 6:SYMBOLE.FON, "Symbol 8,10,12,14,18,24 (VGA res)", "100,96,96" (other entries) 6:SMALLE.FON, "Small (VGA res)", "100,96,96"` **==== [ttfonts] Section ====** TrueType font definitions. Format: `font_header_file, "description", font_file, fontfamily_flags` Field Definitions: | Field | Description | |---|-----| | font\_header\_file | TrueType header source filename | | description | Font description in quotes (first letter of each word capitalized) | | font\_file | TrueType font filename | | fontfamily\_flags | Font family name plus flag in quotes | Font Family Flags: \* 0000 = Normal weight font \* 0100 = Bold font \* 1000 = Italic font \* 1100 = Bold Italic font Complete TrueType Entries: `[ttfonts] S:ARIAL.FOT, "Arial (TrueType)", 5:arial.ttf, "" S:ARIALBD.FOT, "Arial Bold (TrueType)", S:arialbd.ttf, "Arial0100" S:ARIALBI.FOT, "Arial Bold Italic (TrueType)", 5:arialbi.ttf, "Arial1100" S:ARIALI.FOT, "Arial Italic (TrueType)", 5:ariali.ttf, "Arial1000" S:COUR.FOT, "Courier New (True Type)", 5:cour.ttf, "" S:COURBD.FOT, "Courier New Bold (TrueType)", 5:courbd.ttf, "Courier0100" S:COURBI.FOT, "Courier New Bold Italic (TrueType)", 5:courbi.ttf, "Courier1100" S:COURI.FOT, "Courier New Italic (TrueType)", 5:couri.ttf, "Courier1000" 3:TIMES.FOT, "Times New Roman (TrueType)", 4:times.ttf, "" 3:TIMESBD.FOT, "Times New Roman Bold (TrueType)", 4:timesbd.ttf, "Times New Roman0100" 3:TIMESBI.FOT, "Times New Roman Bold Italic (TrueType)", 4:timesbi.ttf, "Times New Roman1100" 4:TIMESI.FOT, "Times New Roman Italic (TrueType)", 4:timesi.ttf, "Times New Roman1000" S:SYMBOL.FOT, "Symbol (TrueType)", S:symbol.ttf, ""` **==== Incompatible Driver Sections ====** Four sections for checking incompatible drivers and TSRs: **==== [compatibility] Section ====** Files listed here have their CONFIG.SYS commands removed during Setup. **==== [incompTSR1] Section ====** TSRs and drivers that prevent Setup from running. Must be removed from memory and AUTOEXEC.BAT/CONFIG.SYS before running Setup. **====**

**[incompTSR2] Section === TSRs and drivers that cause problems during Setup or when starting Windows. === [block\_devices] Section === Block devices that are not compatible with Windows 3.1. ===== Miscellaneous Sections ===== [installable.drivers] Section === Multimedia driver specifications. Format: `<code> keyname= filename, types, description, VxDs, default_parameters </code>` Complete Standard Entries: `<code> [installable.drivers] adlib= 3:adlib.driv, "MIDI", "Ad Lib", 4:vadlibd.386, lapc1= 3:mpu401.driv, "MIDI", "Roland LAPC1",, midimapper= 4:midimap.driv, "MidiMapper", "MIDI Mapper",, mpu401= 3:mpu401.driv, "MIDI", "Roland MPU401",, sequencer= 4:mciseq.driv, "Sequencer", "[MCI] MIDI Sequencer",, soundblaster= 3:sndblst.driv, "Wave,MIDI", "Creative Labs Sound Blaster 1.0", 3:vsbd.386,, adlib soundblaster2= 3:sndblst2.driv, "Wave,MIDI", "Creative Labs Sound Blaster 1.5", 3:vsbd.386,, adlib timer= 5:timer.driv, "Timer", "Timer", 5:vtdapi.386, thunder= 3:sndblst2.driv, "Wave,MIDI", "Media Vision Thunder Board", 3:vsbd.386,, adlib wave= 4:mcwave.driv, "WaveAudio", "[MCI] Sound",,, "4" </code>` === [translate] Section === OEM filename translations. Example Entries: `<code> [translate] WinDiskName=25, "Microsoft Windows 3.0 Disk" WinDiskName=14, "Windows Disk#" *vddvga=2:vddvga30.386 cti1.lgo=2:herclogo.lgo herclogo.rle=1:herclogo.rle vddega=2:vddega.386 vgacolor.gr2=3:vgacolor.2gr vgalogo.lgo=2:vgalogo.lgo vgalogo.rle=2:vgalogo.rle rgdi.lgo=2:vgalogo.lgo rgdi.rle=2:vgalogo.rle vga.gr3=1:vga30.3gr </code>` === [update.files] Section === Installable drivers to update if earlier versions are found. Format: `<code> location, filename </code>` Location Values: \* 0: = WINDOWS directory \* O:system = SYSTEM subdirectory Complete Standard Entries: `<code> [update.files] O:system,3:sndblst.driv O:system,3:sndblst2.driv O:system,3:adlib.driv O:system,3:mpu401.driv O:system,5:vtdapi.386 O:system,3:vsbd.386 O:system,4:vadlibd.386 </code>` === [Update.Dependents] Section === Updates files dependent on those listed in [Update.Files]. Complete Standard Entries: `<code> [Update.Dependents] msadlib.driv = 4:vadlibd.386 sndblst.driv = 4:vsbd.386 sndblst2.driv = 4:vsbd.386 </code>` === [ini.upd.patches] Section === Temporary INI file profile renaming during Setup. Format: `<code> ini_file, section, temporary_profile, original_profile </code>` Example Entries: `<code> [ini.upd.patches] system.ini, Boot, "olddrivers", "drivers" win.ini, Desktop, "oldwallpaper", "wallpaper" system.ini, Boot, "oldtaskman", "taskman.exe" </code>` === [blowaway] Section === Marks the end of installation information and start of configuration sections. Windows Mode Setup stops reading SETUP.INF here. === [ini.upd.31] Section === INI file updates for Windows 3.0 to 3.1 upgrade. Format: `<code> ini_file, section, OldEntry, NewEntry </code>` Example Entries: `<code> [ini.upd.31] system.ini, 386Enh, "keyboard=*vkd" system.ini, 386Enh, "device=*vpicd" system.ini, 386Enh, "device=*blockdev" system.ini, 386Enh, , "device=*pagefile" win.ini, FontSubstitutes, , "Helv=MS Sans Serif" win.ini, FontSubstitutes, , "Tms Rmn=MS Serif" win.ini, FontSubstitutes, , "Courier=Courier New" win.ini, FontSubstitutes, , "Times=Times New Roman" win.ini, FontSubstitutes, , "Helvetica=Arial" </code>` ===== System Configuration Sections ===== [system] Section === Maps system short names to appropriate system files. Complete Standard Entries: `<code> [system] system= 2:system.driv sound= 1:mmsound.driv comm= 1:comm.driv hpsystem= 2:hpsystem.driv </code>` === [machine] Section === Installs appropriate system files for various machines. Format: `<code> keyname= description, system_drv, kbd_drv, kbd_type, mouse_drv, disp_drv, sound_drv, comm_drv, himem_switch, ebios, Cookz </code>` Complete Field Definitions: | Field | Description | |----|----| | keyname | Profile string for the system | | description | System description in quotes for Setup dialog | | system\_drv | Short name for system driver from [system] | |**

**kbd\_drv** | Short name for keyboard driver from [keyboard.drivers] | | **kbd\_type** | Short name for keyboard type from [keyboard.types] | | **mouse\_drv** | Mouse driver keyname from [pointing.device] | | **disp\_drv** | Display driver keyname from [display] | | **sound\_drv** | Sound driver name from [system] | | **comm\_drv** | Communications driver name from [system] | | **himem\_switch** | A20 handler number (1-8) for HIMEM.SYS extended memory access | | **ebios** | EBIOS support defined in [ebios] | | **Cookz** | [Cookz] section with INI file modification information | Important: **Do not change the order of entries in [machine] section.** Example Entries: `[machine] ibm_compatible= "MS-DOS System",system,kbd,t4s0enha,nomouse,egahires,sound,comm,,ebios, ast_386_486= "AST: all 80386 and 80486 based machines",system,kbd,t4s0enha,nomouse,egahires,sound,comm,,ebios,ast_cookz at_and_t= "AT&T PC",system,kbd,t4s0enha,nomouse,vgahires,sound,comm,,ebios,` `</code> === [Cookz] Sections === Hardware-specific INI file modifications. Available Cookz Sections: <code> [apm_cookz], [apm_sl_cookz], [ast_cookz], [everex_cookz], [ibm140_cookz], [ncr386sx_cookz], [nec_pm_cookz], [t5200_cookz], [zen386_cookz] </code> Format: <code> ini_file, section, cookie, needed_file </code> Example Entries: <code> [ast_cookz] system.ini,386enh,"emmexclude=E000-EFFF", [apm_cookz] specialdriver,,5:power.driv system.ini,386enh, "device=vpowrd.386",5:vpowrd.386 ,,5:power.hlp </code> === [special_adapter] Section === Support for special adapters requiring additional files. Complete Standard Entries: <code> [special_adapter] etherlink= DMAbuffersize,32 ; Needed if EtherLink MC is detected cdrom= 2:lanman10.386 ; Needed for CD ROM extensions </code> === [ebios] Section === Extended BIOS support files. Complete Standard Entries: <code> [ebios] ebios= x:*ebios ; Required for most 386 machines hpebios= 2:hpebios.386,x:*ebios ; Required for HP 386 machines </code> === [language] Section === Language support libraries. Format: <code> keyname= language_DLL, DLL_description, language_ID </code> Complete Standard Entries:**`

```
[language]
dan= 2:langsca.dll, "Danish", 1030
nld= 2:langdut.dll, "Dutch", 1043
enu=, "English (American)", 1033
eng= 2:langeng.dll, "English (International)", 2057
fin= 2:langsca.dll, "Finnish", 1035
fra= 2:langfrn.dll, "French", 1036
frc= 2:langeng.dll, "French Canadian", 3084
deu= 2:langger.dll, "German", 1031
isl= 2:langsca.dll, "Icelandic", 1039
ita= 2:langeng.dll, "Italian", 1040
nor= 2:langsca.dll, "Norwegian", 1044
ptg= 2:langeng.dll, "Portuguese", 2070
esp= 2:langspa.dll, "Spanish (Classic)", 1034
esn= 2:langeng.dll, "Spanish (Modern)", 3082
sve= 2:langsca.dll, "Swedish", 1053
```

## File Copying Mechanism Details

The SETUP.INF file copying system operates through these key mechanisms:

Last update: 2025/10/21 11:57 en:docs:win16:applications:setup <https://www.osfree.ru/doku/doku.php?id=en:docs:win16:applications:setup&rev=1761047875>

---

From:  
<https://www.osfree.ru/doku/> - **osFree wiki**

Permanent link:  
<https://www.osfree.ru/doku/doku.php?id=en:docs:win16:applications:setup&rev=1761047875>

Last update: **2025/10/21 11:57**

