

Contact Addresses

Contact the authors through the internet:

WWW:

<http://come.to/ide64>

<http://singularcrew.hu/ide64warez/>

E-mail:

josef.soucek@ct.cz (IDEDOS)

dundera@volny.cz (hardware, manager)

soci@c64.rulez.org (software, cd, warez, IDEDOS etc.)

This document is a subject of change, so update to the newest version of it to stay informed about the latest features.

This manual was created on 2001.Feb.12. with IDEDOS v0.89 in mind and updated on 2002.Okt.11.

If you have no internet access, or you want to order anything related to IDE64, contact Protovision at

Jakob Voos
Niersstr. 1
40547 Duesseldorf
Germany

contact@protovision-online.de
www.protovision-online.de

Attention: Names, technologies and standards used in this document may be protected by author's rights or patents.

This manual was created by the IDE64 crew and Protovision in 2002.

IDE64 HARDDRIVE/CDROM CONTROLLER

for
Commodore 64/128

instructions manual



Appendix B

Description of IDEDOS error messages

00, OK,00,00

Everything seems to be ok.

23, READ ERROR,00,00

Media error/device timeout. Quite common when using buggy CDs.

26, WRITE PROTECT ON,00,00

File/device is write protected.

29, DISK ID MISMATCH,00,00

CD in drive was replaced.

31, SYNTAX ERROR,00,00

Unknown/not implemented command.

62, FILE NOT FOUND,00,00

File couldn't be found.

63, FILE EXISTS,00,00

File/directory already exists.

64, FILE TYPE MISMATCH,00,00

Usage of an unknown filetype/or a try to change a directory into a file.

73, IDE DOS Vx.xx IDE64,00,00

Identify string for harddrive.

73, IDE DOS Vx.xx CDROM,00,00

Identify string for cdrom.

74, DRIVE NOT READY,00,00

No disk in cdrom.

80, HDD ERROR,00,00

Other errors, which don't have a number yet, like dir not empty.

Appendix A

Description of the harddrive Basic error messages

?HD ERROR

READY.

There is a problem with the IDE harddrive (hardware or configuration problem).

?LAST DIR SECTOR ERROR

READY.

The controller detected a sector problem.

?FILE IS NOT DIR ERROR

READY.

The requested directory name exists on the drive as a standard file name.

?CAN'T LOAD DIR ERROR

READY.

The requested file name exists on the drive as a directory name.

?WRITE PROTECT ERROR

READY.

This message is generated when the controller attempts to delete a protected file.

?FILE TYPE ERROR

READY.

Similar to FILE TYPE MISMATCH

?BAM ERROR

READY.

There is a problem in the BAM (Block Availability Map). To correct this problem, initialize the harddrive or use IDE ScanDisk.

?DIR IS NOT EMPTY ERROR

READY.

The controller cannot delete a directory which is not empty.

?FILE DIR EXIST ERROR

READY.

The file name or directory already exists on the current path.

Table of Contents

Introduction to the IDE64 cartridge	2
Updating the firmware	2
IDE64 Peripherals	2
Installation	3
Setup Menu	4
File Manager	6
Using Manager PlugIns	7
PCLink Data Transfer	8
IDE64 BASIC Extentions	9
IDE64 Kernal Extentions	10
Harddrive Monitor	11
Machine Code Monitor	11
Programmer's Reference	12
- Programming in BASIC	12
- Programming in Machine Language	13
- Creating Manager PlugIns	14
- IDE64 card detection	15
- Calling simple BASIC commands in ML	15
Appendix A (BASIC error messages)	16
Appendix B (IDEDOS error messages)	17
Contact Addresses	18

Introduction to the IDE64 cartridge



The IDE64 cartridge is a device for connecting IDE harddisks and ATAPI CDROM drives to a C64. It's plugged into the Expansion Port and connected to ATA(PI) devices by a 40 wire cable. The IDE64 cartridge contains a 64kB PEROM (flash EPROM, with IDE-DOS, Machine Code Monitor, File Manager and Setup), a 28kB RAM (used for internal buffers), a realtime clock chip (powered by a battery), two LEDs (to indicate the presence of the cartridge and HDD activity), a Short-BUS for peripherals (like DUART Card) and an ispLSI chip.

The firmware of the IDE64 is installed in its PEROM. This means, that the firmware is not burned to an inflexible EPROM, but it can be altered whenever you like by "reflashing" the firmware. On the IDE64 homepage/Singular Crew IDE warez page there are many versions of the firmware. Download the one that fits your needs and just flash them onto your cartridge.

Updating the Firmware

For reflashing your system firmware, you need a .per-file that includes the firmware and the PEROM utility. Connect the two loose pins right to the Lattice-chip with a jumper or a screwdriver, then reset. If you succeeded in connecting them, the green LED lamp will start flashing. Now disconnect the pins again and load the PEROM utility. It will prompt you to enter the filename of the .per-file. Enter it and wait for the new firmware to be installed! When the utility has done its job, reset the computer once more and you are using the updated firmware.

IDE64 Peripherals

IDE64 has a free port for plugging other new generation peripherals to it. The only peripheral available right now is the high-speed RS232-Interface DUART. It can serve for fast internet connections and a fast PC-Link connection at one time as it comes with 2 COM-ports.

Currently, a MP3-board and an ethernet card for IDE64 are under development.



IDE64 card detection

This routine can detect the IDE64 interface with IDE-DOS V0.88 and higher.

```
$de60-$de62 = $49,$44,$45 "IDE"
    lda #$3f
    sta $00
    lda #$37
    sta $01
    lda $de60
    cmp $de60
    bne noide64
    cmp #$49
    bne noide64
    lda $de61
    cmp $de61
    bne noide64
    cmp #$44
    bne noide64
    lda $de62
    cmp $de62
    bne noide64
    cmp #$45
    bne noide64
    beq ide64found
```

Calling simple BASIC commands in assembler

This sourcecode demonstrates how to call simple BASIC commands through machine code language.

```
lda #<restart
sta $0302
lda #>restart
sta $0303
lda #$f5 ;$F0 = LL; $F1 = DIR; $F5 = HDINIT
sta $0200
lda #$00
sta $0201
lda #<$01ff
sta $7a
lda #>$01ff
sta $7b
jmp ($0308)
restart = *
```

Creating Manager 'plugins'

A plugin has 2 start addresses -> \$1000 and \$1003

\$1000 - checks for filename - should it run independantly from the manager?

\$1003 - the manager uses this address to start the file

Do not destroy any memory at \$0000-\$03FF, \$0800-\$0FFF!!! Don't forget to restore the VIC before exiting. Not a bad idea to load the file first, and then change the screen.

```

                *= $1000
start           = *
                jmp called_by_sys4096

;called_by_manager
                jsr setnam
                lda #1
                ldx $ba
                ldy #0
                jsr setlfs
                jsr open

                ldx #1
                jsr chkin
read_file_loop = *
                jsr chrin
                ;...
                ;...
                ;...
                ldx $90
                beq read_file_loop

                lda #$01 ;Close file and return
                jsr close ;to the FileManager
                jsr clall

                ;...
                ;Do some nice things...
                ;...

                rts ;return to manager
;-----
called_by_sys4096 = * ;Ask for filename, display credits, etc.
                rts
```

Installation

Before plugging the IDE cartridge into the Expansion port turn off your C64.

Connect the IDE harddisk with the cartridge using the 40 wire flat ribbon cable aligning pin 1 of the cartridge with pin 1 of the hard disk drive (pin 1 on cartridge is on the upper right hand side near RESET button). Make sure that the hard disk drive is correctly configured. If you are using only one harddisk, switch to single mode. If you are using two harddisks, switch the first one to master and second one to slave. Connect the hard disk drive to the power supply and then turn on hardisk(s) and C64. Wait until READY appears on the screen. Before using hard disk drives the first time you must format them. Load end run the program IDEformat. Follow the instructions given in the program. After the hard disk drive is formatted RESET the C64 and type HDINIT (one HDD) or HDINIT1 (two HDDs).

CDROMs are ready to use if inserted CDs are in ISO9660/Joliet format. (all data CDs are in these formats ;-)

To back-up the real time clock and IDE64 configuration you can optionally insert a battery CR2032 into the IDE64 controller while the C64 is turned off. Be careful of polarity!!

If you receive the unassembled controller, use the diagram when constructing IDE64. Work carefully when soldering CMOS components. If you think you're talented enough, you can fit the cartridge inside a C64C under the keyboard, connecting it with ribbon cables to expansion port. The IDE cable has the same size as the tape port, so let's pass though it that cable. Take care about not to damage the ribbon cables placed under the mainboard by it's pins! Drill some nice holes for the HDD activity leds and reset button where you think they look best. Short bus has also the same size as the tape port, and if you don't own/won't use such device then it's the right place for it, with a nice long cable. (it's not mandatory to build the short bus) Don't solder the short bus slot on the cartridge board, if you want to fit it under the keyboard! (the mainboard expansion slot is free, because we soldered the ribbon cables directly under the mainboard to it's pins, and the short bus is at the tape port..) Forget to say that there's also no space for the battery holder on the solderside of the cartridge, place this somewhere else, you can also replace it with two AAA size 1.5V longlife batterys for military usage ;-)

If you're a lucky owner of a low power 2.5" laptop HDD then it's worth to mention that it's so small that it also fits under the keyboard with the IDE64 cartridge, and no additional power supply is needed. (Next project is a laptop CDROM inside the case, if I can get my hands on one) For more informations about integrating the IDE64 cartridge mail Soci/Singular. And don't forget to create a 'Powered by IDE64' sticker ;-)

Setup Menu

The setup utility is contained in the ROM of the IDE controller. Start it by typing Escape (arrow left) together with the RESTORE key. The setup consists of 2 sections: the Standard Setup and the Advanced Setup. Use the cursor keys and RETURN, and press Run/Stop to return to the main menu.



Standard Setup:

In Standard Setup you can change logical device numbers, screen colors, floppyspeeder and set the real time clock.

Clock setting is done using +/- . The day of week calculation is automatic. If the clock is stopped in the setup and shows 2000.jan.01 00:00:80 after switching on, and settings getting lost frequently, then you'd better check the battery on the backside of the cartridge!

The 'Floppy drive 8' option selects the logical drive number for your physical drive 8. This allows for example changing your floppy's drive number without setting the jumpers on the device.

The 'HDD0' and 'HDD1' options control the logical drive number of the devices attached to the ATA interface on the IDE64 card. For example if there's a HDD as master, and a CDROM as slave connected to the IDE64 card, and HDD0 is set to 12, HDD1 to 13, then the HDD will be drive #12, the CDROM drive #13.

The 'PC-LINK' option controls the drive number of the virtual PC-LINK drive. This functionality requires a parallel/serial cable, and a server program on a pc. To determine which cable you may use check the first line on the top. In this case it says 'DUART', which means serial transfer using the duart card. If you want to use parallel cable, then upgrade your IDEDDOS to the parallel version.

The 'Paper color', 'Border color', and 'Char color' allows overriding the default system colors.

Programming in Assembler

Use the same syntax as in BASIC.

And here is how to keep compatible to IDE64:

Do not use floppy speeders to access HDD, non kernel calls direct to serial BUS, undocumented kernel calls, and do not use a fixed device number #8 (Use \$ba instead).

Do not call the RESTORE (\$FF8A/\$FD15) routine. Do not assume that the vectors at \$0300 are always pointing to the same location in \$DE60-\$DEFF, and do not replace them!

Do not use ACPTR (\$FFA5/\$EE13), CIOUT (\$FFA8/\$EDDD), LISTEN (\$FFB1), SECOND (\$FF93), TALK (\$FFB4), TKSA (\$FF96), UNLSN (\$FFAE), UNTLK (\$FFAB), these calls are designed in the kernel for serial bus devices.

When fixing existing programs, watch out for these calls, and replace them with IDE64 compatible ones. It would be nice if after fixing the program would still run with a usual floppy drive...

NONE of the IDEDDOS calls break the IRQ/NMI (it is possible to replace IRQloaders this way!), but IRQs/NMIs can get delayed by 30-40 cycles (like VIC's bad lines) while executing IDE64 calls. This may cause "raster bugs" and similar, to avoid this do the raster interrupt a line earlier, and then check \$D012.

You can safely use these kernel routines:

LOAD - \$FFD5

It's possible to load from ~\$0400 to \$FFFF

(IDE64 switch \$0001 memory configuration register automatically).

SAVE - \$FFD8

It's possible to save RAM from ~\$0400 to \$9FFF (and \$C000-\$CFFF)

OPEN - \$FFC0

CHKIN - \$FFC6

CHRIN - \$FFCF

GETIN - \$FFE4

CHKOUT - \$FFC9

CHROUT - \$FFD2

CLOSE - \$FFC3

CLALL - \$FFE7

CLRCHN - \$FFCC

The most powerful routines for using IDE64 are described in IDE64 kernel extensions. Also take a look at "calling simple BASIC commands in ML"

Programmer's Reference

Programming in BASIC

See IDEDOS BASIC extentions for more IDE64 BASIC commands. Described here are the loading/saving routines:

Open Files for Read or Write:

Read PRG file:

```
10 OPEN 1,12,0,"FILENAME"
```

```
20 GET#1, A$
```

```
...
```

```
XX CLOSE 1
```

Write PRG file:

```
10 OPEN 1,12,1,"FILENAME"
```

```
20 PRINT#1, A$
```

```
...
```

```
XX CLOSE 1
```

Read PRG, SEQ or USR file:

```
10 OPEN 1,12,2,"FILENAME,Type,R"
```

```
20 INPUT#1, A$
```

```
...
```

```
XX CLOSE 1
```

Write PRG, SEQ or USR file:

```
10 OPEN 1,12,2,"FILENAME,Type,W"
```

```
20 PRINT#1, A$
```

```
...
```

```
XX CLOSE 1
```

Type:

P = PRG file

S = SEQ file

U = USR file

It is possible to open up to six channels for reading and one channel for writing at the same time (for one device). (IDE64 DOS V0.70 or higher)

The 'Boot file' option controls the loading & starting of 'BOOT' file from the root directory of the primary harddrive on power-up/reset. If this option is 'Always', then C64 loads 'BOOT' (LOAD"/BOOT",12,1) always and starts it by RUN. If this option is 'Power up' then file is not loaded on reset, just at power up. 'Disabled' turns this feature off. Pressing the 'C=' or 'RUN/STOP' key during power-up/reset skips loading, if it was enabled.

The 'Floppy speeder' controls if LOAD and Manager uses disk turbo for accessing physical drive 8. Floppy Speeder assumes a 1541 as drive 8, if there's another type of drive instead of, then better disable this option. Disk turbo also works on NTSC and is faster than JiffyDos.

The 'Set basic clock' option synchronizes the TI\$/TI basic variables on power up/reset to the realtime clock build in the cartridge, if option enabled.

Advanced Setup:

The Advanced Setup lets change drive specific features, like power management, error retry, write cache, look-ahead, and CDROM speed control on per drive basis. Don't forget that these settings won't take affect until a HDINIT(1) or powerup!

If 'Power management' is enabled, hard disks/cdroms will be turned off after the selected amount of time (1 min - 2 hours) in case of inactivity. Turning on is automatic done at the next disk access. This function is usefull if you don't want to listen continuously to the big noise some harddrives make when spinning. (not all drives support power management!)

The 'Retry on error' controls error retry. This option is set enabled as default, and is good to leave it there.

The 'Write cache' tells the drive not to write sectors imediatly to disk, but first store in cache and then write it out in big chunks. If drive supports it, it results in increased performance on disk writes.

The 'Read look-ahead' tells the drive to read some more sectors in then it needs currently, in hope that next time these sectors will be read. This option enhances sequentia read performance significantly if drive supports it.

The 'Slow down CDROM' sets CDROM spinning speed to a slow value. This results in faster access, and more quiet operation of 50x CDROMs. Not all drives support this.

File Manager

The File Manager is a software for copying, deleting, renaming, making directories, and starting programs. Working with this program is very easy. Start it from Basic typing MAN.

There are two windows showing the directories of two different drives, so you can easily copy files from one directory to another.

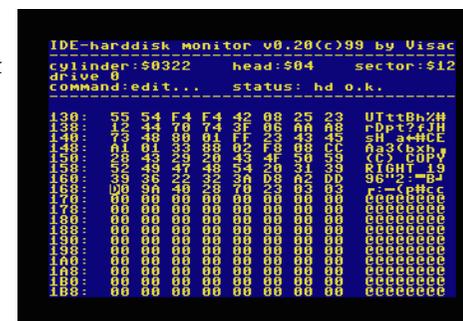


The control keys:

- CTRL - switch between windows
- C= - change device
- F1/CRSR LEFT - page up
- F2/HOME - home
- F7/CRSR RIGHT - page down
- F8/CLR - end
- CRSR UP/DOWN - move cursor
- + - select all (only files)
- - unselect all (only files)
- INS/DEL - select some files
- 1 - reread directory
- 3 - view file using "/VIEWER"
- 5 - copy (single file names may be changed)
- 6 - rename file/directories
- 7 - create directory
- 8 - delete (only files or empty directory)
- RETURN - change directory or load+start program or view file with plugin
- SHIFT + RETURN - change directory or load program
- / - go to root directory (extremely usefull with a 1581 ;-)
- ^ - open/close CDROM tray of the current panel's drive
- <- quit manager

Harddrive Monitor

The IDE harddrive monitor is a utility for reading/editing/writing sectors on HDD. It may be useful for repairing a demolished directory structure or files. (This program is available on the IDE64 homepage)



Key commands:

- I - identify drive
- R - read sector (cylinder,head and sector number)
- W - write sector (cylinder, head and sector number)
- N - follow file linking and load next sector
- + - load next sector
- - load previous sector
- E - edit sector (back with RUN/STOP)
- C - change drive (primary or secondary, if exist)
- F - format harddrive (obsolete type) -use IDEformat instead.

Machine Code Monitor

A program for reading and editing the memory and registers, and for writing and debugging simple programs in ML. Works in the external ROM, while the C64 RAM remains mostly untouched. This program pops up when pressing the 'C=' key and 'RESTORE' or when the code runs into a BRK instruction. For more detailed information, see the ide64-homepage, section "manual". Most of the commands resemble the Action Replay monitor.

Function keys:

- F1 - disk command
- F3 - move up
- F5 - move down
- F7 - directory

CHANGE

Swaps device number between harddrive and floppy drive (HDD will be device number 8, floppy drive no. 12). Use HDINIT(1) command or reset for swap back.

CHANGE1 works similarly for the second harddrive, if available. 13 <-> 8.

MAN

Starts the File Manager.

CDOPEN

Opens cdrom tray.

CDCLOSE

Closes cdrom tray.

KILL

Deactivates the cartridge, and shuts down HDDs if power management is enabled.

Use the "RESET" button for unkill.

INIT

Clears the memory, and performs a reset.

IDE Kernal Extensions

The standard kernel of the C-64 was extended with these optimized calls:

\$DEF4 - READ (read part of file to memory); precede with OPEN and CHKIN

Input: register X,Y - LOW/HI maximum number of bytes to read; register A - pointer (to zeropage) to startaddress of the buffer.

Output: register X,Y - LOW/HI number of bytes read in; \$90 - status.

This call is faster than byte by byte (CHRIN/GETIN) reading when more than 10 bytes are read but runs on IDE64 only. Please install an alternate loader aswell. If the initing fails, the carry will be set in return.

\$DEF1 - WRITE (write memory area to harddrive); precede with OPEN and CHKOUT

Input: register X,Y - LOW/HI length of data to be written in bytes; register A - pointer (to zeropage) to startaddress of buffer.

Warning: These functions can't access the ram under the IO area \$D000-\$DFFF, they will access the I/O area itself instead! To access the ram at \$A000-\$BFFF and \$E000-\$FFFF, set the processor ports \$00/\$01 correctly.

Using Manager Plugins

The File Manager can associate different applications (manager 'plugins') with different file name patterns. This feature allows you to view e.g. an Art Studio or FLI picture, listen to a SID music or desolve a D64 file with just one press on the RETURN key, for example.

The patterns are case sensitive, but you can associate two patters to one and the same plugin. (It's not a bad idea to have the patterns in uppercase, too, so you won't be suprised if you'll try to view something from an old, non-joliet CD, where everything is in uppercase ;-))

Download the Manager plugins and install the "CONFIG EXT" file (containing the file name pattern associations) into the 'root' directory of the primary harddrive.

;Example for a CONFIG EXT file

"shi:/Plugins/shi viewer"	; SuperHiresInterlace viewer
"mpic:/Plugins/mpic viewer"	; Art Studio v1.2b viewer
".mpi:/Plugins/mpic viewer"	; Art Studio v1.2b viewer
"pic:/Plugins/hpic viewer"	; Art Studio v1.1 viewer
".hpi:/Plugins/hpic viewer"	; Art Studio v1.2b viewer
".drl:/Plugins/drl viewer"	; Drazlace viewer
".drp:/Plugins/drp viewer"	; DrazPaint viewer
".fun:/Plugins/fun viewer"	; Funpaint II viewer
".ifl:/Plugins/ifl viewer"	; Gunpaint viewer
".afl:/Plugins/afl viewer"	; Afl-editor viewer
".fli:/Plugins/fli viewer"	; Flidesigner viewer
".sid:/Plugins/sid player"	; HVSID player
".d64:/Plugins/d64 writer"	; ID64 writer
".flii:/Plugins/flii viewer"	; FLI interlace viewer

For creating your own CONFIG EXT file you can use any program that can create SEQ-files. Here are the most important commands for EasyScript (downloadable on our webpage):

F1 + L: load file

F1 + F: save file ("@:filename" to replace file)

F4: disc command (\$ for directory)

PC-Link Data Transfer

PC-Link is a part of the IDE64-DOS which enables the user to access PC storage peripherals directly from the C64 the standard way using BASIC commands (LOAD, SAVE, OPEN) or using KERNAL table calls. PC-Link works also in the File Manager. You only need to solder the cable and execute a small server program on the PC. If you do not want to solder the cable yourself you can order one from Protovision (see contact page).

There are two versions of PC-Link: Serial PC-Link and Parallel PC-Link, each available for Linux and Windows. Parallel PC-Link is a direct connection between IDE64 and a PC. Serial PC-Link is faster than the parallel method. It uses the high-speed RS232-interface DUART for the transfer.

Once you have decided which kind of PC-Link you want to use, you have to reflash the corresponding firmware to your IDE64. (see "Updating your firmware").

The schematics for the parallel cable:

Pin	USER_PORT-C64	Pin	LPT_PORT-PC
A,1	GND	18..25	GND
C	PB0 out	15	ERROR
D	PB1 out	13	SELECT
E	PB2 out	12	PAPER
F	PB3 out	10	ACK IN
H	PB4 in	2	D0
J	PB5 in	3	D1
K	PB6 in	4	D2
L	PB7 in	5	D3
M	PA2 out	11	BUSY
N,12	GND	18..25	GND

The schematics of the serial cable:

Pin	DUART_PORT-C64	9 Pin	25 Pin	COM_PORT-PC
2	RxD	3	2	TxD
3	TxD	2	3	RxD
7	RTS	8	5	CTS
8	CTS	7	4	RTS
5	GND	5	7	GND

ATTENTION! Never plug your cables to any of the computers while they are turned on. A wrong soldering may also damage either computer. Use at your own risc.

IDE64 BASIC Extensions

IDE64 comes with a number of useful commands for your every-day usage. They can also be used in your BASIC programs. Most of these functions are accessible to ML programmers using channel 15 commands. See the programming section! Here is a list of the commands. Optional features are in brackets. A more detailed description of the commands (with examples) can be found on the ide64 homepage, section "manual". "dn" stands for device number.

HDINIT

Initialize the internal variables and buffers of the harddrive (command is executed automatically, after turning-on the computer). If the harddrive was turned on later than the computer, it is necessary to use this command manually. This command disables the slave device, even if it exists!

HDINIT1

Same as HDINIT, but for two devices.

CD "path"(,dn)

Change directory

If the device number is missing, the current device number is used. (This rule is valid for other commands too.) Examples: CD "/", CD "/BIN", CD "/BIN/DATA",12

LL ("path",dn)

List directory in UNIX format. The filesize is shown in bytes/Kbytes.

Description of the first 3 rows of info-characters:

row 1: D = Directory, - = file

row 2: X = eXecutable, U = User file, - = other filetype

row 3: T = proTected, - = normal, * = splat

DIR("",dn)

List directory in C64 format.

MKDIR "name"(,dn) or MKDIR "path/name"(,dn)

Creates directory in current path or in selected existing path.

RM "name"(,dn) or RM "path/name"(,dn)

Deletes (removes) file or empty directory in current or selected path.

DATE

Prints out date and time from the external time chip.