



## Cypress DOS Driver User's Guide

### Getting Started

#### Included Files

The DUSE USB Driver software is delivered as a single executable (.EXE) file, DUSE.EXE. The DUSE.EXE file supports USB hardware that meets the Open Host Controller Interface (OHCI) specification, the Universal Host Controller Interface (UHCI) Specification, and the Enhanced Host Controller Interface (EHCI) specifications.

#### Device Support

This version provides support for the following Mass Storage devices:

- USB ZIP drives
- USB Hard disk
- USB Super Floppy/HiFD drives
  - Some floppy disks require a default sector size of 1024 or 2048 bytes. See the SECTORSIZE command line parameter for details on how this can be set.
- USB CD-ROM
- USB MO drives
  - 640K and 1.3MB disks require a default sector size of 2048 bytes. See the SECTORSIZE command line parameter for details on how this can be set.

### Installing the DOS Driver

#### Copy the driver file

Create a directory on your boot device to store the executable driver files. As an example:

```
C:\> mkdir duse
C:\> copy sourcefiles c:\duse
```

#### Config.sys

Add the following text to your config.sys file:  
DEVICE=C:\DUSE\DUSE.EXE [options]

#### Command-Line Parameters

DUSE allows the following command line parameters. Only the first characters before [.] are required.

- VER[BOSE]=[x], Enables status messages. Optional valid values for .x. are 0, 1, and 2. Value of 0 disables support of status messages. Value of 1 enables text based status messages. Value of 2 enables a pop-up window, which displays the status messages. Default value is 0. If the VERBOSE is used without a .x. value, the default is 2.
- WAI[T]=x, Sets the number of seconds to wait for a Pop-Up Windows to be displayed before automatically removing it. Valid values for .x. are 0 through 255. Default value is 3. Only valid if VERBOSE is set to 2 (Pop-Up status messages enabled).
- DRI[VES]=x, Sets the number of USB drives allowed, where .x. is the number of drive letters. Valid values for .x. are 0 through 3. Value of 0 disables support for USB drives. Default value is 1.
- NOD[RIVES], Disables support for USB drives. This will override the DRIVES parameter.
- NOC[DROM], Disables support for USB CD-ROM drives.
- MEM[POOL]=x, Sets the size of additional memory pool allocated, where .x. is the size in Kbytes. DUSE performs a runtime calculation, based on the number of controllers, the size of transfers, etc., to determine the optimal memory pool size. Setting MEMPOOL adds to the memory pool size. Valid values are 0 thru 128.
- XFER[SIZE]=x, Sets the maximum allowed transfer buffer size, where .x. is the transfer buffer size in Kbytes. Increasing the transfer buffer size will increase performance, but also requires a larger memory pool. Default value is 16.
- SEC[TORSIZE]=x, Sets the default sector size for USB drives, where .x. is the default sector size in bytes. Valid values for .x. are 512, 1024, and 2048. Default value is 512.
- APM[STATE], Sets the default Advanced Power Management (APM) level for the USB devices in which APM is enabled. Device

performance may increase with increasing power management level and power consumption may increase with increasing power management levels. The following lists the valid APM states:

- Maximum performance FEh
- Intermediate power management levels without Standby 81h-FDh
- Minimum power consumption without Standby 80h
- Intermediate power management levels with Standby 02h-7Fh
- Minimum power consumption with Standby 01h

The Default value for APM is 0x7F.

- INT[13], Enables INT13 support for fixed disks. This adds additional support for some disk utilities like FDISK.
- DMA[SAFE], Allocates a DMA safe buffer which will be used for data transfer. This parameter is required for applications that might use buffers allocated in extended memory.
- NOU[HCI], Prevents initialization of UHCI controllers.
- NOO[HCI], Prevents initialization of OHCI controllers.
- NOE[HCI], Prevents initialization of EHCI controllers.
- UHCN=x, Specifies that DUSE will initialize a maximum of x UHCI controllers.
- OHCN=x, Specifies that DUSE will initialize a maximum of x OHCI controllers.
- EHCN=x, Specifies that DUSE will initialize a maximum of x EHCI controllers.
- LATE[INIT], Enables late initialization. When using late initialization, the driver is loaded during the config.sys phase of DOS initialization, but it is not enabled until the INIT command-line is invoked, typically near the end of autoexec.bat initialization. See the sections on Late Initialization and Using DUSE as a Program, Command Line Parameters below for more information on late initialization.
- EBAR=x, Specifies the address offset where the EHCI Base Address Register (BAR) should be placed in memory. .x. must be a

four digit hexadecimal number with a prefix of .0x.. i.e. to place the EHCI BAR at 0xE0000, the correct command line switch would be EBAR=0xE000. The acceptable addresses are in the range of 0xA000 through 0xF400.

- OBAR=x, Specifies the address where the OHCI BAR should be placed in memory. The description of EBAR (above) applies to OBAR.
- UBAR=x, Specifies the address where the UHCI IO Base Address should be placed in memory. The description of EBAR applies to UBAR, but the valid address range is 0x0400 through 0xF400.
- CDW[AIT][=x], Specifies that DUSE initialization will NOT complete until the first USB CDROM device has been mounted, OR until x seconds have elapsed. Valid values for .x. are 0 through 30. x=0 is a special case for DUSE as it instructs DUSE to wait forever for the first CDROM device to be mounted, OR until the ESCAPE key is pressed. If x is not specified, it defaults to x=0.
- CDD[EVICE]=<device name>, Allows you to specify the name of the USB CDROM device driver. The default is .USBCDROM.
- VFLOP[PY], Creates a virtual image of the boot floppy at initialization.

The following is an example of using the DOS driver command line parameters. Consider this line added to your config.sys file:

```
DEVICE=C:\DUSE\DUSE.EXE VERBOSE
DRIVES=2 XFER=8 SEC=2048 NOCD
```

This example will:

- Display pop-up status messages
- Setup two USB drives
- Set the transfer size to 8K
- Set the default sector size to 2048 bytes
- Disable CD-ROM support

## Using the Driver

### Loading the Driver

After editing the config.sys file, reboot the DOS system. The driver will be loaded at boot time and will display a message similar to the following:

```
DUSE version X.X Build XXXX
DOS USB Stack with USB Storage Extensions
Copyright Cypress Semiconductor, 2000-20002
```

```
UHCI USB Controller initialized (BASE 0xXXXX[,
IRQ XX])
```

### USB Hub

A USB Hub provides additional down stream ports to allow multiple devices to be connected to a single root hub. As additional devices are attached to the downstream port of a USB Hub, the devices will be started in the same fashion as devices attached directly to the root hub.

### Mass Storage

When attached to a USB port, a USB Mass Storage device will automatically mount and get assigned the next available drive letter. For example, if a single drive exists on a DOS system C:, and a USB drive is inserted, then the DUSE stack will assign D: to the USB driver. The drive will be accessible in the same manner as regular drives in a DOS system. If the device is removed and then reinserted, the DUSE will reassign the D: drive letter to the USB Mass Storage device and the device will become accessible once again.

USB CD-ROM drives require loading the Microsoft CD Extension (MSCDEX) driver. Assuming a default device name of "USBCDROM", the command line will look like:

```
MSCDEX /D:USBCDROM /S /M:15 /V
```

MSCDEX will return the drive letter where the USBCDROM media is mounted.

### Late Initialization

When DUSE is loaded as a driver in config.sys, using the LATE[INIT] option, it must be initialized prior to using any USB devices. The late initialization is performed by invoking DUSE again, this time from a command prompt or from a batch file, and specifying INIT on the command line.

## Using DUSE as a program

### Command Line Parameters

The following parameter is recognized when running DUSE as a program (from the command prompt or from a batch file, including AUTOEXEC.BAT).

- INIT, This causes the DUSE executable to communicate an initialization command to the DUSE device driver. Host controllers will be initialized and any recognized devices on the bus will be enumerated.

## The DUSELDR Utility

### Loading DUSE.EXE from the command line

The DUSELDR utility will load DUSE.EXE from the command line instead of loading it in the config.sys file. Loading DUSE.EXE from the command line would look as follows:

```
C:\> C:\DUSE\DUSELDR.EXE C:\DUSE\DUSE.EXE
VERBOSE XFER=64 NOCD
```



**Document Revision History**

Revision #	Date	Comments
1.0	4/17/2002	Initial public version, covering DUSE 4.2