

Printable Version of Topic

[Click here to view this topic in its original format](#)

The CD Forum _ General _ Multiboot Usb (pe, Linux And Dos)

Posted by: diddy Nov 9 2006, 05:07 PM

Guide for Multi-booting from a USB Drive

During my experiments (+ searching forums) I have come across two different methods for multibooting USB sticks. Method 1 involves using a bootloader (such as Syslinux or GRUB) with a USB stick containing one partition. Method 2 uses a USB stick containing multiple partitions, each containing their own boot sector.

The following guide will create a multi-boot USB stick with options for BartPE, Linux (DSL, Puppy Linux & Gparted), Windows PE 2.0 and various floppy boot disk images (Windows98, Memtest86) - additional options can be added. You will need a 512MB USB stick if you want to include all of the options (excluding WindowsPE 2.0) in this guide (files take up approximately 312MB, leaving plenty of room for data storage). If you want to add Windows PE 2.0 you will need a larger drive - if adding PE 2.0 read step 10 before proceeding.

I take no credit for the ideas in this guide, as it is a compilation of ideas taken from various web sites/forums. In a frantic effort to get this working I chased links from forum to forum without taking any notes and cannot remember (or find) where some of the information came from.

NOTE

Bios support for booting from a USB stick is dependent on your motherboard. There is no guarantee that this guide will work for you - I take no responsibility if anything goes wrong. For information about booting from USB drives look here:

<http://www.pcw.co.uk/articles/print/2152896>

<http://home.graffiti.net/jaclaz:graffiti.net/Projects/USB/USBfaqs.html>

I used a 1GB Kingston Datatraveller 2.0 USB with Asus A7V400-MX SE Motherboard.

Method 1_1 - Using Syslinux

Step 1 - download the following

* <http://h18000.www1.hp.com/support/files/serveroptions/us/download/20306.html> (HP USB Disk Storage Format Tool) - used Version 2.0.6

* <http://syslinux.zytor.com/download.php> - used Version 3.31

*

<http://www.microsoft.com/downloads/details.aspx?amp;displaylang=en&familyid=22CFC239-337C-4D81-8354-7259>
(if using BartPE)

* OS's you would like to include (BartPE, WindowsPE etc)

Step 2a - Test whether drive is bootable (optional)

As noted earlier, Bios support for booting from USB devices is inconsistent. I would advise you to test whether you are able to boot at this point before proceeding with the following steps, as you will save yourself a great deal of time. You will need to insert a DOS floppy boot disk (or mount a disk image in a virtual floppy drive). Use HP's USB Disk Storage Format Tool to make your drive bootable.

* Run HPUSBFW.EXE and select the following options.

Device - your USB Drive

File System - FAT

Create DOS Startup Disk - check box.

Using DOS System Files Located at: - browse to your floppy disk

- * START
- * Following files should have been copied to your USB drive - command.com, MSdos.sys and io.sys
- * Copy other files from your floppy to your USB drive (do not replace command.com, MSdos.sys and io.sys)

A detailed guide is available here:

<http://www.nforcershq.com/forum/viewtopic.php?p=575175>

Now reboot your computer and change the bios settings to boot from your USB drive. The settings will vary depending on your motherboard. On ASUS A7V400-MX SE Motherboard - press DEL to enter setup. Goto the "boot" menu and select the "Hard Disk Drives" option, and change the list so that your USB Key is displayed as the first hard disk to boot (prior to formatting with HPUSBFW.EXE my USB drive did not show up in the bios settings). Hopefully your USB drive will boot successfully.

Step 2b - Format USB Drive

Use HP's USB Disk Storage Format Tool to make your drive bootable.

- * Run HPUSBFW.EXE

Device = your USB Drive

File System = FAT

Leave all other settings empty

START

Step 3 - Install syslinux

- * Create folder C:\syslinux
- * Extract the contents of the syslinux zip file downloaded in step 1 to C:\syslinux
- * Run the following command C:\syslinux\win32\syslinux E: (where E: is your USB Drive)

If the install was successful ldlinux.sys should have been created in the USB drive's root.

Step 4 - Create boot menu + config file

- * Create menu.txt (in notepad) - leave blank for now
- * Create syslinux.cfg (in notepad) and add the following text

CODE

```
PROMPT 1
TIMEOUT 0
DEFAULT 1
DISPLAY Menu.txt
```

Step 5 - Add bootable floppy disk image(s)

- a) Copy memdisk from C:\syslinux\memdisk to the root of your USB Drive
- b) Copy a disk image to the root of your USB Drive (e.g. win98.ima)
- c) Edit menu.txt and include the following text (without inverted commas) "1.- Windows 98 boot disk"
- d) Edit syslinux.cfg and add the following lines

CODE

```
LABEL 1
KERNEL memdisk
APPEND initrd=win98.ima
```

Repeat b), c) and d) for each bootable disk image you want to include, and add corresponding entries in menu.txt and syslinux.cfg. e.g.

- * Copy Memtest86.ima to the root of your USB Drive
- * Edit menu.txt and include the following text "2.- Memtest86"
- * Edit syslinux.cfg and add the following lines

CODE

```
LABEL 2
```

```
KERNEL memdisk
APPEND initrd=Memtest86.ima
```

Step 6 - Add Puppy Linux

- * Download the latest Puppy Linux ISO file - Tested using "puppy-2.11-seamonkey.iso"
- * Extract the following from the ISO file - initrd.gz, pup_2**.sfs, vmlinuz
- * Copy initrd.gz, pup_2**.sfs and vmlinuz to the root of your USB Drive
- * Create USBFLASH file in the root of your USB Drive
- * Edit menu.txt and include the following text "3.- Linux Puppy"
- * Edit syslinux.cfg and add the following lines

CODE

```
LABEL 3
KERNEL vmlinuz
APPEND root=/dev/ram0 initrd=initrd.gz PMEDIA=usbflash PKEYS=uk
```

Step 7 - Add DSL (Damn Small Linux)

- * Download the latest DSL Embedded .ZIP file - Tested using "dsl-3.1RC2-embedded.zip"
- * Extract the KNOPPIX folder (with contents) from the .ZIP file to the root of your USB Drive
- * Extract linux24 and minirt24.gz from the .ZIP file to the root of your USB Drive
- * Edit menu.txt and include the following text "4.- Damn Small Linux"
- * Edit syslinux.cfg and add the following lines

CODE

```
LABEL 4
KERNEL linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=791 initrd=minirt24.gz nomce noapic qemu quiet
BOOT_IMAGE=knoppix frugal
```

Step 8 - Add gparted

GParted (Gnome Partition Editor) offers some of the functionality of Paragon's Partition Manager at no cost (including creating, resizing and moving partitions).

- * Download the latest gparted liveusb .ZIP file - Tested using "gparted-liveusb-0.3.1-1.zip"
- * Extract the contents of the .ZIP file to a temporary location
- * Rename initrd.gz to gpinitrd.gz (so that it can co-exist with initrd from Puppy Linux)
- * Copy gpinitrd.gz, linux and gparted files from the temporary location to the root of your USB Drive
- * Edit menu.txt and include the following lines

CODE

```
5.- GParted
6.- GParted (Use if RAM < 64MB)
```

- * Edit syslinux.cfg and add the following lines

CODE

```
LABEL 5
KERNEL linux
APPEND noapic initrd=gpinitrd.gz root=/dev/ram0 init=/linuxrc ramdisk_size=65000

LABEL 6
KERNEL linux
APPEND noapic initrd=gpinitrd.gz root=/dev/ram0 init=/linuxrc ramdisk_size=16384
```

Step 9 - Add BartPE

You will need to create a RAM loading image to boot BartPE from this drive. This process is beyond the scope of this tutorial, however the relevant information is available elsewhere. Either an ISO file or SDI file can be booted. Follow the instructions here for ISO:

<http://www.911cd.net/forums//index.php?showtopic=10482>

or here for SDI:

<http://www.911cd.net/forums//index.php?showtopic=12413>

<http://www.myitforum.com/articles/8/view.asp?id=8832>

IMPORTANT Remember to overwrite the RAMDISK.SYS in BartPE with the RAMDISK.SYS from Windows 2003 sp1 before building the image file

* Copy either the ISO or SDI file to the root of your USB Drive and change name to PE.ISO (or PE.SDI).

* Create a WINNT.SIF file at the root of your USB Drive.

* If booting an ISO file contents of WINNT.SIF should be as follows (where PE.ISO is the name of your ISO file):

CODE

```
[SetupData]
BootDevice = "ramdisk(0)"
BootPath = "\i386\System32\"
OsLoadOptions = "/noguiboot /fastdetect /minint /rdexportascd /rdpath=PE.iso"
```

* If booting an SDI file contents of WINNT.SIF should be as follows (where PE.SDI is the name of your SDI file):

CODE

```
[SetupData]
BootDevice = "ramdisk(0)"
BootPath = "\i386\System32\"
OsLoadOptions = "/noguiboot /fastdetect /minint /rdimageoffset=36352 /rdpath=PE.SDI"
```

* Copy SETUPLDR.BIN from Windows 2003 SP1 to the root of your USB drive and rename to NTLDR

* Copy NTDETECT.COM to the root of your USB drive

* Copy PE2USB.BIN from pebuilder3110a to the root of your USB drive and rename to PE.bss

* Edit menu.txt and include the following text "7.- BartPE"

* Edit syslinux.cfg and add the following lines

CODE

```
LABEL 7
KERNEL pe.bss
APPEND -
```

Step 9a - Add another BartPE

Information has been adapted from "Use Multiple Unattended Files In A Win2k/xp/2k3 Cd" by the Truth

A second disc image (again either an ISO or SDI file) can be added to the USB drive. To do so you will need to use a cracked SETUPLDR.BIN from Windows2003 SP1 (search this forum for information re what to hex edit, or download a cracked version).

* Rename your cracked SETUPLDR.BIN to NTLD2 and copy to the root of your USB drive

* Hex edit NTLD2 and replace all references to WINNT.SIF with NTLD2.SIF (should be 9 entries replaced)

* Copy PE2USB.BIN to the root of your USB drive and rename to NTLD2.bss

* Hex edit NTLD2.bss and replace NTLDR with NTLD2

* Create a file named NTLD2.SIF - following the instructions for WINNT.SIF in Step 9. Replace the text "/rdpath=PE.ISO" (or "/rdpath=PE.ISO") with "/rdpath=NTLD2.ISO" (or "/rdpath=NTLD2.SDI").

* Name your new image file NTLD2.ISO (or NTLD2.SDI) and copy to the root of your USB drive

* Add an entry in menu.txt (eg "8.- BartPE 2")

* Add an entry in syslinux.cfg

CODE

```
LABEL 8
KERNEL NTLD2.bss
APPEND -
```

The name of your .SIF file is irrelevant, as long as it is 5 characters long. This also applies to your renamed SETUPLDR.BIN. Step 9a can be repeated as many times as you like to include multiple BartPE images.

Step 10 - Add Windows PE 2.0 (optional)

Windows PE 2.0 can be added - you will need to download, and install, the Windows Automated Installation Kit (WAIK) from the following site

<http://www.microsoft.com/downloads/thankyou.aspx?familyId=c7d4bc6d-15f3-4284-9123-679830d629f2&displayLang>

To add PE 2.0 you will need to extract a bootsector from a working installation (on USB stick) using Bart's mkbnt.exe (contained in pebuilder). To create a working installation you will need to follow the instructions in the user guide contained in the WAIK (in the "Windows PE Walkthroughs section"). Ignore the instructions to use FAT32 file system and use FAT instead.

Copy the extracted bootsector to the root of your USB drive and rename to PE2.bss

* Follow the instructions in the WAIK guide to create PE 2.0 files.

* Copy the contents of the "ISO" folder created in the preceding step to the root of your USB stick.

* Add an entry in menu.txt (eg "9.- WinPE 2.0")

* Edit syslinux.cfg and add the following lines

CODE

```
LABEL 9
KERNEL pe2.bss
APPEND -
```

Finished

If all steps were followed your completed menu.txt should look like this:

CODE

```
1.- Windows 98 boot disk
2.- Memtest86
3.- Linux Puppy
4.- Damn Small Linux
5.- GParted
6.- GParted (Use if RAM < 64MB)
7.- BartPE
8.- BartPE 2
9.- WinPE 2.0
```

syslinux.cfg should look something like this:

CODE

```
PROMPT 1
TIMEOUT 0
DEFAULT 1
DISPLAY Menu.txt
LABEL 1
KERNEL memdisk
APPEND initrd=win98.ima

LABEL 2
KERNEL memdisk
APPEND initrd=Memtest86.ima

LABEL 3
KERNEL vmlinuz
APPEND root=/dev/ram0 initrd=initrd.gz PMEDIA=usbflash PKEYS=uk

LABEL 4
KERNEL linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=791 initrd=mini24.gz nomce noapic qemu quiet
BOOT_IMAGE=knoppix frugal

LABEL 5
KERNEL linux
APPEND noapic initrd=gpinitrd.gz root=/dev/ram0 init=/linuxrc ramdisk_size=65000

LABEL 6
KERNEL linux
APPEND noapic initrd=gpinitrd.gz root=/dev/ram0 init=/linuxrc ramdisk_size=16384

LABEL 7
KERNEL pe.bss
APPEND -

LABEL 8
```

```

KERNEL NTLD2.bss
APPEND -

LABEL 9
KERNEL pe2.bss
APPEND -

```

IMPORTANT The APPEND sections may have become wordwrapped on this page and should be on one line.

Now reboot your PC and change Bios settings to boot from your USB Drive. If everything went according to plan you should see a menu with options 1-7 - type the corresponding number and press ENTER

Final Thoughts

* Booting floppy disk images is very straightforward and adding additional images should not be difficult (as long as MEMDISK is able to boot them).

* Booting Linux is more complicated and involved some trial and error. If the distro includes its own syslinux config file (syslinux.cfg), this can be edited. As an example the syslinux.cfg file in dsl-3.1RC2-embedded.zip contained the following entries:

CODE

```

DEFAULT linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=791 initrd=minirt24.gz nomce noapic qemu quiet
BOOT_IMAGE=knoppix frugal
TIMEOUT 300

PROMPT 1
DISPLAY boot.msg
F1 boot.msg
F2 f2
F3 f3
LABEL dsl
KERNEL linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=791 initrd=minirt24.gz nomce noapic qemu quiet
BOOT_IMAGE=knoppix frugal
LABEL knoppix-txt
KERNEL linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=normal initrd=minirt24.gz nomce noapic qemu
quiet BOOT_IMAGE=knoppix frugal
LABEL expert
KERNEL linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=791 initrd=minirt24.gz nomce noapic
BOOT_IMAGE=expert frugal
LABEL fb1280x1024
KERNEL linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=794 xmodule=fbdev initrd=minirt24.gz nomce
noapic qemu quiet BOOT_IMAGE=knoppix frugal
LABEL fb1024x768
KERNEL linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=791 xmodule=fbdev initrd=minirt24.gz nomce
noapic qemu quiet BOOT_IMAGE=knoppix frugal
LABEL fb800x600
KERNEL linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=788 xmodule=fbdev initrd=minirt24.gz nomce
noapic qemu quiet BOOT_IMAGE=knoppix frugal
LABEL failsafe
KERNEL linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us vga=normal atapid nosound noapic noacpi acpi=off noscsi noapm
noub nopcmcia nofirewire noagp nomce nodhcp xmodule=vesa initrd=minirt24.gz BOOT_IMAGE=knoppix frugal
LABEL userdef
KERNEL linux24
APPEND #####
#####
#
#####
#
#####
#
#####
#####

```

```
#
#####
#
#####
```

In this guide only the default entry was used.

CODE

```
DEFAULT linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=791 initrd=minirt24.gz nomce noapic qemu quiet
BOOT_IMAGE=knoppix frugal
```

was edited to

CODE

```
LABEL 4
KERNEL linux24
APPEND ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=791 initrd=minirt24.gz nomce noapic qemu quiet
BOOT_IMAGE=knoppix frugal
```

* Booting BartPE took many hours of trial and error before stumbling on renaming a bootsector to .bss extension and using this as a kernel entry to chain load PE. Typically after all this work and having just completed this guide I came across the following forum entry (with link to Irongeeks site/guide), which would have saved me days of work. It's definately worth checking this link as it includes information on dualbooting with SLAX Linux, which can be added to your USB drive:
<http://www.911cd.net/forums//index.php?showtopic=18741>

The boot menu (menu.txt) outlined in this guide is very basic. Two other menu systems are referenced in the syslinux download and will be worth adding. I would however recommend that you stick with menu.txt until everything is tested and working.

Method 1_2 - Using GRUB4DOS

This section is dedicated to Jaclaz whose name seems to crop up whenever GRUB is mentioned. I have found his help invaluable and would not have been able to complete this section without him.

I don't want to repeat myself too much in this section, as its very similar to the Syslinux method above. You will need to download the latest version of **GRUB4DOS** - I used version <http://sarovar.org/projects/grub4dos/>. I previously tried using version 0.4.1 from Sourceforge, however this did not work on my USB drive - it failed to find the **menu.lst** configuration file.

Repeat the following steps from the Syslinux method:

Step 1

Step 2a

Step 2b

Ignore **Step 3** and **Step 4**

To install **GRUB4DOS**:

- * Download latest **GRUB4DOS** package and extract contents to a temporary location.
- * Install an NT boot sector to the USB partition using Bart's mkbt.exe (**mkbt.exe** and bootsector **pe2usb.bin** are included with pebuilder) - e.g.

CODE

```
mkbt -x pe2usb.bin F:
```

Where pe2usb.bin is bootsector and F: is the partition on USB stick.

* Now create a **boot.ini** file on the USB stick, with the following code:

CODE

```
[boot loader]
timeout=10
default=C:\grldr
[operating systems]
C:\grldr="Grub4Dos"
```

- * Copy **NTDETECT.COM** and **NTLDR** from a working Windows XP installation to the USB stick.
- * Copy **GRLDR** from the **GRUB4DOS** package downloaded earlier to the USB stick.

* Create **menu.lst** file (using notepad) on the USB stick with following text - this is the GRUB configuration file.

CODE

```
color white/red black/red
default 0
timeout 10
```

To add bootable floppy disk images

Repeat **Step 5**, but ignore instructions to add entries in menu.txt and syslinux.cfg. To add Win98.ima to GRUB add the following lines to menu.lst:

CODE

```
title Windows 98 boot disk
find --set-root /grldr
kernel /memdisk
initrd /win98.ima
```

To add Memtest86.ima to GRUB add the following lines to menu.lst:

CODE

```
title Memtest86
find --set-root /grldr
kernel /memdisk
initrd /Memtest86.ima
```

To add Puppy Linux

Repeat **Step 6**, but ignore instructions to add entries in menu.txt and syslinux.cfg. Add following lines to menu.lst:

CODE

```
title Linux Puppy
find --set-root /grldr
kernel /vmlinuz root=/dev/ram0 initrd=initrd.gz PMEDIA=usbflash PKEYS=uk
initrd /initrd.gz
```

To add DSL (Damn Small Linux)

Repeat **Step 7**, but ignore instructions to add entries in menu.txt and syslinux.cfg. Add following lines to menu.lst:

CODE

```
title Damn Small Linux
find --set-root /grldr
kernel /linux24 ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=791 initrd=minirt24.gz nomce noapic qemu
quiet BOOT_IMAGE=knoppix frugal
initrd /minirt24.gz
```

To add gparted

Repeat **Step 8**, but ignore instructions to add entries in menu.txt and syslinux.cfg. Add following lines to menu.lst:

CODE

```
title GParted
find --set-root /grldr
kernel /linux noapic initrd=gpinitrd.gz root=/dev/ram0 init=/linuxrc ramdisk_size=65000
initrd /gpinitrd.gz

title GParted (Use if RAM < 64MB)
find --set-root /grldr
kernel /linux noapic initrd=gpinitrd.gz root=/dev/ram0 init=/linuxrc ramdisk_size=16384
initrd /gpinitrd.gz
```

To add BartPE

Repeat **Step 9**, but ignore instructions to copy and rename pe2usb.bin. As ntlldr already exists on the USB stick (and needs to remain there to load grldr) don't rename setupldr.bin (from w2k3 sp1) - GRUB can chainload this file directly. Also ignore instructions to add entries in menu.txt and syslinux.cfg. Add following lines to menu.lst:

CODE

```
title BartPE
find --set-root /grldr
chainloader /setupldr.bin
```

To add another BartPE

Repeat **Step 9a**, but ignore instructions to copy and rename pe2usb.bin. Also ignore instructions to add entries in menu.txt and syslinux.cfg. Add following lines to menu.lst:

CODE

```
title BartPE 2
find --set-root /grldr
chainloader /ntld2
```

To add Windows PE 2.0

Oops! At present GRUB4DOS does not support directly chainloading **bootmgr** (WinPE 2.0's equivalent of setupldr.bin/ntldr). The only way to add **WinPE 2.0** is to create a separate partition with bootsector (refer to method 2 below) and then chainload the partition using the following entry in menu.lst:

CODE

```
title WinPE 2.0
find --set-root /sources/boot.wim
chainloader +1
```

Completed **menu.lst**:

CODE

```
color white/red black/red
default 0
timeout 10

title Windows 98 boot disk
find --set-root /grldr
kernel /memdisk
initrd /win98.ima

title Memtest86
find --set-root /grldr
kernel /memdisk
initrd /Memtest86.ima

title Linux Puppy
find --set-root /grldr
kernel /vmlinuz root=/dev/ram0 initrd=initrd.gz PMEDIA=usbflash PKEYS=uk
initrd /initrd.gz

title Damn Small Linux
find --set-root /grldr
kernel /linux24 ramdisk_size=100000 init=/etc/init lang=us apm=power-off vga=791 initrd=minirt24.gz nomce noapic qemu
quiet BOOT_IMAGE=knoppix frugal
initrd /minirt24.gz

title GParted
find --set-root /grldr
kernel /linux noapic initrd=gpinitrd.gz root=/dev/ram0 init=/linuxrc ramdisk_size=65000
initrd /gpinitrd.gz

title GParted (Use if RAM < 64MB)
find --set-root /grldr
kernel /linux noapic initrd=gpinitrd.gz root=/dev/ram0 init=/linuxrc ramdisk_size=16384
initrd /gpinitrd.gz
```

```

title BartPE
find --set-root /grldr
chainloader /setupldr.bin

title BartPE 2
find --set-root /grldr
chainloader /ntld2

title WinPE 2.0
find --set-root /sources/boot.wim
chainloader +1

```

So why bother with GRUB if Syslinux works? IMHO there are advantages and disadvantages to each. I've personally found Syslinux to be more stable and user friendly than GRUB, however as USB sticks get bigger we will eventually run into problems as, according to the documentation **...SYSLINUX is unsafe to use on any filesystem that extends past cylinder 1024...**

GRUB has a lot of features and can boot XP/BartPE/WinPE from a non first hard disk by remapping the drive containing the OS as 1st Hard Drive. I've not had any stability problems since switching to version 0.4.2pre11. GRUB can also access subfolders, which will enable you (with the correct entries in menu.lst) to clean up the root of the drive. GRUB can also be used with method 2 to save the need to install a boot manager to MBR (see Step 9a below).

Method 2 - Using multiple partitions

In Windows XP a USB stick is usually mounted as a removable disk - even if it contains multiple partitions only the first will be accessible. To access other partitions on the USB stick you will need to mount the stick as a fixed drive. This can be done using the **Hitachi Microdrive Filter** (cfadisk.sys). The following link contains a detailed walkthrough and was used to compile most of this guide:
<http://www.msfm.org/board/lofiversion/index.php/t69211.html>

You will need to download the following:

- * <http://www.xpefiles.com/viewtopic.php?t=92> - download from xpefiles
- * <http://h18000.www1.hp.com/support/files/serveroptions/us/download/20306.html> (HP USB Disk Storage Format Tool) - used Version 2.0.6
- * <http://syslinux.zytor.com/download.php> - used Version 3.31
- *
- <http://www.microsoft.com/downloads/details.aspx?amp;displaylang=en&familyid=22CFC239-337C-4D81-8354-72593f> (if using BartPE)
- * OS's you would like to include (BartPE, WindowsPE etc)
- * <http://spfdisk.sourceforge.net/Spf2K3rE.exe>

Step 1 - Planning

Plan which Operating Systems you want to include + how many partitions are needed. I used a 1GB Kingston Datatraveller 2.0 (actual size = 984MB) with 4 primary partitions (all partitions formatted using FAT file system):

```

Partition 1 = 596MB -> Syslinux partition
Partition 2 = 196MB -> WindowsPE 2.0 partition
Partition 3 = 172MB -> BartPE partition
Partition 4 = 15MB -> MSDOS partition

```

Remember only partition 1 will be accessible from a Windows XP PC (unless the **Hitachi Microdrive Filter** is installed). This should therefore be the largest partition and should also be used for data storage.

Step 2 - Installing Microdrive Filter

Download the filters from XPfiles and extract the contents to a hard drive. Open cfadisk.inf and edit the following section:

CODE

```
[cfadisk_device]
%Microdrive_devdesc% = cfadisk_install,IDE\DiskIBM-DSCM-11000_____SC2IC801
%Microdrive_devdesc% = cfadisk_install,IDE\DiskIBM-DSCM-11000_____SC2IC815
%Microdrive_devdesc% = cfadisk_install,IDE\DiskIBM-DSCM-11000_____SC2IC915

%Microdrive_devdesc% = cfadisk_install,IDE\DiskIBM-DSCM-10512_____SC1IC801
%Microdrive_devdesc% = cfadisk_install,IDE\DiskIBM-DSCM-10512_____SC1IC815
%Microdrive_devdesc% = cfadisk_install,IDE\DiskIBM-DSCM-10512_____SC1IC915

%Microdrive_devdesc% = cfadisk_install,IDE\DiskIBM-DMDM-10340_____MD2IC501
%Microdrive_devdesc% = cfadisk_install,IDE\DiskIBM-DMDM-10340_____MD2IC601

; debug on VMWare/special drive
; %Microdrive_devdesc% = cfadisk_install,IDE\DiskVMware_Virtual_IDE_Hard_Drive_____00000001
; %Microdrive_devdesc% = cfadisk_install,IDE\DiskIC25N040ATCS04-0_____CA4OA71A
```

to

CODE

```
[cfadisk_device]
%Microdrive_devdesc% = cfadisk_install,
```

Now add ID specific to your usb stick. To find the ID insert your USB stick and use REGEDIT to check the value for your device in **HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\USBSTOR** registry key

For Kingston DataTraveler the key is

HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\USBSTOR\Disk&Ven_Kingston&Prod_DataTraveler

There will be an entry under a subkey for **HardwareID**. Open this and there will be a string of values - for the kingston Datatraveller values are:

CODE

```
USBSTOR\DiskKingstonDataTraveler_2.0PMAP
USBSTOR\DiskKingstonDataTraveler_2.0
USBSTOR\DiskKingston
USBSTOR\KingstonDataTraveler_2.0P
KingstonDataTraveler_2.0P
USBSTOR\GenDisk
GenDisk
```

We only need to use the first value in the HardwareID string - in this case

USBSTOR\DiskKingstonDataTraveler_2.0PMAP. Now copy this value into the [cfadisk_device] section in cfadisk.inf. My new cfadisk.inf contains the following code:

CODE

```
[Version]
Signature="$Windows NT$"
Class=DiskDrive
ClassGuid={4D36E967-E325-11CE-BFC1-08002BE10318}
Provider=%HGST%
LayoutFile=layout.inf
DriverVer=12/14/2002,1.00.0000.1

[Manufacturer]
%HGST% = cfadisk_device

[DestinationDirs]
cfadisk_copyfiles=12 ; %SystemRoot%\system32\drivers
gendisk_copyfiles=12 ; %SystemRoot%\system32\drivers

[cfadisk_copyfiles]
cfadisk.sys

[gendisk_copyfiles]
disk.sys

; -----
```

```
[cfadisk_device]
%Microdrive_devdesc% = cfadisk_install,USBSTOR\DiskKingstonDataTraveler_2.0PMAP

[cfadisk_addreg]
HKR,,"LowerFilters",0x00010008,"cfadisk"

[cfadisk_install]
CopyFiles=cfadisk_copyfiles,gendisk_copyfiles

[cfadisk_install.HW]
AddReg=cfadisk_addreg

[cfadisk_install.Services]
AddService=disk,2,gendisk_ServiceInstallSection
AddService=cfadisk,,"cfadisk_ServiceInstallSection

[gendisk_ServiceInstallSection]
DisplayName = "Disk Driver"
ServiceType = 1
StartType = 0
ErrorControl = 1
ServiceBinary = %12%\disk.sys
LoadOrderGroup = SCSI Class

[cfadisk_ServiceInstallSection]
DisplayName = "CompactFlash Filter Driver"
ServiceType = 1
StartType = 3
ErrorControl = 1
ServiceBinary = %12%\cfadisk.sys
LoadOrderGroup = Pnp Filter

; -----

[Strings]
HGST = "Hitachi Global Storage Technologies"
Microdrive_devdesc = "Hitachi Microdrive"
```

Once cfadisk.inf has been edited with your ID string you will need to update the driver in windows. Open Device manager and check for your USB drive in "Disk Drives". Right click on the entry for your drive and select option to update driver.

- * Don't use windows update
- * Install from a list or specific location
- * Don't search. I will choose the driver to install
- * Have disk
- * Browse to location of edited cfadisk.inf

Update and restart windows

If the driver installed successfully your USB stick should now be mounted as a **Local Disk** not **Removable Disk**

Step 3

Format the disk using HP's USB Disk Storage Format Tool (refer to **Method 1 - Step 2**)

Step 4

Use XP's Disk Management to delete the partition you've just created. Now use Disk Management to create the partitions planned during Step 1.

Step 5

Install boot sector to syslinux partition (refer to **Method 1 - Step 3**) and copy syslinux files, disk images and OS files to the partition. I have included the OS's and images detailed in Method 1 - but have removed

BartPE and WindowsPE.

Step 6

Install boot sector to Windows PE 2.0 partition (refer to **Method 1 - Step 10** for instructions on creating bootsector) using mkbt.exe. - e.g.

CODE

```
mkbt -x WinPE2.bin E:
```

Where WinPE2.bin is bootsector and E: is Windows PE 2.0 partition on USB stick.

Now copy PE 2.0 files to Windows PE 2.0 partition on USB stick.

Step 7

Install boot sector to BartPE partition using mkbt.exe (bootsector **pe2usb.bin** is included with pebuilder) - e.g.

CODE

```
mkbt -x pe2usb.bin F:
```

Where pe2usb.bin is bootsector and F: is BartPE partition on USB stick.

Now copy BartPE files to BartPE partition on USB stick.

Step 8

Install MSDOS bootsector to MSDOS partition using mkbt.exe and copy files to the partition.

Step 9a

If GRUB4DOS is installed instead of (or as well as) Syslinux **Step 9b** can be ignored and GRUB can be used instead of installing a boot manager to MBR. Just add the following entries to the GRUB configuration file

menu.lst:

CODE

```
title WinPE 2.0
chainloader (hd0,1)+1

title BartPE
chainloader (hd0,2)+1

title DOS
chainloader (hd0,3)+1
```

These entries will only work if the USB stick is booted as 1st Hard Disk as the chainloader entries are hardcoded to a particular drive and partition. Alternatively the following **menu.lst** entries can be used:

CODE

```
title WinPE 2.0
find --set-root /sources/boot.wim
chainloader +1

title BartPE
find --set-root /PE.SDI
chainloader /ntldr

title DOS
find --set-root /AUTOEXEC.BAT
chainloader +1
```

These entries all search for a unique file and then set the (1st) drive containing that particular file as root. This might not work properly if the file is present on more than one drive

Step 9b

Install a boot manager - I have successfully tested **BootItNG** and **XOSL** boot managers however the main drawback with these programs is the need to use a separate partition for the boot files, as they will not fit in the MBR. **SPFDISK** fits in the MBR and is able to boot all OS's I have tested on it - I will therefore use SPFDISK in this guide.

Run and install SPeial FDISK (SPFDISK). The easiest way to do this is to include SPFDISK in an MSDOS boot disk - then boot from this disk. If you do not have a floppy drive installed on your computer then include SPFDISK in a disk image on the syslinux partition - then boot the USB drive from this (via syslinux - memdisk kernel). If booting via syslinux use XP's Disk Management to make sure that the syslinux partition is active. *NOTE* - Before running SPFDISK I would advise disabling any other drives in the BIOS to ensure that any changes are not made to existing HDD's. If you do not want to do this then at least ensure that the USB stick is listed as the first HDD in the BIOS settings.

- * Select Option **A. Append Boot Item**
- * Select your USB stick from the list
- * Select partition (this will be the first option in your boot menu).
- * Enter a name for this partition (which will be displayed in your boot menu)

Repeat the above for each partition on your USB stick.

- * When you have added all partitions to the boot menu choose option **G. Begin Install**
- * Select option **2.MBR**
- * Select your USB stick from the list

Now reboot your PC and change Bios settings to boot from your USB Drive. If everything went according to plan you should see a boot menu.

Additional Notes For Hitachi MicroDrive Filter

I have been able to get the Kingston USB stick used to test this guide to mount as a fixed disk in WindowsPE 2.0 but have so far been unsuccessful in repeating this using BartPE. To mount as a fixed drive in Windows PE 2.0:

From a WindowsXP installation (with the Hitachi Microdrive Filter installed) export the following registry keys:

- * **HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\cfadisk**
- * **HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\USBSTOR\Disk&Ven_...** - import the settings for you own USB stick
- * Use notepad to open the two files you've just exported and replace all instances of **HKEY_LOCAL_MACHINE\SYSTEM** with **HKEY_LOCAL_MACHINE_PE_SYSTEM**

Now complete the following steps:

- * **Open [b]Windows PE Tools Command Prompt** (from start menu)
- * Type **copyype.cmd x86 c:\winpe_x86** - to create WindowsPE files
- * Type **imagex /mountrw c:\winpe_x86\ISO\sources\boot.wim 1 c:\winpe_x86\mount** - to mount the image.
- * Open **Regedit** - select **HKLM** key and then select **Load Hive** (from File TAB).
- * Now browse to **c:\winpe_x86\mount\windows\system32\config** directory and select **SYSTEM**
- * When prompted for a key name type **_PE_SYSTEM**
- * Import the two files we edited earlier
- * Unload the **_PE_SYSTEM** hive
- * Close registry editor
- * Copy cfadisk.sys (from Hitachi MicroDrive Filter download) to **c:\winpe_x86\mount\windows\system32\drivers**
- * In the **Windows PE Tools Command Prompt** type **imagex /unmount c:\winpe_x86\mount /commit** - to unmount and save changes to **boot.wim**
- * Copy the files from **c:\winpe_x86\ISO** to the Windows PE partition on your USB stick
- * Now reboot your PC and change Bios settings to boot from your USB Drive.

On booting from the PE 2.0 files on my USB stick all four partitions were visible (and mounted as Local Disks) in my file manager.

I hope that someone finds this guide useful and that it adds to existing information.

I would like to get some feedback re which Motherboards and USB sticks work. Please post any successful combinations.

Many thanks to the authors and all who posted in the links I have included. Thanks also to flyakite's original "Creating a Multi-Boot DVD" guide, which got me started on multi-booting.

Last but definately not least, many thanks to Bart Lagerweij for making pebuilder available.

Posted by: jaclaz Nov 10 2006, 05:36 AM

VERY good!

You did an exceptionally good work in putting together info scattered all around the forum. 😊

If I may, to "squeeze" some more the disk space occupied by floppy images, remember that memlinux is "gzip" compatible, this means that you can use any .zip compatible compression program to compress floppy images (or use Winimage compressed format, which is 100% compatible).

Of course this gives a very sensible advantage when 1.44 Floppy images are not "full to the brim", just as an example a BCDL 2.01a image, once compressed, occupies 10.896 bytes, (12.288 bytes on disk), with a compression rate of $1.474.560/12.288 = 120!$

However an "average" filled floppy image most of the time takes less than 600 kbytes.

I found that using 7-zip in .zip compression mode gives slightly better results.

[BEGIN SLIGHTLY OUT OF TOPIC] (you can ignore the following)

If you have some time (and will) you could replicate this guide using Grub4dos instead of syslinux (this has the advantage that you do not need to install anything on the MBR of the stick, as you can invoke Grub4Dos from DOS using Grub.exe and that you do not need to create bootsector imahes like pe.bss as you can directly chainload your(s) SETUPLDR.BIN(s)).

The procedure however is very similar, you can use memdisk as well, entries in menu.lst (instead of syslinux.cfg would change from (just an example):

CODE

```
LABEL 1
KERNEL memdisk
APPEND initrd=win98.ima
```

to

CODE

```
TITLE 1 Win98.ima through memdisk
KERNEL /memdisk
INITRD /Win98.ima
```

[/END SLIGHTLY OUT OF TOPIC] (you can ignore the above)