

# **Hercules Documentation Demo Documentation**

***Release 1.0.3***

**© 2023**

October 12, 2023



<b>1</b>	<b>Installation Instructions</b>	<b>1</b>
1.1	Overview . . . . .	1
1.2	Linux installation. . . . .	1
1.3	Windows installation . . . . .	1
1.4	Other platforms . . . . .	1
<b>2</b>	<b>Configuration File</b>	<b>3</b>
2.1	Example configuration file . . . . .	3
2.2	Comment lines . . . . .	5
2.3	System parameters. . . . .	5
2.4	Symbol substitutions . . . . .	22
2.5	Enhanced symbol substitutions . . . . .	23
2.6	Process and Thread Priorities . . . . .	24
2.7	Device statements. . . . .	25
<b>3</b>	<b>Operation</b>	<b>49</b>
3.1	Overview . . . . .	49
3.2	Other information . . . . .	49
<b>4</b>	<b>System Messages</b>	<b>51</b>
4.1	Overview . . . . .	51
4.2	Message Format. . . . .	51



---

## Installation Instructions

---

### 1.1 Overview

Overview section goes here.

### 1.2 Linux installation

etc etc etc etc

### 1.3 Windows installation

etc etc etc etc

### 1.4 Other platforms

etc etc etc etc



---

## Configuration File

---

This page describes the configuration file for the Hercules S/370, ESA/390, and z/Architecture emulator.

The configuration file, often named `hercules.cnf`, contains the processor and device layout. It is roughly equivalent to the IOCDS on a real System/390. The configuration file is an ASCII text file.

### 2.1 Example configuration file

**Note** Please note that the below example configuration file should not be considered a good example of how an actual configuration file should look! It is only meant to illustrate what some of the supported configuration file statements look like and how they are used.

```
#####
#                HERCULES EMULATOR CONTROL FILE                #
#                (Note: not all parameters are shown)             #
#####

#-----
#                SYSTEM PARAMETERS                               #
#-----

ARCHLVL      ESA/390
FACILITY     ENABLE 044_PFP0

PGMPRDOS     restricted
ECPSVM       no  notrap

OSTAILOR     OS/390
LOADPARM     0120....

CPUSERIAL    000611
CPUMODEL     3090
CPUVERID     FD
LPARNAME     HERCULES
LPARNUM      01
CPUIDFMT     1
MODEL        EMULATOR
PLANT        ZZ
MANUFACTURER HRC

MAINSIZE     1G
XPNDSIZE     0
NUMCPU       4
```

```
MAXCPU      8
ENGINES     CP, CP, AP, IP

SYSEPOCH    1900
YROFFSET    -28
TZOFFSET    -0500

HTTP        PORT    8081  NOAUTH
HTTP        ROOT    /usr/local/share/hercules/
HTTP        START

MODPATH     /usr/local/hercules
LDMOD       dyncrypt
NETDEV      /dev/net/tun

CCKD        NOSFD=1
SHRDPORT    3990

PANOPT      NAMEONLY RATE=FAST MSGCOLOR=DARK "TITLE=My own private MAINFRAME!"
LOGOPT      TIMESTAMP NODATESTAMP

CODEPAGE    819/1047
CNSLPORT    3270
SYSGPORT    3278
CONKPALV    (3,1,10)
LEGACYSENSEID  OFF

TIMERINT    DEFAULT
TODDRAG     1.0
DEVTMAX     8

SHCMDOPT    disable  nodiag8
DIAG8CMD    disable  noecho
CMDSEP      OFF

DEFSYM      TAPEDIR   "$(HOME)/tapes"
AUTOMOUNT   $(TAPEDIR)
AUTOMOUNT   +/tapes
AUTOMOUNT   -/tapes/vault

MOUNTED_TAPE_REINIT  allow
AUTOINIT    on
SCSIMOUNT   no

INCLUDE     mydevs.cfg
IGNORE      INCLUDE_ERRORS
INCLUDE     optdevs.cfg

#-----
#                DEVICE STATEMENTS
#                (see supported device types table)
#-----

0009        3215-C  /

000A        1442    adrdmprs.rdr
000C        3505    jcl.txt      ascii  trunc
000D        3525    pch00d.txt   ascii
000E        1403    prt00e.txt   append      cctape=legacy
001E        3211    192.168.200.1:1403 sockdev  fcb=legacy

001F        3270    * 192.168.0.1
0200.4      3270    * 192.168.0.0 255.255.255.0
```



0220.8	3270	GROUP1	192.168.100.0	255.255.255.0
0228.8	3270	GROUP2		
0230.16	3270			
0000	SYSG	SYSGCONS		
0100	3390	disks/linux.dsk	sf=shadows/linux_*.dsk ser=000000000001	
0120	3380	\${DASD_PATH=dasd/}mvsv5r.120		
0121	3380	\${DASD_PATH=dasd/}mvsv5d.121		
0122	3380	\${DASD_PATH=dasd/}mvswk1.122		
0123	3380	192.168.1.100		
0140	3370	dosres.140		
0141	3370	syswk1.141		
0300	3370	sysres.300		
0A00.3	QETH	chpid F0	iface /dev/net/tun	ipaddr 192.168.0.4 netmask 255.255.0.0
0440.2	LCS	-n	/dev/net/tun	192.168.200.2
0420.2	CTCI	192.168.200.1	192.168.200.2	
0430.2	CTCI	tun0		
#0E40	LCS	-e SNA tap0		
0E40	CTCE	31880	192.168.1.202	32880
0E41	CTCE	31882	192.168.1.202	32882
0E42.2	CTCE	1=192.168.1.202		
0460.2	PTP	192.168.200.1	192.168.200.2/24	
0470.2	PTP	tun0		
0580	3420	ickdsf.aws noautomount		
0581	3420	/cdrom/tapes/uaa196.tdf		
0582-0587	3420	\$(TAPEDIR)/volumes.\${CUU} maxsizeM=170 eotmargin=131072		
0590	3590	\\.\Tape0 # SCSI (Windows only)		
0591	3590	/dev/nst0 # SCSI (Linux or Windows)		
0592	3490	/dev/nst1 --no-erg --blkid-32 # Quantum DLT SCSI		
0020	2703	lport=32003 dial=IN lnctl=tele2 uctrans=yes term=tty skip=88C9DF		
iskip=0A				
0023	2703	lport=3780 rhost=localhost rport=3781 dial=no		
0045	2703	lport=32003 dial=IN lnctl=ibm1 term=2741 skip=5EDE code=ebcd		

## 2.2 Comment lines

Blank lines, and lines beginning with a # sign or an asterisk, are treated as comments.

## 2.3 System parameters

Except for the ARCHLVL and LPARNUM statements, system parameter statements may appear in any order but must precede any device statements. Each system parameter must be on a separate line. The following system parameters may be specified:

### 2.3.1 ARCHLVL

ARCHLVL S/370 | ESA/390 | ESAME | z/Arch

Specifies the initial architecture mode:

- use S/370 for OS/360, VM/370, and MVS 3.8.

- use `ESA/390` for `MVS/XA`, `MVS/ESA`, `OS/390`, `VM/ESA`, `VSE/ESA`, `Linux/390`, and `ZZSA`.
- use `z/Arch` or `ESAME` for `z/OS` and `zLinux`. This is the default.

When `z/Arch` or `ESAME` is specified, the machine will always IPL in `ESA/390` mode, but is capable of being switched into `z/Architecture` mode after IPL. This is handled automatically by all `z/Architecture` operating systems.

When `ARCHLVL S/370` is set, the current [Section 2.3.33](#) and [Section 2.3.11](#) settings will be automatically changed to `BASIC`. When `ARCHLVL z/Arch` is set, `LPARNUM` and `CPUIDFMT` will be reset back to 1 and 0 respectively (if needed). Refer to the “*Limited automatic LPARNUM updating when setting certain architecture modes*” section of the Release Notes document for more information.

The `ARCHLVL` statement used to be called `ARCHMODE` in previous versions of Hercules but the use of `ARCHMODE` has been deprecated in favor of the new `ARCHLVL` statement. Existing `ARCHMODE` statements should be changed to `ARCHLVL` instead. For the time being however, `ARCHMODE` is still accepted and is treated as simply a synonym for the `ARCHLVL` statement.

### 2.3.2 ASN\_AND\_LX\_REUSE

Syntax: `ASN_AND_LX_REUSE`    `ENABLE` | `DISABLE` (deprecated; use [Section 2.3.20](#))

### 2.3.3 AUTOINIT

Syntax: `AUTOINIT`    `ON` | `OFF`

The `AUTOINIT` option controls whether device files for emulated tape volumes should be automatically created or not.

When `AUTOINIT` is `ON`, a `devinit` command specifying a file that does not yet exist causes the tape driver to automatically create an empty unlabeled tape volume consisting of just two tapemarks when it discovers the specified file does not exist yet. When `AUTOINIT` is `OFF` a `devinit` command instead fails with an expected “file not found” error. For convenience the default setting is `ON`.

### 2.3.4 AUTOMOUNT

Syntax: `AUTOMOUNT`    `[±]directory`

Specifies the host system directory where the guest is allowed or not allowed to automatically load virtual tape volumes from. Prefix allowable directories with a + plus sign and unallowable directories with a - minus sign. The default prefix if neither is specified is the + plus sign (i.e. an allowable directory).

**Caution!** Enabling this feature may have security consequences depending on which allowable host system directories you specify as well as how your guest operating system enforces authorized use of the `Set Diagnose (X'4B')` channel command code.

All host system virtual tape volumes to be “automounted” by the guest must reside within one of the specified allowable host system directories or any of its subdirectories while not also being within any of the specified unallowable directories or any of their subdirectories, in order for the guest-invoked automount to be accepted.

---

**Note** Specifying a disallowed automount directory does not preclude the Hercules operator from manually mounting any desired file via the `devinit` panel command – even one in a currently defined “disallowed” automount directory. The `AUTOMOUNT` statement only controls guest-invoked automatic tape mounts and not manual tape mounts performed by the Hercules operator.

---

All directories must be specified on separate statements, but as many statements as needed may be specified in order to describe the desired allowable/unallowable directories layout. For convenience, an automount panel command is also provided to dynamically add/remove new/existing allowable/unallowable automount directories at any time.

The automount feature is activated whenever you specify at least one allowable or unallowable directory. If only unallowable directories are specified, then the current directory becomes the only defined allowable automount directory by default.

All specified directories are always resolved to fully-qualified absolute directory paths before being saved.

Refer to the description of the virtual tape device `noautomount` option for more information.

### 2.3.5 CCKD

Syntax: CCKD `cckd-parameters`

The CCKD command and initialization statement can be used to affect cckd processing. The CCKD initialization statement is specified as a Hercules configuration file statement and supports the same options as the cckd panel command. Refer to the Compressed Dasd Emulation web page for more information.

### 2.3.6 CMDSEP

Syntax: CMDSEP `OFF | c`

A command line separator character allows multiple commands to be entered on a single line. The character `c` defines the command separator character. The values `'` (period or dot), `!` (exclamation mark or bang) and `-` (dash or hyphen) are reserved and cannot be used. The default value is `'OFF'` indicating command separation is disabled.

**Warning** Choose your separator character carefully. Setting it to an alphabetic value for example disables all commands containing that character. Setting it to `e` for example will disable the `exit` command making it impossible to exit the emulator. Similarly, setting it to `o` or `f` will make it impossible to disable command separation once enabled, and setting it to `#` (hash) will prevent lines with comments from being processed correctly.

### 2.3.7 CMPSCPAD

Syntax: CMPSCPAD `alignment`

The CMPSCPAD command and initialization statement is used to define the zero padding storage alignment boundary for the CMPSC-Enhancement Facility. It must be a power of 2 value ranging anywhere from 1 to 12.

### 2.3.8 CNSLPORT

Syntax: CNSLPORT `3270 | nnnn | host:port`

Specifies (typically) the port number (in decimal) to which `tn3270` and `telnet` clients should connect. If an invalid value is specified Hercules defaults to port number 3270. See also the `SYSGPORT` statement.

The CNSLPORT statement may also be specified as `host:port`, where `host` identifies the IP address of the host interface the telnet console server should bind to (listen for connections on). If not specified the server will accept connections on the port from any host interface.

See the Telnet/tn3270 Console How-To for additional information about setting up a telnet or tn3270 client.

### 2.3.9 CODEPAGE

Syntax: CODEPAGE `mapping`

Specifies the codepage conversion mapping table used for ASCII/EBCDIC translation.

default specifies traditional Hercules codepage mapping, which is non-transparent.

Other supported predefined codepage mappings are:

Mapping	ASCII	EBCDIC	Transparent?
437/037	437 PC United States	037 United States/Canada	no
437/500	437 PC United States	500 International	no
437/1047	437 PC United States	1047 Open Systems Latin 1	no
819/037	819 ISO-8859-1	037 United States/Canada	YES
819/037v2	819 ISO-8859-1	037 United States/Canada version 2	YES
819/273	819 ISO-8859-1	273 Austria/Germany	YES
819/277	819 ISO-8859-1	277 Denmark/Norway	YES
819/278	819 ISO-8859-1	278 Finland/Sweden	YES
819/280	819 ISO-8859-1	280 Italy	YES
819/284	819 ISO-8859-1	284 Spain	YES
819/285	819 ISO-8859-1	285 United Kingdom	YES
819/297	819 ISO-8859-1	297 France	YES
819/500	819 ISO-8859-1	500 International	YES
819/1047	819 ISO-8859-1	1047 Open Systems Latin 1	YES
850/273	850 PC Latin 1	273 Austria/Germany	YES
850/1047	850 PC Latin 1	1047 Open Systems Latin 1	no
1252/037	1252 Windows Latin 1	037 United States/Canada	no
1252/037v2	1252 Windows Latin 1	037 United States/Canada version 2	no
1252/1047	1252 Windows Latin 1	1047 Open Systems Latin 1	no
1252/1140	1252 Windows Latin 1	1140 United States/Canada with Euro	YES
ISOANSI/037	ISO ANSI	037 United States/Canada	YES

The transparency column indicates whether translating from ASCII to EBCDIC (or vice versa) and back again yields results identical to the original text.

If no codepage is specified then the environment variable `HERCULES_CP` will be inspected. If the environment variable is not found then the traditional non-transparent default codepage mapping is used.

Other codepages can be defined by means of the `cp_updt` panel command (which is supported as a configuration file statement as well). Enter the panel command `help cp_updt` for more information.

The recommended code page for Linux guests is “819/500”, as it is both transparent and appears to be the code page that s390x Linux actually uses, thus allowing boot/startup messages to be parsed and displayed properly.

The recommended code page for non-Linux guests (e.g. z/OS, etc) is “819/1047”, as it is both transparent and properly translates all ASCII characters to their EBCDIC equivalents, including all extended ASCII characters too, such as:

Character	Description
!	exclamation point
[	left square bracket
]	right square bracket
{	left curly bracket
}	right curly bracket
	solid vertical bar
¡	broken vertical bar
\	backslash
¬	logical not
^	caret / circumflex
_	underscore

`	grave accent
~	tilde
¢	cent
£	pound
¤	currency sign
¥	yen
§	section sign
¶	pilcrow/paragraph
©	copyright symbol
®	registered trademark
<sup>a</sup>	feminine ordinal indicator
°	degree
<sup>o</sup>	masculine ordinal indicator
±	plus-minus
μ	mu/micro
¿	inverted question mark
×	multiplication sign
÷	obelus/divide sign

### 2.3.10 CONKPALV

Syntax: CONKPALV (idle, intv, count)

Specifies the tn3270 console and telnet clients keepalive option values that control automatic detection of disconnected tn3270/telnet client sessions.

*idle* specifies the number of seconds of inactivity until the first keepalive probe is sent (idle time until first probe, or probe frequency). *intv* specifies the interval in seconds between when successive keepalive packets are sent if no acknowledgement is received from the previous one (i.e. the timeout value of the probes themselves). *count* specifies the number of unacknowledged keepalive packets sent before the connection is considered to have failed.

The default values for Windows are 3, 1, and 10. For non-Windows systems it is 3, 1, and 9. That is, send the initial probe 3 seconds after the connection goes idle and then wait no more than one second for it to be responded to. If it is not responded to within one second, then send up to 9 more probes (for a total of 10), each of which must also timeout without being responded to before the client is considered as having died and the connection thus automatically closed.

---

**Note** This is a built-in feature of TCP/IP and allows detection of unresponsive TCP/IP connections and not idle clients. That is to say, your connection will not be terminated after 3 seconds of idle time. Your 3270 session can remain idle for many minutes or hours or days without any data being transmitted. If the TCP/IP stack at the other end of the connection – not your 3270 client itself – fails to respond to the internal keepalive probe packets however, then it means that the TCP/IP stack itself is down or there has been a physical break in the connection.

---

Thus, even if your 3270 client is completely idle, your system's TCP/IP stack itself should still respond to the keepalive probes sent by the TCP/IP stack at the Hercules end of the link. If it doesn't, then TCP/IP will terminate the tn3270/telnet session which will cause Hercules to disconnect the terminal.

The three values can also be modified on-demand via the conkpalv panel command, which has the exact same syntax. Note that the syntax is very unforgiving: no spaces are allowed anywhere within the parentheses and each value must be separated from the other with a single comma.

Please also note that not all systems support being able to modify all three values. That is, not all values may be modifiable. It is operating system dependent which values you can change and which values you cannot. On Windows for example, the count value is ignored and cannot be changed from its default value of 10. Other systems may ignore one or more or all three values and use platform

defaults instead. This is entirely system dependent. Check your system's documentation for details regarding which values can be changed and which cannot as well as how to adjust your system's default values.

### 2.3.11 CPUIDFMT

Syntax: CPUIDFMT    0 | 1 | BASIC

Specifies the format of the CPU ID the STIDP instruction should store. Refer to the [Section 2.3.33](#) statement for more information.

### 2.3.12 CPUMODEL

Syntax: CPUMODEL    0586 | xxxx | \$(symbol)

Specifies the 4 hexadecimal digit CPU machine type number (known prior to ESA/390 as the model number) stored by the STIDP instruction.

To make it easier to specify the model number for certain known models, the following symbols are now automatically predefined starting with Hercules version 4.4:

Symbol	Model
zPDT	1090
EC12	2827
BC12	2828
z13	2964
z13s	2965
z14	3906
z14ZR1	3907
z15	8561
z15T02	8562
z16	3931

Note: Hercules makes no attempt to emulate all aspects of, or features of, a given CPU model. The *CPUMODEL* statement defines a purely cosmetic value only. It defines only the value that the STIDP (Store CPU ID) instruction stores, and nothing more.

### 2.3.13 CPUSERIAL

Syntax: CPUSERIAL    000001 | xxxxxxx

Specifies the 6 hexadecimal digit CPU serial number stored by the STIDP instruction. In BASIC mode, the high-order digit may be replaced with the processor number when MAXCPU > 1; in LPAR mode, the two high-order digits are replaced with either the LPAR number or the CPU number and LPAR number with the full serial number available via the STSI instruction. The default serial number is 000001.

### 2.3.14 CPUVERID

Syntax: CPUVERID    FD | xx    [FORCE]

Specifies the 2 hexadecimal digit CPU version code stored by the STIDP instruction.

The default *cpuverid* version code at startup is FD, and that value will be stored by the STIDP instruction – even for z/Arch – unless and UNTIL you set it to a different value via the *cpuverid* command/statement.

If you try using the *cpuverid* command/statement to set a non-zero *cpuverid* value when the architecture mode is currently set to z/Arch, the version code stored by the STIDP instruction will still be stored as 00 anyway, unless ... the FORCE option is used. For z/Arch, the FORCE option is the only way to cause the *cpuverid* command to force the STIDP instruction to store a non-zero version code. (But as explained, at startup, the value stored will still be FD even for z/Arch, since that is Hercules's

default version code value. This means if you want your STIDP version code to be 00 for z/Arch, then you must use a `cpuverid` command/statement in your configuration file to set it to that value.)

### 2.3.15 DEFSYM

Syntax: `DEFSYM symbol value`

Defines symbol `symbol` as to contain value `value`. The symbol can then be the object of a substitution later in the configuration file or for panel commands. If `value` contains blanks or spaces, then it should be enclosed in double quotation marks (`"`). See substitutions for a more in-depth discussion on this feature.

Substitution is available even in configuration statements, meaning it is possible to perform substitution in the `DEFSYM` statement itself. However, symbols are always defined as the last step in the process, so attempting to self define a symbol will result in an empty string:

```
DEFSYM FOO $(FOO)
```

Will set symbol `FOO` to `""`

### 2.3.16 DEVTMAX

Syntax: `DEVTMAX -1 | 0 | nnn`

Specifies the maximum number of device threads allowed.

Specify `-1` to cause 'one time only' temporary threads to be created to service each I/O request to a device. Once the I/O request is complete, the thread exits. Subsequent I/O to the same device will cause another worker thread to be created again.

Specify `0` to cause an unlimited number of 'semi-permanent' threads to be created on an 'as-needed' basis. With this option, a thread is created to service an I/O request for a device if one doesn't already exist, but once the I/O is complete, the thread enters an idle state waiting for new work. If a new I/O request for the device arrives before the timeout period expires, the existing thread will be reused. The timeout value is currently hard coded at 5 minutes. Note that this option can cause one thread (or possibly more) to be created for each device defined in your configuration. Specifying `0` means there is no limit to the number of threads that can be created.

Specify a value from `1` to `nnn` to set an upper limit to the number of threads that can be created to service any I/O request to any device. Like the `0` option, each thread, once done servicing an I/O request, enters an idle state. If a new request arrives before the timeout period expires, the thread is reused. If all threads are busy when a new I/O request arrives however, a new thread is created only if the specified maximum has not yet been reached. If the specified maximum number of threads has already been reached, then the I/O request is placed in a queue and will be serviced by the first available thread (i.e. by whichever thread becomes idle first). This option was created to address a threading issue (possibly related to the `cygwin` Pthreads implementation) on Windows systems.

The default for Windows is `8`. The default for all other systems is `0`.

### 2.3.17 DIAG8CMD

Syntax: `DIAG8CMD DISABLE | ENABLE [ECHO | NOECHO]`

When `ENABLE` is specified the Hercules Diagnose 8 instruction command interface is enabled, allowing the guest to directly issue Hercules commands via the Diagnose 8 instruction. When set to `DISABLE` all Diagnose 8 instructions cause a Specification Exception program interrupt to occur instead.

An optional second argument can be given to request whether an audit trail of such commands should be created or not. When `ECHO` is specified, a message is issued when the command is about to be issued, when the command is redisplayed (as is normally done when entered from the command line), as well as a final message indicating the command has finished executing. When `NOECHO` is specified no such audit trail messages are displayed and the command instead completes silently (except for whatever messages the command itself may issue).

Security Alert: Enabling this feature has security consequences. When this feature is enabled it is



possible for guest operating systems running under Hercules to issue commands directly to the host operating system by means of the `sh` (host shell command) and `exec` (execute Rexx script) commands. This ability may be disabled via the `SHCMDOPT` statement's `NODIAG8` option.

The value of `ECHO` or `NOECHO` has no effect on whether or not command output will be placed into the Diagnose 8 instruction's response buffer if the instruction requested one, nor does it cause the resulting audit trail messages from being placed into the response buffer either. The `ECHO` option only impacts what is displayed on the hardware console (and what appears in the hardcopy logfile) but does not otherwise impact what is placed into the instruction's response buffer.

The default is `DISABLE NOECHO`

### 2.3.18 ECPSVM

Syntax: `ECPSVM YES | NO | LEVEL nn [ TRAP | NOTRAP ]`

Specifies whether ECPS:VM (Extended Control Program Support : Virtual Machine) support is to be enabled.

If `YES` is specified, then the support level reported to the operating system is 20. The purpose of ECPS:VM is to provide to the VM/370 Operating system a set of shortcut facilities to perform hyper-visor functions (CP Assists) and virtual machine simulation (VM Assists).

Although this feature does not affect VM Operating system products operating in XA, ESA or z/Architecture mode, it will affect VM/370 and VM/SP products running under VM/XA, VM/ESA or z/VM.

Running VM/370 and VM/SP products under VM/XA, VM/ESA or z/VM should be done with ECPS:VM disabled. ECPS:VM should not be enabled in an AP or MP environment either. ECPS:VM has no effect on non-VM operating systems. It is however recommended to disable ECPS:VM when running native non-VM operating systems.

If a specific `LEVEL` is specified, this value will be reported to the operating system when it issues a Store ECPS:VM level, but it doesn't otherwise alter the ECPS:VM facility operations.

This is a partial (but mostly complete) implementation.

It is however not a 100% complete implementation.

Please refer to the `README.ECPSVM` document for more detailed information, including an explanation of the `TRAP` and `NOTRAP` options.

### 2.3.19 ENGINES

Syntax: `ENGINES [nn*]CP|IL|AP|IP[, ...]`

Specifies the type of engine for each installed processor. The default engine type is CP.

`nn*` is an optional repeat count. Spaces are not permitted.

Examples: `ENGINES CP,CP,AP,IP` specifies that processor engines 0 and 1 are of type CP, engine 2 is type AP, and engine 3 is type IP. `ENGINES 4*CP,2*AP,2*IP` specifies that the first four processor engines (engines 0-3) are of type CP, the next two (engines 4-5) are of type AP, and the next two (engines 6-7) are of type IP.

The number of installed processor engines is determined by the `MAXCPU` statement. If the `ENGINES` statement specifies more than `MAXCPU` engines, the excess engines are ignored. If fewer than `MAXCPU` engines are specified, the remaining engines are set to type CP.

### 2.3.20 FACILITY

Syntax: `FACILITY ENABLE | DISABLE | QUERY facility [archlvl]`

Specifies a particular STFL/STFLE facility to be enabled or disabled, or a request to query of the current settings. Use `QUERY ALL` to obtain a list of valid facility names that may be used for the given `archlvl`. Enter `help facility` for detailed `FACILITY` command/statement use.

Alternatively, you can also specify the actual STFL/STFLE bit number to be turned off or on (disabled



or enabled) using the format `BITnn` where ‘nn’ corresponds to the exact STFL/STFLE facility bit you wish to be forced on or off. A popular one among the VM crowd is `ENABLE BIT44` to force the PFPO (Perform Floating-Point Operation Facility) bit on, since the facility is not enabled by default in SDL Hyperion version 4.1 or earlier. The facility is only enabled by default starting with SDL Hyperion version 4.2 or later. Specifying `ENABLE BIT44` allows z/VM guests running on SDL Hyperion 4.1 or earlier to IPL.

The optional `archlvl` argument limits the enable, disable or query function to a specific architecture. It should be noted that attempts to enable or disable a facility that a given architecture does not support are accepted without error. The default value is the value that was set by a preceding `ARCHLVL` statement or the default mode if there was no preceding `ARCHLVL` statement.

### 2.3.21 HERCPRIO

`HERCPRIO nn` (deprecated; unsupported) `TODPRIO nn` (deprecated; unsupported)

`DEVPRIO nn` (deprecated; unsupported) `CPUPRIO nn` (deprecated; unsupported)

The ability to define process and thread priorities has been removed from SDL Hyperion as of version 4.1. You should remove all such statements from your configuration file.

### 2.3.22 HTTP PORT

Syntax: `HTTP PORT nnnn [[NOAUTH] | [AUTH userid password]]`

Specifies the port number (in decimal) on which the HTTP server will listen. The port number must either be 80 or within the range 1024 - 65535 inclusive. The server is not started until a subsequent `HTTP START` statement is found.

*AUTH* indicates that a *userid* and *password* are required to access the HTTP server, whereas *NOAUTH* indicates that a *userid* and *password* are not required. The *userid* and *password* may be any valid string.

---

#### Security Alert

When *AUTH* is specified (and specifying a *userid* and *password* is thus required), one must exercise due diligence to prevent unauthorized access to Hercules’s configuration file that contains those *userids* and *passwords*.

---



---

#### Security Alert

The HTTP Server currently utilizes the insecure “http” protocol, not the more secure “https” protocol. All commands and responses are transmitted over the network in the clear, allowing anyone sniffing network traffic to see everything you do and possibly inject unauthorized commands of their own. Exercise caution when using the HTTP Server feature.

---

### 2.3.23 HTTP ROOT

Syntax: `HTTP ROOT directory`

Specifies the root directory where the HTTP server’s files reside. If not specified, the default value for Win32 builds of Hercules is the directory where the Hercules executable itself is executing out of, and for non-Win32 builds it is the directory specified as the default package installation directory when the Hercules executable was built (which can vary depending on how the Hercules package was built, but is usually `/usr/local/share/hercules/`).

### 2.3.24 HTTP START

Syntax: `HTTP START`

Starts the HTTP server. (Note: The server is no longer started by default.)

### 2.3.25 IGNORE INCLUDE\_ERRORS

Syntax: IGNORE INCLUDE\_ERRORS

Indicates that errors caused by subsequent INCLUDE statements for files which do not exist should instead be ignored rather than causing startup to be aborted (as would otherwise normally occur).

### 2.3.26 INCLUDE

Syntax: INCLUDE filepath

An INCLUDE statement tells Hercules configuration file processing to treat the contents of the file specified by filepath as if its contents had appeared in the configuration file at the point where the INCLUDE statement appears.

Note that the included file may itself contain yet another INCLUDE statement as long as the maximum nesting depth (current 8) is not exceeded.

### 2.3.27 IODELAY

Syntax: IODELAY usec

Specifies the amount of time (in microseconds) to wait after an I/O interrupt is ready to be set pending. This value can also be set using the Hercules console. The purpose of this parameter is to bypass a bug in the Linux/390 and zLinux dasd.c device driver. The problem is more apt to happen under Hercules than on a real machine because we may present an I/O interrupt sooner than a real machine.

---

**Note** OSTAILOR LINUX no longer sets IODELAY to 800 since the problem described above is no longer present in recent versions of the Linux kernel.

---

### 2.3.28 LDMOD

Syntax: LDMOD module list

Specifies additional modules that are to be loaded by the Hercules dynamic loader. The default search order is with the Hercules directory in the default DLL search path. Most systems also support absolute filenames (ie names starting with '/' or '.') in which case the default search path is not taken.

Multiple LDMOD statements may be used.

### 2.3.29 LEGACYSENSEID

Syntax: LEGACYSENSEID OFF | DISABLE | ON | ENABLE

Specifies whether the SENSE ID CCW (X'E4') will be honored for the devices that originally didn't support that feature. This includes (but may not be limited to) 3410 and 3420 tape drives, 2311 and 2314 direct access storage devices, and 2703 communication controllers.

Specify ON or ENABLE if your guest operating system needs the Sense ID support to dynamically detect those devices. Note that most current operating systems will not detect those devices even though Sense ID is enabled because those devices never supported the Sense ID in the first place. So this mainly applies to custom built or modified versions of guest operating systems that are aware of this specific Hercules capability.

Because those legacy devices didn't originally support this command, and for compatibility reasons, the default is OFF or DISABLE.

### 2.3.30 LOADPARM

Syntax: LOADPARM xxxxxxxx

Specifies the default eight-character IPL parameter used by some operating systems to select system

parameters.

The specified value is used as the IPL command's default LOADPARM option parameter when the IPL command is issued without the LOADPARM option. Regardless of the value specified for this setting, the value can always be overridden by specifying a different value for the LOADPARM option on the IPL command itself. The LOADPARM configuration file option simply specifies a default value that will be used unless overridden with a different value on the IPL command itself.

### 2.3.31 LOGOPT

Syntax: LOGOPT    TIMESTAMP | NOTIMESTAMP | DATESTAMP | NODATESTAMP

Sets logfile options. **TIMESTAMP** inserts a time stamp in front of each log message. **NOTIMESTAMP** logs messages without time stamps. Similarly, **DATESTAMP** and **NODATESTAMP** prefixes logfile messages with or without the current date. The current resolution of the stamp is one second.

The default is *TIMESTAMP NODATESTAMP*.

### 2.3.32 LPARNAME

Syntax: LPARNAME    name

Specifies the LPAR name returned by DIAG X'204'. The default is *HERCULES*.

### 2.3.33 LPARNUM

Syntax: LPARNUM    BASIC | 1 | xx

Specifies the one- or two-digit hexadecimal LPAR identification number stored by the STIDP instruction, or BASIC.

If a one-digit number from 1 to F (hexadecimal) is specified, then STIDP stores a format-0 CPU ID unless a subsequent CPUIDFMT 1 statement is specified.

If 0 or a two-digit hexadecimal number (except 10 hexadecimal) is specified, then STIDP stores a format-1 CPU ID. For LPARNUM 10 the current CPUIDFMT is not changed.

When LPARNUM is BASIC, then STIDP stores a basic-mode CPU ID (6-hexadecimal digit serial number when MAXCPU = 1, or a one-hexadecimal digit CPU number and 5-hexadecimal digit serial number when MAXCPU > 1).

The LPARNUM setting will be automatically changed if needed to BASIC when ARCHLVL S/370 is set (which also changes the CPUIDFMT setting to BASIC too), and is automatically set to LPARNUM 1 (and CPUIDFMT 0) if needed when ARCHLVL z/Arch is set. Refer to the "Limited automatic LPARNUM updating when setting certain architecture modes" section of the Release Notes document for more information.

The default is LPARNUM 1 with a format-0 CPU ID.

### 2.3.34 MAINSIZE

Syntax: MAINSIZE    nnnn | nnnK | nnnM | nnnG | nnnT | nnnP | nnnE

Specifies the main storage size in megabytes, where nnnn is a decimal number. Or, nnnM where M is K - Kilobytes, M - Megabytes, G - Gigabytes, T - Terabytes, P - Petabytes, E - Exabytes. The default on startup is 2M.

For storage sizes less than 16M, sizes not on a 4K boundary are rounded up to the next 4K boundary. Otherwise, storage sizes not on a 1M boundary are rounded up to the next 1M boundary.

The minimum size is 4K for S/370 and ESA/390, and 8K for z/Arch. A maximum of 64M may be specified for S/370, 2048M (2G) for ESA/390, and 16E for z/Arch.

Notes:

The actual upper limit is determined by your host system's architecture and operating system, the guest operating system, and the amount of physical memory and available paging space. The total of mainsize and xpndsize on host systems with a 32-bit architecture will be limited to less than 4G;

host systems with a 64-bit architecture will be limited to less than 16E. .. caution:: Using minimum storage sizes, storage sizes less than 64K or a size that is not a multiple of 64K for S/370, or a size less than 1M or is not a multiple of 1M for z/Arch is not recommended as it could generate error conditions which are not covered by the Principles of Operations. Use of storage sizes greater than supported by the guest operating system may generate incorrect results or error conditions within the guest operating system.

### 2.3.35 MANUFACTURER

Syntax: MANUFACTURER    name

Specifies the 1-16 character MANUFACTURER name returned the STSI instruction. Valid characters are 0-9 and uppercase A-Z only. The default is *HRC*.

### 2.3.36 MAXCPU

Syntax: MAXCPU    nn

Specifies the number of installed processor engines. The NUMCPU statement specifies the number of engines which will be configured online at startup time. All processors are CP engines unless otherwise specified by the ENGINES statement.

The value of MAXCPU cannot exceed the value of MAX\_CPU\_ENGS. If MAXCPU is not specified in the Hercules configuration file, then its initial value is equal to NUMCPU. If MAXCPU and NUMCPU are both omitted, MAXCPU is set to 1.

MAX\_CPU\_ENGS is a compile-time variable which sets an upper limit on the value of MAXCPU. The value of MAX\_CPU\_ENGS is displayed in the Build information message on the Hercules control panel at startup time. To change the value of MAX\_CPU\_ENGS you must rebuild Hercules. For Unix builds, specify `./configure --enable-multi-cpu=nn` before performing make. For Windows builds, specify `SET MAX_CPU_ENGS=nn` before performing nmake.

MAX\_CPU\_ENGS cannot exceed 64. For performance reasons, values above 32 are not recommended for 32-bit platforms. If MAX\_CPU\_ENGS is set to 1 then multiprocessing is disabled. See also NUMCPU for a discussion of the performance implications of MAX\_CPU\_ENGS.

### 2.3.37 MODEL

Syntax: MODEL    hardware\_model [ capacity\_model ] [ perm\_capacity\_model ] [ temp\_capacity\_model ]

Specifies the 1-16 character MODEL names returned the STSI instruction. Valid characters are 0-9 and uppercase A-Z only. The default is *EMULATOR*.

If two operands are supplied, the first is the hardware model name (CPC ND model) and the second is the capacity model name (CPC SI model). If only one operand is supplied, it is used as both the hardware model name and the capacity model name. The optional third and fourth operands specify the permanent capacity model name and the temporary capacity model name respectively.

### 2.3.38 MODPATH

Syntax: MODPATH    path

Specifies the relative or absolute path of the directory where dynamic modules should be loaded from. Only one directory may be specified. The path should be enclosed within double quotes if it contains any blanks.

When a modpath statement is specified, the path on the modpath statement is searched before the default path is searched. The system default varies depending on the host platform where Hercules is being run.

### 2.3.39 MOUNTED\_TAPE\_REINIT

Syntax: MOUNTED\_TAPE\_REINIT    DISALLOW | DISABLE | ALLOW | ENABLE

Specifies whether reinitialization of tape drive devices (via the `devinit` command, in order to mount a new tape) should be allowed if there is already a tape mounted on the drive.

Specifying `ALLOW|ENABLE` (default) indicates new tapes may be mounted (via `'devinit nnnn new-tape-filename'`) irrespective of whether or not there is already a tape mounted on the drive.

Specifying `DISALLOW|DISABLE` prevents new tapes from being mounted if one is already mounted. When `DISALLOW` is specified and a tape is already mounted on the drive, it must first be unmounted (via the command `'devinit nnnn *'`) before the new tape can be mounted. Otherwise the `devinit` attempt to mount the new tape is rejected.

This option is meant as a safety mechanism to protect against accidentally dismounting a tape from the wrong drive as a result of a simple typo (thereby cancelling a potentially important tape job) and was added by user request.

Also note that for SCSI tape drives the `'devinit nnnn *'` command has no effect as the tape must be unmounted manually (since it is a real physical device and not one emulated via a disk file like `.AWS` tapes).

### 2.3.40 NETDEV

Syntax: `NETDEV devname`

Specifies the name (or for Windows, the IP or MAC address) of the underlying default host network adapter to be used for Hercules communications devices unless overridden on the device statement itself.

The default for Linux (except Apple and FreeBSD) is `/dev/net/tun`. The default for Apple and FreeBSD is `/dev/tun`.

The default for Windows is whichever host adapter that SoftDevLab's CTCI-WIN product returns as its "default host network adapter", which for older versions of CTCI-WIN (3.5.0 and earlier) is the first network adapter in the Windows host's binding order (which may not be desirable for some users). The `NETDEV` statement allows you to override this.

Refer to the 'Help' file included in newer versions of CTCI-WIN (version 3.6.0 or greater) for information regarding modifying Windows's "Adapter Binding Order" and/or defining a preferred CTCI-WIN default host network adapter.

---

**Note** Hercules's networking support requires privileged access to your host's networking devices. If Hercules is not started with Administrative (root) privileges then initialization of your networking devices will fail and your guest's networking will not work.

---

### 2.3.41 NUMCPU

Syntax: `NUMCPU nn`

Specifies the number of emulated processor engines which will be configured online at startup time. `NUMCPU` cannot exceed the value of `MAXCPU`. If `NUMCPU` is less than `MAXCPU` then the remaining engines can be configured online later. The default `NUMCPU` value is 1. The minimum value is 0.

Multiprocessor emulation works best if your host system actually has more than one physical CPU, but you can still emulate multiple CPUs nevertheless even on a uniprocessor system (and you might even achieve a small performance benefit when you do). There is little point, however, in specifying `NUMCPU` greater than 1 unless your guest operating system (running under Hercules) is actually able to support multiple CPUs (and if you do not actually need multiprocessor emulation, then setting `MAX_CPU_ENGS` to 1 at compile time might even produce a slight performance advantage too).

### 2.3.42 OSTAILOR

Syntax: `OSTAILOR DEFAULT | OS/390 | z/OS | VM | z/VM | VSE | z/VSE | LINUX`

| OpenSolaris | QUIET | NULL

Specifies the intended operating system. The effect of this parameter is to reduce control panel message traffic by selectively suppressing trace messages for program checks which are considered normal in the specified environment. QUIET suppresses all exception messages. NULL displays all exception messages.

---

**Note** Neither QUIET nor NULL should ever be used in normal circumstances! QUIET hides what otherwise might be important messages needed to diagnose incorrect Hercules and/or guest functionality. Only use QUIET under the guidance of Hercules product support personnel. Instead, please specify an OSTAILOR value that is appropriate for the guest operating system you intend to run under Hercules, or else let it default by not specifying it at all.

---

If no OSTAILOR statement is specified at all, then DEFAULT is used, which suppresses only program check messages for interruption codes 10 (segment-translation exception), 11 (page-translation exception), 16 (trace-table exception) and 1C (space-switch event), which are considered to be completely normal and unremarkable for virtually all known mainframe operating systems.

Optionally prefix any value (except QUIET or NULL) with '+' to cause the suppressions for that environment to be combined (added) to those already specified, or with '-' to remove such suppressions (i.e. to allow them).

This allows you to, for example, suppress messages for both z/VM as well as for z/OS too (for situations where you intend to run z/OS as a guest under z/VM) by specifying two OSTAILOR statements as follows:

OSTAILOR z/VM OSTAILOR +Z/OS Use the "pgmtrace" panel command to fine tune the current settings.

### 2.3.43 PANOPT

Syntax: PANOPT      FULLPATH|NAMEONLY   RATE=SLOW|FAST|nnn   MSGCOLOR=NO|DARK|LIGHT  
TITLE=xxx

Defines panel display options. NAMEONLY requests the extended panel screen (that displays the list of devices and is reached by pressing the ESC key) to display only the emulated device's base filename. The default is FULLPATH which displays the file's full path filename.

RATE=nnn specifies the panel refresh rate in milliseconds between screen refreshes. RATE=SLOW is the same as 500. RATE=FAST is the same as 50. A value less than the system clock tick interval or greater than 5000 will be rejected. SLOW is the default.

MSGCOLOR=DARK displays colorized panel messages meant for 'dark' panels (such as white text on black background) whereas MSGCOLOR=LIGHT is obviously meant for panels using dark text on light backgrounds. Only 'E' (error), 'W' (warning), 'D' (debug) and 'I' (informational) messages are colorized. Any message not detected as a Hercules "HHCnnnnnX" format message are not colorized. The colors are currently hard coded and cannot be changed. NO is the default, but DARK or LIGHT (as appropriate for your system) is strongly encouraged as it makes error and warning messages more noticeable and less likely to be missed.

TITLE=xxx defines an optional console window title-bar string to be used in place of the default supplied by the windowing system, and allows you to distinguish between different Hercules sessions (instances) running on the same machine. If the value contains any blanks, the entire option specification should be enclosed within double-quotes (e.g. "TITLE=my title", not TITLE="my title" which is an error).

The TITLE= option takes effect only when the Hercules console is displayed on either an xterm terminal (commonly used on Unix systems) or in a Command Prompt window on Windows systems.

---

**Note** Neither the MSGCOLOR= nor the TITLE= option has any effect when Hercules is run under the control of an external GUI since since Hercules's console window is hidden in favor of using the external GUI's window instead (and the GUI is in control of its own colors). The RATE= option however, controls how often the external GUI will refresh its own window and other user-interface

---



controls. Similarly, the FULLPATH and NAMEONLY option controls how the external GUI displays your list of emulated devices.

---

### 2.3.44 PANRATE

Syntax: PANRATE SLOW | FAST | nn (deprecated; use PANOPT instead)

This statement has been deprecated in favor of the PANOPT statement instead.

### 2.3.45 PANTITLE

Syntax: PANTITLE "title-string" (deprecated; use PANOPT instead)

This statement has been deprecated in favor of the PANOPT statement instead.

### 2.3.46 PGMPRDOS

Syntax: PGMPRDOS RESTRICTED | LICENSED

Specifies whether or not Hercules will run licensed program product ESA or z/Architecture operating systems. If RESTRICTED is specified, Hercules will stop all CPUs when a licensed program product operating system is detected. Specify LICENSED to allow these operating systems to run normally. This parameter has no effect on Linux/390, Linux for z/Series, or any 370-mode OS.

---

**Note** It is YOUR responsibility to comply with the terms of the license for the operating system you intend to run on Hercules. If you specify LICENSED and run a licensed operating system in violation of that license, then don't come after the Hercules developers when the vendor sends their lawyers after you!

---

RESTRICTED is the default. Specifying LICENSED will produce a message when a licensed operating system is detected to remind you of your responsibility to comply with software license terms.

### 2.3.47 PLANT

Syntax: PLANT name

Specifies the 1-4 character PLANT name returned the STSI instruction. Valid characters are 0-9 and uppercase A-Z only. The default is ZZ.

### 2.3.48 SCSIMOUNT

Syntax: SCSIMOUNT NO | YES | nn

Specifies whether automatic detection of SCSI tape mounts are to be enabled or not.

Specifying NO or 0 seconds (the default) indicates the option is disabled, forcing all SCSI tape mounts to be done manually via an appropriate devinit command.

A value from 1 to 99 seconds inclusive enables the option and causes periodic queries of the SCSI tape drive to automatically detect when a new tape is mounted. Specifying YES is the same as specifying 5 seconds.

The scsimount panel command may also be used to display and/or modify this value on demand once Hercules has been started. Note too that the scsimount panel command also lists any mounts and/or dismounts that may still be pending on the drive, as long as you've defined your tape drive as a model that has an LCD "display" (such as a model 3480, 3490 or 3590).

---

**Note** Enabling this option may cause Hercules to take longer to shutdown depending on the value specified for this option as well as how the host operating system (Windows, Linux, etc) and associated hardware (SCSI adapter) behaves to drive status queries for drives which do not have any media

---

currently mounted on them.

---

### 2.3.49 SHCMDOPT

Syntax: SHCMDOPT    DISABLE | ENABLE    [DIAG8 | NODIAG8]

When set to DISABLE, the sh (host shell command) and exec (Rexx execute script) commands are globally disabled and will result in an error if entered either directly via the hardware console or programmatically via the DIAG8CMD interface.

If the optional NODIAG8 option is specified, then only the programmatic execution of commands via the the Diagnose 8 interface are disabled, but shell and Rexx commands entered directly via the Hercules command line still work. This includes commands entered via the HTTP server facility as well as commands issued by .rc “run command” scripts too (automatically at startup or directly or indirectly via the script command).

Security Alert: Enabling this feature has security consequences. When ENABLE DIAG8 is specified it is possible for guest operating systems running under Hercules to issue commands directly to the host operating system.

### 2.3.50 SHRDPORT

Syntax: SHRDPORT    nnnn

Specifies the port number (in decimal) on which the Shared Device server will listen. Specifying SHRDPORT will allow other Hercules instances to access devices on this instance. (Currently only DASD devices may be shared). By default, the other Hercules instances (clients) will use port 3990. If you specify a different port number, then you will have to specify this port number on the device statement for the other Hercules clients. If no SHRDPORT statement is present then the Shared Device server thread will not be activated.

### 2.3.51 SYSEPOCH

Syntax: SYSEPOCH    yyyy [±years]

Specifies the base date for the TOD clock. Use the default value (1900) for all systems except OS/360. Use 1960 for OS/360. Values other than these were formerly used to offset the TOD clock by a number of years to move the date before the year 2000 for non-Y2K-compliant operating systems. This use is deprecated, and support will be removed in a future release; at that time, only values of 1900 or 1960 will be accepted. Other values will produce a warning message with the equivalent values to specify in the SYSEPOCH statement. An optional year offset may be specified, and will be treated as though it had been specified on a YROFFSET statement.

### 2.3.52 SYSGPORT

Syntax: SYSGPORT    NO   -or-   3278   -or-   nnnn   -or-   host:port

Specifies (typically) the port number (in decimal) to which tn3270 and telnet clients should use to connect to a SYSG console device or the value NO. The default is NO, meaning no separate listening socket will be created for SYSG console connections. If an invalid value is specified Hercules defaults to port number 3278. See also the CNLSPORT statement.

The SYSGPORT statement may also be specified as host:port, where host identifies the IP address of the host interface the telnet console server should bind to (listen for connections on). If not specified the server will accept connections on the port from any host interface.

See the Telnet/tn3270 Console How-To for additional information about setting up a telnet or tn3270 client.

### 2.3.53 TIMERINT

Syntax: TIMERINT    DEFAULT | nnnn

Specifies the internal timers update interval, in microseconds. This parameter specifies how



frequently Hercules's internal timers-update thread updates the TOD Clock, CPU Timer, and other architectural related clock/timer values.

When the z/Architecture Transactional-Execution Facility (073\_TRANSACT\_EXEC) is not installed or enabled, the minimum and default intervals are 1 and 50 microseconds respectively, which strikes a reasonable balance between clock accuracy and overall host performance.

When the z/Architecture Transactional-Execution Facility is installed and enabled, the minimum and default intervals are 200 and 400 microseconds.

The maximum allowed interval is 999999 microseconds (one microsecond less than one second).

Also note that due to host system limitations and/or design, some hosts may round up and/or coalesce short microsecond intervals to a much longer millisecond interval instead.

**Caution!** While lower TIMERINT values may help increase the accuracy of your guest's TOD Clock and CPU Timer values, it could also have a severe negative impact on host operating system performance as well. You should exercise extreme caution when choosing your TIMERINT value in relationship to the actual process priority (nice value) of the Hercules process itself.

### 2.3.54 TODDRAG

Syntax: TODDRAG    n.nn

Specifies the TOD clock drag factor. This parameter can be used to slow down or speed up the TOD clock by a factor of nn. A significant slowdown can improve the performance of some operating systems which consume significant amounts of CPU time processing timer interrupts. A drag factor of 2.0 slows down the clock by 50%. A drag factor of 0.5 doubles the speed of the clock. A drag factor of 1.01 slows down the clock by 1%, and 0.99 speeds up the clock by 1%.

### 2.3.55 TRACEOPT

Syntax: TRACEOPT    [TRADITIONAL | REGSFIRST | NOREGS]    [NOCH9OFLOW]

Sets the Hercules instruction and device tracing option(s).

TRADITIONAL (the default), displays the instruction about to be executed followed by the current register values such that pressing the ENTER key (to execute the displayed instruction) then shows the next instruction to be executed followed by a display of the updated registers.

REGSFIRST displays the current register values first, followed by the instruction about to be executed, such that pressing the ENTER key (to execute the displayed instruction) then shows the newly updated register values (that the instruction just updated) followed by the next instruction about to be executed.

NOREGS suppresses the registers display altogether and shows just the instruction about to be executed.

NOCH9OFLOW suppresses CCW tracing of printer channel-9 overflow unit checks which are considered completely normal and not true device errors. The unit checks still occur, they are simply not traced unless CCW tracing is explicitly enabled on the device.

### 2.3.56 TZOFFSET

Syntax: TZOFFSET    ±hhmm

Specifies the hours and minutes by which the TOD clock will be offset from the current system time. For GMT, use the default value (0000). For timezones west of Greenwich, specify a negative value (example: -0500 for US Eastern Standard Time, -0800 for US Pacific Standard Time). For timezones east of Greenwich, specify a positive value (example: +0100 for Central European Time, +0930 for South Australian Time).

### 2.3.57 XPNDSIZE

Syntax: XPNDSIZE    nnnn | nnnM | nnnG | nnnT | nnnP | nnnE

Specifies the expanded storage size in megabytes, where nnnn is a decimal number. Or, nnnM where M is K - Kilobytes, M - Megabytes, G - Gigabytes, T - Terabytes, P - Petabytes, E - Exabytes.

Storage sizes not on a 1M boundary are rounded up to the next 1M boundary. The lower limit and default is 0.

Notes:

The actual upper limit is determined by your host system's architecture and operating system, the guest operating system, and the amount of physical memory and available paging space. The total of mainsize and xpndsize on host systems with a 32-bit architecture will be limited to less than 4G; host systems with a 64-bit architecture will be limited to less than 16E. Use of storage sizes greater than supported by the guest operating system may generate incorrect results or error conditions within the guest operating system.

### 2.3.58 YROFFSET

Syntax: YROFFSET    ±years

Specifies a number of years to offset the TOD clock from the actual date. Positive numbers will move the clock forward in time, while negative numbers will move it backward. A common value for non-Y2K-compliant operating systems is YROFFSET -28, which has the advantage that the day of the week and the presence or absence of February 29 is the same as the current year. This value may not be specified as greater than ±142 years, the total range of the TOD clock. Specifying a value that causes the computed TOD clock year to be earlier than the value of SYSEPOCH or more than 142 years later than that value will produce unexpected results.

A comment preceded by a # sign may be appended to any system parameter statement.

## 2.4 Symbol substitutions

In configuration and device statements, as well as in panel commands and OAT files, symbols may be substituted for text.

### 2.4.1 Syntax

To substitute symbol symbol with its contents, the symbol should be enclosed within parenthesis and preceded by a \$ sign. For example, if symbol FOO contains the text string "BAR" then \$(FOO) will be substituted with the string "BAR";. Symbol names are case sensitive.

#### Examples

```
DEFSYM  TAPEDIR  "/home/hercules/tapes"
0380    3420    $(TAPEDIR)/scratch.aws
...
```

In this example, device 0380 will be a 3420 loaded with the AWS tape file in /home/hercules/-tapes/scratch.aws

### 2.4.2 Special symbols

#### Device group symbols

When multiple devices are defined with a single device definition statement, then the symbols

CUU (3 digits device number, upper case hexadecimal digits) CCUU (4 digits device number, upper case hexadecimal digits) DEVN (4 digits device number, upper case hexadecimal digits)

are defined to contain for each device the relevant device address. For example:

```
0200, 0201 3340 /home/hercules/dasds/myvols.$(CUU)
```

will define two 3340 packs, with device 0200 being loaded with the file myvols.200 and device 0201 defined with myvols.201.

### Environment variables

If a symbol is not explicitly defined by a DEFSYM statement and an environment variable by the same name exists, the string contents of that environment variable will be used for substitution.

### Undefined symbols

If a symbol is not defined by an explicit DEFSYM, is not an automatically generated symbol and is not an environment variable, an empty string will be substituted.

## 2.4.3 Escaping substitution, recursion

To be able to specify the `$ (` string without incurring substitution, an additional `$` sign should be used. For example, `$$ (FOO)` will not be substituted. If substitution is required but the preceding text is to contain a `$` sign as the very last character, then `$$$ (FOO)` should be specified. Thus, if symbol `FOO` contains `BAR`, then `$$ (FOO)` will remain `$$ (FOO)` while `$$$ (FOO)` will become `"$BAR"`.

Substitution is not recursive (only one substitution pass is made).

## 2.5 Enhanced symbol substitutions

Enhanced symbol substitution differs from the above normal symbol substitution in several very important ways:

First, the syntax is different. Enhanced substitution symbol names are specified using `$(var)` (dollar + brace) rather than `$(var)` (dollar + parenthesis).

Second, the enhanced syntax supports specifying a default value that is to be used instead whenever the name symbol is otherwise not defined. The default value is placed within the opening and closing braces just as the symbol name is, but separated from it by either a single equal sign `=` or a colon-equal-sign `:=`.

For example, specifying `$(DASD_PATH=dasd/)` in your configuration file requests that the value of the `DASD_PATH` symbol or environment variable be substituted, or, if the variable is undefined, to use the value `dasd/` instead. If no default value is specified then an empty string is used instead.

Finally, enhanced symbol substitution occurs only from host defined environment variables and not from any identically named DEFSYM symbol should one exist. For example, if environment variable `'FOO'` is defined with the value `"bar"`, then the configuration file statement `DEFSYM FOO myfoo` followed immediately by the statement `$(FOO)` causes the value `bar` to be substituted and not `myfoo` as might otherwise be believed, whereas the statement `$(FOO)`, since it is a normal symbol substitution sequence does get replaced with `myfoo` (since that was the value defined to it via the preceding DEFSYM statement).

In other words each symbol substitution technique is supported completely separately from one another. DEFSYM allows one to define/undefine/use private (internally defined) symbols separate from the host operating system's environment variable pool, whereas the enhanced symbol substitution does not and instead only allows read-only access to the host's environment variable pool with no support for modifying an already defined symbol (environment variable) but a nonetheless convenient means of defining a default value to be used should the specified host environment variable be currently undefined.

Further note that symbol names, being the names of environment variables, are subject to whatever case sensitivity or case insensitivity that the host operating system happens to enforce/allow. On Windows, environment variables are not case sensitive, whereas on other operating systems they may be. Thus `"${FOO}"`, `"${foo}"`, `"${Foo}"`, etc, all cause the same value to be substituted on Windows,

whereas the DEFSYM symbols `$(FOO)` and `$(foo)`, being two completely different and unique symbols, could be substituted with two completely different values (since DEFSYM is case sensitive across all supported platforms, including Windows).

### 2.5.1 Syntax

To substitute symbol symbol with the current environment variable value, the symbol should be enclosed within braces and preceded by a `$` sign. For example, if an environment variable named `FOO` holds the value `"BAR"`, then `${FOO}` will be substituted with the string `"BAR"`. If the environment variable `"FOO"` is not defined then a null (empty) string is substituted instead.

If the string `${FOO:=myfoo}` is used instead, then the value `BAR` will still be substituted if the value `BAR` was indeed previously assigned to `FOO`, but will be substituted with the value `myfoo` instead if the environment variable `FOO` is currently undefined.

Note too that the default value is a literal string and no substitution is applied to it. Thus attempting to use the syntax `"${foo=${bar}}"` will not yield the expected results. It will not be substituted with the currently defined value of the `"bar"` environment variable, but rather will always be substituted with the literal string `"${bar}"` followed immediately by the literal character `'`.

Symbol names (environment variable names) are not case sensitive on Windows whereas they might be on other host operating systems.

## 2.6 Process and Thread Priorities

### 2.6.1 Process Priorities

For Windows, the following conversions are used for translating Unix process `'nice'` values to Windows process priority classes:

'Nice' Value	Windows Process Priority Class	Foo Meaning
-20 to -16	Real-time	Process that has the highest possible priority. The threads of the process preempt the threads of all other processes, including operating system processes performing important tasks. For example, a real-time process that executes for more than a very brief interval can cause disk caches not to flush or cause the mouse to be unresponsive.
-15 to -9	High	Process that performs time-critical tasks that must be executed immediately. The threads of the process preempt the threads of normal or idle priority class processes. An example is the Task List, which must respond quickly when called by the user, regardless of the load on the operating system. Use extreme care when using the high-priority class, because a high-priority class application can use nearly all available CPU time.
-8 to -1	Above Normal	Process that has priority above the Normal class but below the High class.

**Caution!** A high process priority (or low `'nice'` value) could have a impact on how Hercules's internal thread priorities are interpreted, thereby impacting the overall performance of your host system.

### 2.6.2 Thread Priorities

The following are the currently assigned internal relative thread priorities for Hercules:

Relative priority Thread Type Description

- 1            (watchdog)    The “watch dog” thread is the internal Hercules thread that monitors all CPU threads for a malfunctioning CPU (i.e. one that, due to a bug, has stopped executing instructions). The watch dog thread is always set to a relative priority one less than the priority assigned to the CPU threads.
- 2            CPU        The CPU threads are those within Hercules that actually execute the emulated System/370, ESA/390, or z/Architecture instructions. There is one CPU thread for each defined ENGINE. Except for the watch dog thread, CPU threads are always assigned the lowest relative priority of any thread within Hercules. This allows, for example, I/O requests to be scheduled and completed in favour of CPU cycles being burned.
- 3            Device    Device threads within Hercules manage the I/O requests to emulated devices. Assigning the internal relative priority of device threads to be higher than that of the CPU threads ensures no compute-bound CPU thread impacts Hercules’s ability to start and complete its I/O requests to its emulated devices.
- 4            Server    Server threads are those threads that provide a service to either the user or to other threads internal to Hercules. Threads that monitor for incoming emulated display terminal connection requests for example (or the internal logger thread that manages the passing of messages between threads) are two examples of server threads. Such threads must react to connection requests and/or internal messages quickly since other threads are waiting for them to complete their task before they themselves can proceed.
- 5            Panel     The main Hercules thread is the one that reads commands from the keyboard and displays messages on the screen. Except for the TOD Clock and Timer thread, it should normally be the highest priority of any thread within Hercules.
- 6            (unused)    —
- 7            Timer     TOD Clock and Timer thread is the thread which manages the internal emulated TOD Clock and CPU Timer components of your emulated mainframe. In order to ensure accurate time of day and elapsed time and/or CPU time measurement, it should always be the very highest priority thread within Hercules. .. caution:: Hercules’s internal thread priorities could be interpreted differently based on its process priority (or ‘nice’ value), thereby impacting the overall performance of your host system.

## 2.7 Device statements

The remaining statements in the configuration file are device statements. There must be one device statement for each I/O device or group of identical I/O devices. The format of the device statement is:

devnum(s) devtype [ arguments ] [ # comments... ] where the generic syntax for device numbers is [n:]CCUU[CCUU][CCUU][.nn][...] as explained below:

devnum(s) is either a single devnum, a range of devnums (separated by a ‘-’ (dash)), a count of devnums (separated by a ‘.’ (dot/period/stop)), or a comma separated list of devnums. Examples would be 200-210 or 0300.10 or 0400,0403 or 0100,0110-011F.

All devices defined when devnums specifies more than one device have identical characteristics (except for the device number itself). All devices defined as a group must be defined on a single channel. A channel is defined as a contiguous group of 256 (or hexadecimal 100) devices. 0010 and 0020 are on the same channels. 0100 and 0210 are not.

See devnum immediately below for an explanation of how each device number is specified.

The 4 special substitution symbols CUU, CCUU, cuu and ccuu are also defined for each device in a device group. See substitutions for details.

devnum is either a 1 to 4 digit hexadecimal number in the range 0000 to FFFF for ESA/390, or 0000 to 0FFF for S/370. The device number uniquely identifies each device to the operating system.

Channel Set / Logical Channel Subsystem An optional Channel Set or Logical Channel Subsystem Identification can be specified for a device number or group of devices. The Identification number is

specified at the beginning of the definition, followed by a ':' character. For example :

1:0400-040F 3270

defines 3270 devices 400 to 40F to be on S/370 Channel Set 1 or on S/390 or z/Architecture Logical Channel Subsystem # 1.

Since each Logical Channel Subsystem defines its own device numbering space, care should be taken in S/370 mode as to define a coherent set of device numbers.

Not all S/390 or z/Architecture operating systems support Multiple Logical Channel Subsystems (this feature was introduced with the z9-109).

If no Channel Set or Logical Channel Subsystem Identification is specified, then it is assumed to be 0.

devtype is the device type. Valid device types are shown in the table just below.

arguments is a list of parameters whose meaning depends on the device type. The arguments required for each class of device are shown further below.

# comments... A comment preceded by a # sign may be appended to any device definition statement.

Device type	Description	Emulated by
3270, 3287	Local non-SNA 3270 display or printer	TN3270 client connection
SYSG	Integrated 3270 console	TN3270 client connection
1052, 3215	Console printer-keyboards	Telnet client connection
1052-C, 3215-C	Integrated console printer-keyboards	Integrated on Hercules console
1442, 2501, 3505	Card readers	Disk file(s) (ASCII or EBCDIC)
3525	Card punch	Disk file (ASCII or EBCDIC)
1403, 3203, 3211	Line printers	Disk file (ASCII)
3410, 3420, 3422, 3430, 3480, 3490, 3590, 9347, 8809	Tape drives	Disk file, CDROM, or SCSI tape
many	Communication and Channel-to-Channel devices	many
(( OSA ))	OSA Express adapter operating on QDIO mode. Both layer-2 and layer-3 modes are supported.	QETH (OSA/QDIO Ethernet Adapter) Tun/Tap Driver
(( LCS ))	IBM 8232 LCS device, LCS3172 driver of a P/390, IBM 2216 router, IBM 3172 running ICP.	LCS (LAN Channel Station) Tun/Tap Driver
(( CTCI ))	Channel-to-Channel link to host TCP/IP stack	CTCI Tun/Tap Driver
(( CTCE ))	Enhanced Channel-to-Channel Emulation via TCP connection (true 3088 CTCA)	CTCE driver
(( PTP ))	MPCPTP/MPCPTP6 Channel to Channel link	PTP Tun/Tap Driver
3310, 3370, 9332, 9335, 9336, 0671	FBA direct access storage devices	Disk file
2305, 2311, 2314, 3330, 3340, 3350, 3375, 3380, 3390, 9345	CKD direct access storage devices	Disk file
2703	Communication Line, Remote Teletype, etc.	TCP Socket

Device type	Description	Emulated by
3270, 3287	Local non-SNA 3270 display or printer	TN3270 client connection
SYSG	Integrated 3270 console	TN3270 client connection
1052, 3215	Console printer-keyboards	Telnet client connection
1052-C, 3215-C	Integrated console printer-key-	



boards Integrated on Hercules console 1442, 2501, 3505      Card readers      Disk file(s) (ASCII or EBCDIC) 3525      Card punch      Disk file (ASCII or EBCDIC) 1403, 3203, 3211      Line printers      Disk file (ASCII) 3410, 3420, 3422, 3430, 3480, 3490, 3590, 9347, 8809      Tape drives      Disk file, CDROM, or SCSI tape

many Communication and Channel-to-Channel devices      many

(( OSA ))      OSA Express adapter operating on QDIO mode. Both layer-2 and layer-3 modes are supported.      "QETH" (OSA/QDIO Ethernet Adapter) Tun/Tap Driver (( LCS ))      IBM 8232 LCS device, LCS3172 driver of a P/390, IBM 2216 router, IBM 3172 running ICP.      "LCS" (LAN Channel Station) Tun/Tap Driver (( CTCI ))      Channel-to-Channel link to host TCP/IP stack      "CTCI" Tun/Tap Driver (( CTCE ))      Enhanced Channel-to-Channel Emulation via TCP connection (true 3088 CTCA)      "CTCE" driver (( PTP ))      MPCPTP/MPCPTP6 Channel to Channel link      "PTP" Tun/Tap Driver 3310, 3370, 9332, 9335, 9336, 0671      FBA direct access storage devices      Disk file 2305, 2311, 2314, 3330, 3340, 3350, 3375, 3380, 3390, 9345      CKD direct access storage devices      Disk file 2703      Communication Line, Remote Teletype, etc.      TCP Socket

## 2.7.1 Arguments required for each device type

### Local non-SNA 3270 devices

There are no required arguments for this particular device type, but there are however several optional arguments which are discussed below.

To use this device, a tn3270 client must connect to the host machine via the port number specified on the CNSLPORT statement. A valid tn3270 device type, such as IBM-3278, must be used. See the Telnet/tn3270 Console How-To for additional information about setting up a tn3270 client.

If your tn3270 client software allows you to specify a device type suffix (e.g. [IBM-3278@001F](#) ), then you can use the suffix to connect to that specific device number, if eligible. If no suffix is specified, then your client will be connected to the first available 3270 device for which it is eligible, if any.

If you specify a specific terminal device address (via the device type suffix of your tn3270 client software), then you must be eligible to connect at that device address or your connection is immediately rejected; an alternative terminal device for which you might be eligible is not automatically selected instead.

Optional arguments:

**groupname** If a terminal group name is given on the device statement, a device type suffix with this group name can be used to indicate that a device in this group is to be used. If a group name is specified as a terminal type suffix (e.g. [IBM-3278@GROUPNAME](#) ) and there are no devices defined for that group (or there are no more available devices remaining in that group), then the connection is rejected. If no group name is specified as a terminal type suffix, then the connection will only be eligible for any terminal devices which do not have a group name specified on their device statements. The terminal group name, if specified, should be 1 to 8 alphanumeric characters, the first character being alphabetic, and it should not be a hexadecimal number. Upper and lower case letters in the group name are considered to be equivalent.

**ipaddr [ mask ]** The optional IP address and optional subnet mask specify the IP address(es) of which client(s) are allowed to connect at the device address identified by the device statement on which they appear. This provides an alternative and/or additional means of specifying to which device(s) a client tn3270 session may, or should, connect.

If the IP address of the tn3270 client trying to connect, when 'and'ed with the optional subnet mask (which defaults to 255.255.255.255 if not specified), matches the IP address entered on the device statement, then the client is eligible to connect at that device address. Otherwise the client is ineligible to connect at that address and the next available device, if any, for which the client is eligible to connect (if any) is selected instead.

If no permissible terminal devices remain (i.e. terminal devices for which the client is eligible to connect), or there are no more available terminal devices remaining, then the client connection is rejected.

The optional IP address and subnet mask may also be specified in conjunction with the previously

mentioned terminal group argument, but the terminal group argument, if specified, must be specified ahead of (i.e. before) the optional IP address and subnet mask arguments. To specify an IP address and subnet mask without also specifying a terminal group, simply use “\*” as the group name instead.

If an IP address / subnet mask are not specified, then any client tn3270 session is allowed to connect to the device (provided they are also a member of the specified terminal group, if any).

The terminal group name argument, if specified, always takes precedence over any optional IP address and subnet mask which may also be specified.

To summarize, the device number suffix always takes precedence over any group name which may also be specified, and any group name, if specified, always takes precedence over any IP address / subnet mask value which may also be specified.

### Integrated 3270 console device (SYSG)

The integrated 3270 (SYSG) console is similar to a local non-SNA 3270 device, except that it is not addressed by subchannel number and it is supported only by certain system control programs. The SYSG console is defined like a 3270 device except that the device number is ignored (but is usually specified as 0000) and the device type is specified as “SYSG”. Only one SYSG console device may be defined in a configuration.

Use tn3270 client software to connect to the SYSG console device via the port number specified on the SYSGPORT statement if specified (or to the port number specified on the CNLSLPORT statement if not specified), just as you would connect to a regular local non-SNA 3270 device. See the Telnet/t-n3270 Console How-To for additional information about setting up a tn3270 client.

The SYSG console device configuration statement recognizes optional arguments which specify group name and IP address in the same way as previously described for a local non-SNA 3270 devices. When the SYSGPORT statement is not specified, these optional arguments provide an alternate way to ensure that a given tn3270 client can connect directly to the SYSG device. If the group name and IP address arguments are not specified (and no SYSGPORT statement is specified), then the SYSG console is considered to be a member of the general pool of devices eligible for connection by any incoming tn3270 client.

### Integrated Console printer-keyboard devices (-C)

There are two optional arguments: the command prefix argument and the optional noprompt argument.

Since integrated console printer-keyboard devices use the Hercules HMC panel (Hardware Management Console) for all input and output, the command prefix is needed so Hercules can distinguish between input meant for the device and normal Hercules panel command input.

All integrated console devices must use a different command prefix and must not be a subset or superset of any other device’s command prefix. If one is not specified then the next available default is chosen from the following list:

Hex	Glyph	Description
0x2F	/	slash (default) 0x60 ` backtick 0x3D = equals 0x7E ~ tilde 0x40
@	at sign	0x24 \$ dollar 0x25 % percent 0x5E ^ caret 0x26 & amper-
sand	0x5F _	underline 0x3A : colon 0x3F ? question 0x30 0 zero 0x31
1	one	0x32 2 two 0x33 3 three 0x34 4 four 0x35 5 five 0x36 6
six	0x37 7	seven 0x38 8 eight 0x39 9 nine

If your command prefix was the “/” slash character for example, then to send a logon command to a 1052-C or 3215-C device, you would enter “/logon” on the Hercules console. Or, if your command prefix was “foo=” then you would enter “foo=logon”, etc.

When your guest operating system writes a message to an integrated console printer-keyboard device, the message displayed on the Hercules console is always prefixed with the device’s command prefix string so you can distinguish it from messages written by other integrated console devices and/or Hercules itself.

The second optional argument for integrated console printer-keyboard devices is the noprompt



keyword. If not specified, then whenever the system is awaiting input on that device, the prompting message "Enter input for console device nnnn" is displayed on the Hercules console. The noprompt option suppresses this prompting message.

### Console printer-keyboard devices

There are no required arguments for this particular device type, but there are however several optional arguments discussed below.

To use this device, a telnet client must connect to the host machine via the port number specified on the CNSLPORT statement. See the Telnet/tn3270 Console How-To for additional information about setting up a telnet or tn3270 client.

If your telnet client software allows you to specify a device type suffix (for example: `ansi@0009`), then you can use that suffix to specify the specific 1052 or 3215 device to which you wish to connect. If you do not specify a suffix in your telnet client software (or your software does not allow it), then your client will be connected to the first available 1052 or 3215 device for which it is eligible.

An optional noprompt argument may be specified on the device statement to cause suppression of the "Enter input for console device nnnn" prompt message which is otherwise normally issued to the device whenever the system is awaiting input on that device.

Additionally, a terminal group name, IP address and subnet mask may all also be optionally specified in the exact same manner as discussed in the previous Local non-SNA 3270 devices section with the exception that the "noprompt" option, if specified, must precede the other arguments.

### Card reader devices

The argument specifies a list of file names containing card images. Additional arguments may be specified after the file names:

**sockdev** Indicates the card reader is a socket device wherein the filename is actually a socket specification instead of a device filename. When used, there must only be one filename specified in the form: port or host:port or sockpath/sockname. The device then accepts remote connections on the given TCP/IP port or Unix Domain Socket, and reads data from the socket instead of from a device file. This allows automatic remote submission of card reader data. See the Hercules Socket Reader page for more details.

**eof** Specifies that unit exception status is presented after reading the last card in the file. This option is persistent, and will remain in effect until the reader is reinitialized with the **intrq** option.

**intrq** Specifies that unit check status with intervention required sense bytes is presented after reading the last card in the file. This option is persistent, and will remain in effect until the reader is reinitialized with the **eof** option.

**multifile** Specifies, when multiple input files are entered, to automatically open the next input file and continue reading whenever EOF is encountered on a given file. If not specified, then reading stops once EOF is reached on a given file and an attention interrupt is then required to open and begin reading the next file.

**ebcdic** Specifies that the file contains fixed length 80-byte EBCDIC records with no line-end delimiters.

**ascii** Specifies that the file contains variable length lines of ASCII characters delimited by LF (line feed) sequences or CRLF (carriage return line feed) sequences at the end of each line.

If neither EBCDIC nor ASCII is specified, then the device handler attempts to detect the format of the card image file when the device is first accessed. Auto-detection is not supported for socket devices, and the default is ASCII if **sockdev** is specified.

**trunc** Specifies, for ASCII files, that lines longer than 80 characters are truncated instead of producing a unit check error.

**autopad** Specifies, for EBCDIC files, that the file is automatically padded to a multiple of 80 bytes if necessary.

append Specifies that the output file, if it already exists, will not be cleared to zero bytes when it is opened. Instead, output will be appended to the end of the existing data. If the noclear argument is not specified, then any previous contents of the file is destroyed when the file is opened (i.e. the file is set to empty (truncated to zero bytes) as soon as it is opened).

`crLf` Specifies, for ASCII files, that carriage return line feed sequences are written at the end of each line. If the `crLf` argument is not specified, then line-feeds only are written at the end of each line.

nuclear (deprecated) This option is deprecated and will be removed in a future release. Please use the more aptly named append option instead.

## Line printer devices

Additional arguments may be specified after the file name:

append Specifies that the output file, if it already exists, will not be cleared to zero bytes when it is opened. Instead, output will be appended to the end of the existing data. If the append argument is not specified, then any previous contents of the file is destroyed when the file is opened (i.e. the file is set to empty (truncated to zero bytes) as soon as it is opened).

cttape= (lll=cc,lll=cc...) | name This option defines the carriage control tape to use for this printer. It is only valid for 1403 printer devices.

The option can be specified as either a series of `lll=cc` values surrounded by parentheses where `lll` identifies the carriage control tape's line number (1-255) and `cc` indicates the channel punch for that line (1-12), or as a predefined carriage control tape name.

More than one channel punch may be specified for a line by either specifying the desired list of channel punches for that line within parentheses or by simply specifying the same line number again in your list but with a different channel number each time.

The name format specifies the name of a predefined carriage control tape. Valid predefined cctape names are: legacy, fcb2std2 or fcb2id1. The default if not specified is legacy.

legacy: cctape=(1=1,7=2,13=3,19=4,25=5,31=6,37=7,43=8,63=9,49=10,55=11,61=12) fcb2std2:  
cctape=(1=1,7=2,13=3,19=4,25=5,31=6,37=7,43=8,49=9,55=10,61=11,63=12) fcb2id1:  
cctape=(1=1,8=2,15=3,22=4,29=5,36=6,43=7,50=8,57=9,64=10,71=11,78=12) Note that the lines per page  
value is not supported by the cctape option. Instead, you must specify your desired lines per page  
value separately via the lpp option. crlf Specifies that carriage return line feed sequences should be  
written at the end of each line. If the crlf argument is not specified then the default behavior is to only  
write line-feeds at the end of each line.

fcbl= ppnnnnnnnnnnnnnnnnnnnnnnnnnnnnnn | cc:lll[,cc:lll]... | name This option is only valid for 3203 or

3211 printer devices. It specifies an initial FCB image to use for this printer.

This option supports three different formats: old, new and name.

The old format argument must be exactly 26 digits long, and consist of digits from 0 to 9. The first pair of digits (pp) specifies the number of lines on a printed page (01-99). It is followed by 12 pairs of digits (nn) which specify the line number on the page (00-99) corresponding to each of the 12 possible FCB channels. Specify the line number as 00 for those channels you wish to leave undefined.

The new format is specified as a series of semicolon-delimited channel and line number pairs (cc:lll), each successive pair separated from the previous pair with a single comma, where cc indicates the channel number and lll indicates the line number for that channel (0-255). To leave a particular FCB channel undefined either specify 0 for the line number or simply don't specify that channel anywhere in your list.

The name format specifies the name of a predefined fcb image. Valid predefined fcb image names are: legacy, fcb2std2, fcb2id1 or hardware. The default if not specified is legacy.

legacy: fcb=01:01,02:07,03:13,04:19,05:25,06:31,07:37,08:43,09:63,10:49,11:55,12:61 fcb2std2:  
fcb=01:01,02:07,03:13,04:19,05:25,06:31,07:37,08:43,09:49,10:55,11:61,12:63 fcb2id1:  
fcb=01:01,02:08,03:15,04:22,05:29,06:36,07:43,08:50,09:57,10:64,11:71,12:78 hardware: fcb=01:01 Note  
that the lines per page value is not supported by the new or name formats of the fcb option. Instead,  
you must specify your desired lines per page value separately via the lpp option. fcbcheck  
(deprecated) This option is only valid for 3203 or 3211 printer devices. It requests that an attempt to  
skip to a FCB channel for which no line number has been set should cause a Unit Check to occur. This  
is the default. This option is deprecated and will be removed in a future release.

index= [-]nn | 0 Specifies the column number of the form (-31 to +31) where each print line should begin. This option is only valid for 3211 printer devices. For 3203 printer devices the option is accepted but is otherwise ignored.

Positive values prepend each print line with a number of blanks equal to one less than the index value, thus "indenting" the print line to reach the desired form column. An index value of 1 means flush-left. A negative value causes the first nn columns of each print line to be chopped off. A value of -1 will thus cause the first character of each print line to be dropped, thereby causing the first column of the page to begin with the second character of each printed line.

The default is 0 meaning to print normally (i.e. indexing is disabled).

lpi= 6 | 8 Specifies the number of vertical lines per inch. The only valid values are either 6 or 8. The default is 6.

lpp= nnn | 66 Specifies the number of vertical lines per page (1-255). The default is 66.

noclear (deprecated) This option is deprecated and will be removed in a future release. Please use the more aptly named append option instead.

nofcbcheck (deprecated) This option is only valid for 3203 or 3211 printer devices. It requests that an attempt to skip to a FCB channel for which no line number has been set should suppress the Unit Check that would normally occur and to instead simply print the next line of output on the next line instead. It is the opposite of the fcbcheck option.

This option is deprecated and will be removed in a future release. The option will continue to be accepted for the time being if specified, but is otherwise completely ignored. Skips to non-existent FCB channels will always cause a Unit Check regardless of this option.

sockdev Indicates the line printer is a socket device wherein the filename is actually a socket specification instead of a device filename. When used, there must only be one filename specified in the form: port or host:port. The device then accepts remote connections on the given TCP/IP port, and writes data to the socket instead of to a device file. This allows automatic remote spooling of line printer data. The sockdev option is mutually exclusive with the crlf and append options.

If the printer filename begins with a vertical bar '|' pipe character, then it is removed and the remainder of the filename is interpreted as a command line (the name of a program or batch file followed by any necessary arguments) to which to "pipe" the printer output to. This is known as the "print-to-pipe" feature. All printer output is then sent to the piped program's stdin input, and all of the piped program's stdout and stderr output is piped back to Hercules for displaying on the hard-

wareconsole.

If the “print-to-pipe” command line contains arguments, then quotes must be placed around the entire filename, including the vertical bar, since any tokens following the filename are parsed as Hercules printer device options:

```
000E 1403 “ | /usr/bin/lpr -Phlj”          (for Unix) 000E 1403 “ | c:utilspr -s -PLPT1:” crlf
(for Windows)
```

If the “print-to-pipe” command line itself contains quotes, then the command line must be enclosed in apostrophes instead of quotes, for example:

```
000E 1403 ‘ | "c:Program FilesMy Utilspr" -s -PLPT1:’ crlf
```

Tim Pinkawa has an example which shows how the print-to-pipe feature can be used to create output in PDF format. For Windows users, SoftDevLabs provides a very nice PDF spooler for Hercules called “HercPrt”.

### **Tape devices**

Five types of tape emulation are supported (see further below): AWS, HET, FakeTape, Optical Media Attach (OMA), and real SCSI.

The only required parameter is the device filename. All other parameters are optional and must follow the filename. Use \* (asterisk) for the filename to specify an empty (unmounted) tape drive.

If the specified device filename does not exist then either an unlabeled empty tape volume will be created for you automatically or a “file not found” error will occur depending on the setting of the AUTOINIT option. Refer to the AUTOINIT option for more information.

If the file name starts with the @ (at sign) character the file instead describes an Automatic Cartridge Feeder (ACF) file containing a list of tape emulation files to be automatically loaded in succession. The syntax of each line is identical to the information that can be specified in the configuration file after the device type. If the emulation file filename in the file list is the \* (asterisk) character however, then the set of options which follow it apply to all of the remaining emulation files in the list.

Parameters are appended in succession. In all cases, if the same parameter is specified more than once, the last instance takes precedence. Therefore, it is possible to specify a set of parameters in the base configuration file, another set on a \* line, and another set for each individual line. Parameters are then appended in that order: options specified on the base device statement itself first, followed by those options specified on the \* statement, and finally those specified on each individual file list statement last.

A device filename identifying a SCSI tape device cannot be specified in an Automatic Cartridge Feeder (ACF) file list.

Refer to the distributed source-code’s README.TAPE document for additional information regarding system and application programming for tape devices and instructions regarding use of the emulated ACF (Automatic Cartridge Feeder) and AUTOMOUNT features for virtual (non-SCSI) tape devices.

#### *AWSTAPE virtual tape files*

These contain a complete tape in one file. AWSTAPE files consist of variable length EBCDIC blocks. Each block is preceded by a 6-byte header. Filemarks are represented by a 6-byte header with no data. This is the same format as is used by the P/390. The argument specifies the location of the AWSTAPE file (for example ickdsf.aws)

#### *FakeTape virtual tape files*

These contain a complete tape in one file. FakeTape files consist of variable length EBCDIC blocks. Each block is preceded by a 12-ASCII-hex-character header. Filemarks are represented by a 12-character header with no data. The FakeTape format is used by the Flex-ES system from Fundamental Software Inc (FSI). The argument specifies the location of the FakeTape file (for example ickdsf.fkt). Note: “FLEX-ES” and “FakeTape” are trademarks of Fundamental Software, Inc.

HET virtual tape files (Hercules Emulated Tape) These contain a complete tape in one file and have the same structure as the AWSTAPE format with the added ability to have compressed data. The first argument specifies the location of the HET file. The filename must end with “.het” to be recognized

by Hercules as an HET file. (for example 023178.het)

Additional arguments that allow you to control various HET settings are:

**AWSTAPE** The AWSTAPE argument causes HET files to be written in AWSTAPE format. This basically, disables the additional features provided by the HET format.

**COMPRESS=n IDRC=n** COMPRESS and IDRC control whether compression should be used when writing to HET files. The value *n* can be 1 to turn on compression (the default) or 0 to turn it off. IDRC is currently a synonym for COMPRESS, but may be used in the future to control other emulated tape drive features.

**METHOD=n** The METHOD option allows you to specify which compression method to use. You may specify 1 for ZLIB compression or 2 for BZIP2 compression. The default is 1.

**LEVEL=n** The LEVEL option controls the level of compression. It ranges from 1 for fastest compression to 9 for best compression. The default is 4.

**CHUNKSIZE=nnnnn** The CHUNKSIZE option allows you to create HET files that contain different chunk sizes. The AWSTAPE (and therefore the HET) format allows each tape block to be logically broken up into smaller chunks. For instance, if your S/3x0 application creates tapes with a block size of 27998, those blocks would be broken down into nnnnn sized chunks. The range is from 4096 to 65535, the latter being the default. Decreasing the value from its default may reduce compression performance. For compatibility with AWSTAPE files created by the P/390, specify AWSTAPE with **CHUNKSIZE=4096**.

The following additional parameters apply to AWS, HET and FakeTape virtual tape files: **MAXSIZE=n[s]** | **MAXSIZEK=n** | **MAXSIZEM=n** Specifies the maximum size (in bytes, Kilobytes or Megabytes) that the emulated file is allowed to grow to.

Or, nns where *s* is either K - KILO, M - MEGA, G - GIGA, or T - TERA - BYTES

Specifying zero for this parameter means “unlimited” (i.e. there is no limit). Note: “T” is not available on all platforms.

**EOTMARGIN=n[s]** Specifies the number of bytes remaining before reaching maxsize at which point the tape device will signal the presence of the “End of Tape” marker (reflector), thus allowing the program to switch to the next tape.

Value is either *n* = bytes, or *ns* where *s* is either a K, M, G, or T multiplier.

**READONLY=n** Specifies whether the tape is mounted read-only (without a write ring or with the cartridge protect switch set to “write protect”). A parameter of 1 means read-only; a parameter of 0 means read-write. If **READONLY=1**, **RO** or **NORING** is not specified, the default is **READONLY=0**. Note that **READONLY=0** does not override the host system file permission settings for the underlying AWS or HET file. If the AWS or HET file is marked read-only, the tape will be mounted read-only despite specification of **READONLY=0**.

**RO NORING** Specifies that the tape is mounted read-only (without a write ring or with the cartridge protect switch set to “write protect”). **RO** and **NORING** are equivalent to **READONLY=1**.

**RW RING** Specifies that the tape should be mounted read-write, if possible. **RW** and **RING** are equivalent to **READONLY=0**. This is the default if **RO**, **NORING** or **READONLY=1** is not specified. Note that **RW** and **RING** do not override the host system file permission settings for the underlying AWS or HET file. If the AWS or HET file is marked read-only, the tape will be mounted read-only despite specification of **RW** or **RING**.

**DEONIRQ=n** Specifies whether a device end is presented if intervention is required during tape motion. A parameter of 1 selects this option; a parameter of 0 turns it off.

**NOAUTOMOUNT** Indicates support for guest-initiated automatic tape volume mounting is to always be disabled for this tape device.

Automatic guest tape-mount support is automatically globally enabled for all virtual (non-SCSI) tape devices by default whenever an allowable automount directory is defined via the **AUTOMOUNT** configuration file statement or the automount panel command. The **NOAUTOMOUNT** option allows you to specifically disable such support for a given device.

The automount feature enables software running in guest operating systems to automatically mount, unmount and/or query for themselves the host “virtual tape volume” filename mounted on a tape drive, via the use of special CCW opcodes (0x4B Set Diagnose and 0xE4 Sense Id) without any intervention on the part of the Hercules operator. An example of such a program for DOS/VSE called TMOUNT is provided in the util subdirectory of the distributed source code.

This is a sticky option. When specified, automount support for the device remains disabled until the option is specifically removed via a devinit command without the option specified. This means if NOAUTOMOUNT is enabled for a device while global automount functionality is currently disabled (because no AUTOMOUNT statement was specified at Hercules startup), then automount functionality remains disabled for the device even should global automount functionality be later manually enabled via an automount panel command.

When the 0x4B Set Diagnose CCW is used to auto-mount a virtual tape volume onto a given tape drive, an absolute (fully-qualified) pathname should normally always be specified, but need not be if a path relative to the currently defined “default allowable” automount directory is used instead.

The default allowable automount directory is always the first “allowable” directory that was defined, or else the current directory if no allowable directories were specifically defined. (There is always a default allowable directory whenever any allowable or unallowable automount directories are defined.)

Fully-resolved, absolute-full-path filenames are defined as being those which, for Windows, have a ‘:’ (colon) in the second position or, for other host operating systems (e.g. Linux), have a ‘/’ (slash) in the first position. Paths which start with a ‘.’ (period) are considered relative paths and will always be appended to the currently defined default allowable automount directory, before being resolved into fully-qualified paths by the host system. (I.e. only fully-resolved absolute pathnames are used in the performance of the actual automatic tape volume mount.)

For example, if more than one allowable automount directory is defined and the volume wishing to be mounted happens to reside in the second one, then a fully-qualified absolute pathname should of course be specified (or else one that is relative to the default directory which happens to resolve to the desired file).

All attempts to automount host files in a “disallowed” directory or any of its subdirectories will be rejected. Similarly any attempt to automount a file which is not within any “allowable” directory or subdirectory will be rejected. An error message is always issued in such cases. A message is also issued whenever a successful mount or unmount is performed.

A sample guest automount program called TMOUNT for the DOS/VSE operating system is provided in the util subdirectory of the distributed source code.

Optical Media Attach (OMA) virtual tape files These are read-only files which usually (but do not necessarily have to) reside on CDROM. OMA virtual tapes consist of one file corresponding to each physical file of the emulated tape. An ASCII text file called the Tape Descriptor File (TDF) specifies the names of the files which make up the virtual tape. The argument specifies the name of the tape descriptor file (for example /cdrom/tapes/uaa196.tdf)

The format of a Tape Descriptor File (TDF) looks like this:

```
@tdf /home/ivan/hercules/systems/DEB001/kernel.debian    FIXED RECSIZE 80 /home/ivan/hercules/systems/DEB001/parmfile.debian    TEXT /home/ivan/hercules/systems/DEB001/initrd.debian    FIXED RECSIZE 80
or: @TDF "C:UsersFishHercGUI_ZzLinux Debian-9.11kernel.debian"    FIXED RECSIZE 80
    "C:UsersFishHercGUI_ZzLinux Debian-9.11parmfile.debian"    TEXT "C:UsersFishHercGUI_ZzLinux Debian-9.11initrd.debian"    FIXED RECSIZE 80 TM TM EOT
or: @tdf kernel.debian    FIXED RECSIZE 80 parmfile.debian    TEXT initrd.debian    FIXED RECSIZE 80
```

The filename on each record can be either absolute or relative and must be enclosed within double quotes if it contains any spaces. If relative, the name is appended to the path of the primary OMA (TDF) to create the full path filename of the file being referenced. Each file record must be followed by a file format (and possibly some additional options) indicating how the file should be interpreted.

Each file listed in the TDF file must be in one of four formats:

UNDEFINED RECSIZE nnnnn UNDEFINED files are treated identically to FIXED files.

FIXED RECSIZE nnnnn FIXED files consist of fixed length EBCDIC blocks of the specified length (nnnnn)

TEXT TEXT files consist of variable length ASCII records delimited by carriage return line feed sequences at the end of each record. Each record is translated to EBCDIC and presented to the program as one physical tape block.

HEADERS HEADERS files consist of variable length EBCDIC blocks. Each block is preceded by a 16-byte header.

If you have any IBM manuals in Bookmanager format on CDROM, you can see some examples of TDF files in the TAPES directory on the CDROM. The official OMA/2 format is described in manuals SC52-1200-00 "Optical Media Attach/2 User's Guide" and SC52-1201-00 "Optical Media Attach/2 Technical Reference", but both are impossible to find.

Real SCSI attached tape drives These are real tape drives attached to the host machine via a SCSI interface. Hercules emulation always makes the drive appear as a channel attached device such as 3420 or 3480, although the underlying physical drive may be any type of SCSI attached tape drive, including 4mm or 8mm DAT, DLT, SCSI attached 3480/3490/3590 cartridge drives, and SCSI attached 3420 open reel tape drives.

Host-attached SCSI tapes are read and written using variable length EBCDIC blocks and filemarks exactly like a mainframe tape volume, and as a result can be freely used and/or exchanged on either one (i.e. SCSI tapes created on a real mainframe can subsequently be read by Hercules just fine, and a SCSI tape created by Hercules can be subsequently read on a mainframe just fine, thus providing a convenient means of exchanging data between the two).

If you plan on using SCSI tapes with Hercules you might also be interested in the SCSIMOUNT configuration option.

The only required device statement parameter for SCSI attached tape drives is the name of the device as it is known by the host operating system, usually `"/dev/nst0"` (for Linux or Windows) or `"\\.\Tape0"` (for Windows only), where '0' means tape drive number 0 (your first or only host-attached SCSI tape drive), '1' means your second host-attached SCSI tape drive, etc.

Depending on what actual model of SCSI tape drive you actually have and how it behaves, you may need to specify one or more additional optional parameters for Hercules to provide proper emulation of the desired device type. For example: a Quantum 'DLT' (Digital Linear Tape) SCSI tape drive does not return nor use a block-id format compatible with 3480/3490 drives (it instead uses a full 32-bit block-id just like the model 3590 does). It also does not support the Erase Gap command at all.

Thus, in order to use, for example, a Quantum DLT drive with Hercules, you MUST specify some special additional options to prevent the Erase Gap command from being issued to the drive as well as to inform Hercules that the drive uses 32-bit block-ids.

Please note that the below options define how the actual SCSI hardware device behaves, which is completely different from the way the emulated device will appear to behave to your guest. That is to say, if you define your tape drive to Hercules as a 3480 device, then Hercules will perform 3480 device type emulation such that the device appears to your guest o/s as if it were a 3480 device. If the actual SCSI device behaves as a 3590 device however (perhaps using/returning 32-bit block-ids instead of the expected 22-bit format block-ids that 3480's use), then you MUST specify the `-blkid-32` special option on your Hercules device statement so that Hercules's emulation logic can know that it needs to translate 22-bit block-ids to 32-bit ones before sending them to the actual SCSI hardware (and vice versa: to translate 32-bit block-ids from the actual SCSI drive into 22-bit format block-ids that your guest expects from a 3480 device).

Special options for SCSI tapes As explained just above, certain model SCSI tape drives such as the Quantum DLT series may require special handling in order to provide the desired proper device type emulation. These special options are:

`-blkid-22` The complete opposite of the below `-blkid-32` option. Use this option if your real SCSI tape drive behaves as a 3480/3490 and expects 22-bit block-ids, but you wish to define the drive to



Hercules as a 3590 tape drive (which uses 32-bit block-ids).

`-blkid-32` This option indicates that your SCSI attached tape drive only supports 32-bit block-ids (as used by 3590 drives) and not the 22-bit format used by 3480/3490 drives. You should only specify this option if you intend to define the drive as a model 3480 or 3490 device to Hercules, and then only if your actual real SCSI drive uses 32-bit block-ids.

If you define your Hercules tape drive as a model 3590 device however, then this option is not needed since model 3590 drives are already presumed to use 32-bit block-ids.

Specifying this option on a 3480/3490 device statement will cause Hercules device emulation logic to automatically translate the actual SCSI tape drive's 32-bit block-id into 22-bit format before returning it back to the guest operating system (since that is the format the guest operating system expects it to be in for a model 3480/3490 drive), and to translate the guest's 22-bit format block-id into 32-bit format before sending it to the actual SCSI hardware (since that is the format that the actual hardware requires it to be in).

`-no-erg` This option is intended to prevent issuance of the Erase Gap command to those SCSI tape drives which do not support it (such as the Quantum DLT series). It causes Hercules's device emulation logic to ignore any Erase Gap commands issued to the drive and to return immediate 'success' instead.

This option should only be used (specified) for drives such as the Quantum, which support switching from read mode to write mode in the middle of a data stream without the need of an intervening Erase Gap command. Specifying it for any other model SCSI drive may cause incorrect functioning as a result of the Erase Gap command not being issued to the actual SCSI hardware.

Check the manufacturer information for your particular model of SCSI attached tape drive (and/or use SoftDevLabs's "ftape" Windows utility) to determine whether or not this option is needed for your particular drive.

`-online` Use this option if your host's magtape device driver sets the GMT\_ONLINE status flag to report when a tape is mounted instead of the more common behavior of clearing the GMT\_DR\_OPEN flag.

Debian 8.3 Linux systems running on HP (Hewlett Packard) hardware using StorageTek 9840 fiber channel drives or DEC TSZ07 9-track drives (via a fc/SCSI bridge) for example, are known to not use the GMT\_DR\_OPEN flag at all. Rather, they set the GMT\_ONLINE status flag to indicate when a tape is mounted, thus requiring the `-online` option to work properly with Hercules.

### Communication and Channel-to-Channel devices

The first argument defines the emulation type, and the remaining arguments depend on the chosen emulation type.

The following are the emulation types that are currently supported:

**QETH** (OSA/QDIO Ethernet Adapter)

Emulates an OSA Express card running in QDIO mode. Both layer-2 and layer-3 are currently supported. The mode of operation is selected by the emulated workload and cannot be configured from Hercules.

The QETH device is a "grouped" device that requires 3 (three) device addresses (device numbers) to be defined per QETH group, with the first device of the group being an even numbered device:

0600.3 QETH [arguments...] You may also optionally use OSA or OSD as the device/emulation type instead of QETH if desired: 0600.3 OSA [arguments...] 0600.3 OSD [arguments...]

.. note:: Hercules's networking support requires privileged access to your host's networking devices. The easiest way to do this on Linux is to enable `setuid` of `hercifc` using the `-enable-setuid-hercifc` configure option. If that's not an option, or you're not running under Linux, using Administrative (root) privileges when running Hercules will work as well.

Please note that this is still considered to be an experimental driver still under development. Not all of the features or functionality that real OSA devices have are currently supported. For example, the current implementation only supports three devices: the read device, the write device and



the datapath device. Real OSA devices support multiple datapath devices. Support for this feature is planned, but is not yet implemented.

**Arguments:** `iface` interface Only available on \*nix Specifies the interface name for the device to be created (e.g. `tun`, `tun0`, etc).

`ipaddr` address `ipaddr` address/prefix Required on Windows Specifies the IPv4 address to be assigned to your virtual OSA Express card. This IP Address should match the IP address in your guest's TCPIP PROFILE, although this is not a requirement; during guest initialization your guest should automatically assign the proper IP address to your OSA device.

The address can be optionally followed by a prefix size expressed in CIDR notation, e.g. `192.168.1.1/24`. For IPv4 the prefix size can have a value from 0 to 32. If not specified a value of 32 is assumed. The prefix size is used to produce an equivalent subnet mask. For example, a value of 24 produces a subnet mask of `255.255.255.0`. Otherwise the subnet mask must be specify via the `netmask` parameter.

`netmask` mask Specifies the subnet mask to be used with your OSA card. On Windows, this value should be the same subnet mask that your Windows system is using (i.e. that the 'iface' adapter is using).

The `netmask` option may only be specified when the subnet mask is not already defined via the optional prefix size parameter of the `ipaddr` option.

`iface` device Specifies, on Linux, the name of the host tunnel device to use. On Windows, this required option identifies your Windows system's actual network adapter. On Linux, the value should be specified by name (e.g. `/dev/net/tun`). On Windows, the value should be specified by either IP or MAC address.

The default value for this option is the same value specified in your NETDEV configuration file statement (or the default value if not specified).

`ipaddr6` address `ipaddr6` address/prefix Specifies the IPv6 address to be assigned to your OSA card.

The address can be optionally followed by the prefix size expressed in CIDR notation, e.g. `2001:d-b8:3003:1::543:210f/48`. For IPv6 the prefix size can have a value from 0 to 128. If not specified a value of 128 is assumed. The prefix size is used to produce an equivalent subnet mask.

`hwaddr` MAC Specifies the MAC address to be assigned to your OSA card.

If not specified then one will be internally generated in the range `02:00:5E:80:00:00` - `02:00:5E:FF:FF:FF` using the low order 23 bits of the IPv4 address. For example, if the ipv4 address is `10.1.2.3` the generated MAC address will be `02:00:5E:81:02:03`.

---

**Note** The MAC address you specify for this option MUST have the 02 locally assigned MAC bit on in the first byte, must NOT have the 01 broadcast bit on in the first byte, and MUST be unique as seen by all other devices on your network segment. It should never, for example, be the same as the host adapter MAC address specified on the `iface` parameter.

---

`mtu` bytes Specifies the Maximum Transmission Unit to be used. The maximum transmission unit (MTU) is the largest packet size, measured in bytes, that can be transmitted over a network.

On Windows, if the value is not specified or is larger than what CTCI-WIN's WinPCap driver can support for the specified `iface` host adapter, a warning is issued and the specified value is ignored and the maximum supported value is used instead.

`chpid` id Specifies the channel path identifier to be used with the device.

This is mostly a cosmetic value, but some guest operating systems such as z/OS might require it to operate correctly.

`debug` Enables debug logging for the device.

When logging is enabled the Hercules driver logs extra progress and status messages used to help debug an incorrectly functioning driver. The Hercules `qeth` panel command can be used to limit the type of debug information that is logged. Enter the command `help qeth` for more information.

**LCS** (LAN Channel Station Emulation) An emulated Lan Channel Station Adapter. This emulation

mode appears to the operating system running in the Hercules machine as either an IBM 8232 LCS device, the LCS3172 driver of a P/390, a 3172 running ICP (Interconnect Communications Program), or as a simple IBM 2216 router.

Beginning with SDL Hyperion version 4.4, an LCS device is now also capable of providing SNA support as well. Refer to the README.SNA document for details.

Except when defined as an SNA device, the LCS device is a “grouped” device that requires 2 (two) device addresses (device numbers) to be defined per LCS group, with the first device of the group being an even numbered device:

0E20.2 LCS [arguments] .. note:: Hercules’s networking support requires privileged access to your host’s networking devices. If Hercules is not started with Administrative (root) privileges then initialization of your networking devices will fail and your guest’s networking will not work.

Rather than a point-to-point link, this emulation creates a virtual ethernet adapter through which the guest operating system running in the Hercules machine can communicate. As such, this mode is not limited to TCP/IP traffic, but in fact will handle any ethernet frame.

There are no required parameters for the LCS emulation, however there are several options that can be specified on the device statement. Also note that on the MAC OS X platform, the long option format (-xxx) is not supported. On MAC, only the short option format (-x) should be used.

The format of the configuration statement for LCS devices is as follows:

`-n devname` or `-dev devname`

where devname is:

(Linux/Unix) the name of the Tun/Tap special character device, normally `/dev/net/tun`.

(Windows) either the IP or MAC address of the host system’s network card.

The default for this option is the value specified by the NETDEV configuration file statement. `-o filename` or `-oat filename` where filename specifies the filename of the OSA Address Table (OAT). If this option is specified, the optional `-mac` and `guestip` entries are ignored in preference to statements in the OAT. (See further below for the syntax of the OAT)

If no OAT is specified, the emulation module will create the following:

An ethernet adapter (port 0) for TCP/IP traffic only. Two device addresses will be defined (devnum and devnum + 1). `-m MAC Address` or `-mac MAC address` where ‘MAC address’ is the optional hardware address for your guest’s virtual adapter/interface in the format: hh:hh:hh:hh:hh:hh. The default value is ‘02:00:5E:nn:nn:nn’ where the :nn:nn:nn portion is constructed from the last 3 octets of the specified guestip.

---

**Note** The MAC address you specify for this option MUST have the 02 locally assigned MAC bit on in the first byte, must NOT have the 01 broadcast bit on in the first byte, and MUST be unique as seen by all other devices on your network segment. It should never, for example, be the same as the host adapter MAC address specified on the `-n` parameter.

---

---

**Note** If you use the `-oat` option, do not specify an address here. Instead, specify your desired guest adapter MAC address in your OAT file via the HWADD statement.

---

`guestip`

is an optional IP address of the Hercules (guest OS) side. Note: This is only used to establish a point-to-point routing table entry on driving system. If you use the `-oat` option, do not specify an address here.

OAT syntax The syntax for the OSA Address Table (OAT) is as follows:

```
0400 IP 00 PRI 172.21.3.32 0402 IP 00 SEC 172.21.3.33 0404 IP 00 NO
172.21.3.38 0406 IP 01 NO 172.21.2.16 040E SNA 00
```

```
HWADD 00 02:00:FE:DF:00:42 HWADD 01 02:00:FE:DF:00:43
```

ROUTE 00 172.21.3.32 255.255.255.224 where: Dev is the base device address Mode is the operation mode: IP or SNA. .. note:: the SNA operation mode is currently not implemented via the OAT file. Rather, a separate LCS device with the -e SNA device option must be specified instead. Refer to the README.SNA document for details.

Port is the virtual (relative) adapter number (i.e. port number). The read/write devices can be swapped by coding the odd address of the even-odd pair in the OAT.

Up to 4 virtual (relative) adapters (i.e. ports) 00-03 are currently supported.

For IP modes, the entry specific information is as follows:

PRI | SEC | NO Specifies where a packet with an unknown IP address is forwarded to. PRI is the primary default entry, SEC specifies the entry to use when the primary is not available, and NO specifies that this is not a default entry.

nnn.nnn.nnn.nnn Specifies the home IP address

When the operation mode is IP specify only the read (even) device number for Dev. The write (odd) address will be created automatically.

Additionally, a HWADD and/or ROUTE statement may also be included in the OAT:

HWADD pp hh:hh:hh:hh:hh:hh Use the HWADD to specify a hardware (MAC) address for a virtual adapter (port). The first parameter after HWADD specifies the relative adapter (port) to which the address is applied.

---

**Note** The MAC address you specify for this option MUST have the 02 locally assigned MAC bit on in the first byte, must NOT have the 01 broadcast bit on in the first byte, and MUST be unique as seen by all other devices on your network segment. It should never, for example, be the same as the host adapter MAC address specified on the -n parameter, nor the same as the HWADD defined for any other port.

---

ROUTE pp nnn.nnn.nnn.nnn ... The ROUTE statement is included for convenience. This requests Hercules's network configuration logic (hercific utility on Linux or CTCI-WIN on Windows) to automatically create a network route for this specified virtual adapter. Please note that it is not necessary to include point-to-point routes for each IP address in the table since this is done automatically by the Hercules device driver's emulation module.

CTCI (Channel to Channel link to TCP/IP stack) A point-to-point IP connection with the TCP/IP stack of the driving system on which Hercules is running. See the Hercules TCP/IP page for unix details, in particular the use of preconfigured interfaces.

The CTCI device is a "grouped" device that requires 2 (two) device addresses (device numbers) to be defined per CTCI group, with the first device of the group being an even numbered device:

0E20.2 CTCI [arguments] .. note:: Hercules's networking support requires privileged access to your host's networking devices. If Hercules is not started with Administrative (root) privileges then initialization of your networking devices will fail and your guest's networking will not work.

The Windows implementation is different from the unix one. See SoftDevLab's CTCI-WIN page for further details and information.

For unix systems, such as Linux, BSD, and OSX, you may use preconfigured interfaces or you may request Hercules to configure the interface for you. In the first case you must know and supply the name of the interface to use; in the second case the kernel supplies an interface name.

Required for Linux when using a preconfigured interface: ifname Specifies the interface name of an interface that is already configured. The flag -if may optionally be specified before the name.

Specify no IP addresses or other arguments as the information is already configured into the interface.

Required for Linux when not using a preconfigured interface, and for Windows: guestip Specifies the IP address of the guest operating system running under Hercules.

hostip Specifies the IP address of the host (Linux or Windows) side of the point-to-point link. This may or may not be the same as your system's regular IP address. For Windows, if the host system is

configured with DHCP, this should instead be the MAC address of the Ethernet adapter you wish to use to have Hercules communicate with the outside world.

Optional for Windows: If these arguments are specified, they must precede the required arguments.

-m MAC address or -mac MAC address where 'MAC address' is the optional hardware address for your guest's virtual adapter/interface in the format: hh:hh:hh:hh:hh:hh. The default value is '02:00:5E:nn:nn:nn' where the :nn:nn:nn portion is constructed from the last 3 octets of the specified guestip.

---

**Note** The MAC address you specify for this option MUST have the 02 locally assigned MAC bit on in the first byte, must NOT have the 01 broadcast bit on in the first byte, and MUST be unique as seen by all other devices on your network segment. It should never, for example, be the same as the host adapter MAC address specified on the -n parameter.

---

-k kernel-capture-buffer-size

-i tuntap32-i/o-buffer-size

See SoftDevLabs's CTCE-WIN page for further details and information.

Optional for both Linux and Windows: If these arguments are specified, they must precede the required arguments:

-n *name* or -dev *name*

Specifies the name of the tunnel device to use. The default is the value specified by the NETDEV configuration file statement.

-s *netmask* where *netmask* is the netmask to use for the automatically added point-to-point route in standard dotted internet notation (e.g. 255.255.255.0)

-x *name* or -if *name*

Specifies the name of the network interface to use.

There is no default for this argument as the kernel assigns an interface name if none is provided.

-d or -debug

Specifies that debugging output is to be produced on the Hercules control panel. This should normally be left unspecified.

CTCE (Enhanced Channel-to-Channel Emulation via TCP connection (true 3088 CTCA)) The CTCE device type emulates a true 3088 Channel to Channel Adapter. CTCE devices are emulated via TCP/IP connections between two or more Hercules instances.

---

**Note** Hercules's networking support requires privileged access to your host's networking devices. If Hercules is not started with Administrative (root) privileges then initialization of your networking devices will fail and your guest's networking will not work.

---

A Hercules CTCE device requires two even-odd pairs of devices, one for reading and the other for writing. In the previous CTCE version these had to be an even-odd pair of port numbers, whereby only the even port numbers had to be specified in the CTCE configuration. This restriction has now been removed. Any port number > 1024 and < 65534 is allowed. The CTCE configuration specifies the listening port number at the receiving end, the sender side port number is a free randomly chosen one.

The socket connection pairs cross-connect, the arrows showing the send->receive direction :

x-lport-random -> y-rport-config x-lport-config <- y-rport-random

CTCE connected Hercules instances can be hosted on either Unix or Windows or MacOS platforms. Neither side needs to be the same as the other. Each side can be different from the other if needed. One side can be Windows and the other side can be Linux if so desired.

The configuration statement for CTCE devices is in one of these 2 possible formats (noting that items between [] brackets are optional):

ldevnum CTCE [lport] [rdevnum=]raddress rport [[mtu]sml] [ATTNDELAY delay] [FICON]  
 ldevnum[.n] CTCE [lport] [rdevnum]=raddress [rport] [[mtu]sml] [ATTNDELAY delay] [FICON]  
 There is only one or two required positional arguments in addition to the ldevnum: ldevnum  
 The device number (CCUU) on the local system. Please note that this can optionally be followed by .n  
 in which case multiple CTCE devices can be configured with one config statement. raddress The IP  
 address of the remote system. rport The listening TCP/IP port number on the remote system. Please  
 note that this parameter is only required when the rdevnum parameter and the following equal sign  
 are not specified, or when the mtu parameter needs to be specified. When not required, the default is  
 3088. The remaining arguments are optional as per the shown [] brackets for the format chosen: lport  
 The listening TCP/IP port number on the local system. The default value is 3088. rdevnum The device  
 number (CCUU) on the remote system. The default is equal to the ldevnum on the local system.  
 Please note that in case only one or two hexadecimal digits are given (i.e. a value up to 255 and thus  
 not a complete device number CCUU), the effective rdevnum will be computed by exclusive-or of this  
 value with the ldevnum specified. In case .n is given and > 1, this will apply to all ldevnum - rdevnum  
 pairs. Please see an example of this down below. mtu Maximum Transmission Unit buffer size.  
 The default value is 62552 bytes. Please note that when this mtu parameter needs to be specified, also  
 the rport parameter needs to be given (e.g. its default value of 3088). sml Small minimum for MTU.  
 The default value is 16 bytes. delay The number of msec ATTN signals will be delayed for in order to  
 circumvent a VM/SP bug causing SIO timeout errors. Override the default of 0, e.g. with 200 or or  
 some smaller value for faster Hercules hosts, but only when needed. FICON The keyword FICON will  
 cause a Fibre channel CTC (i.e. a FCTC device) to be emulated. A sample CTCE device configuration  
 is shown below, using the previous CTCE version 1 format which is still fully supported:

First Hercules PC Host with IP address 192.168.1.100:

```
0E40 CTCE 30880 192.168.1.200 30880 0E41 CTCE 30882 192.168.1.200 30882
```

Second Hercules PC Host with IP address 192.168.1.200:

```
0E40 CTCE 30880 192.168.1.100 30880 0E41 CTCE 30882 192.168.1.100 30882
```

Exploiting the new features of the newer CTCE version 2 format, this can be simplified omitting all  
 port numbers

First Hercules PC Host with IP address 192.168.1.100:

```
0E40 CTCE 0E40=192.168.1.200 0E41 CTCE 0E41=192.168.1.200
```

Second Hercules PC Host with IP address 192.168.1.200:

```
0E40 CTCE 0E40=192.168.1.100 0E41 CTCE 0E41=192.168.1.100
```

As there are 2 equal ldevnum - rdevnum pairs, this can be simplified further:

First Hercules PC Host with IP address 192.168.1.100:

```
0E40.2 CTCE =192.168.1.200
```

Second Hercules PC Host with IP address 192.168.1.200:

```
0E40.2 CTCE =192.168.1.100
```

Showing an example of specifying a rdevnum using the exclusive-or operation with a small value:

```
0E40.4 CTCE 1=192.168.1.200
```

The above is exactly the same as this specification:

```
0E40 CTCE 0E41=192.168.1.200 0E41 CTCE 0E40=192.168.1.200 0E42 CTCE  
0E43=192.168.1.200 0E43 CTCE 0E42=192.168.1.200
```

The above example could be used to establish a redundant pair of read/write CTC links, where each  
 Hercules side uses the even devnum addresses for reading, and the odd ones for writing (or the other  
 way around). That way, the operating system definitions on each side can be identical, e.g. for  
 a VTAM MPC CTC link :

```
CTCATRL VBUILD TYPE=TRL C0E40TRL TRLE LNCTL=MPC,READ-  
=(0E40,0E42),WRITE=(0E41,0E43)
```

```
CTCALCL VBUILD TYPE=LOCAL C0E40LCL PU TRLE=C0E40TRL,XID=YES,CONNTYPE-
```

=APPN,CPCP=YES,TGP=CHANNELTRL=C0E40TRL,XID=YES,CONNTYPE=APP-  
N,CPCP=YES,TGP=CHANNEL

**PTP** (MPCPTP/MPCPTP6 Channel to Channel link) A point-to-point IP connection with the TCP/IP stack of the host system on which Hercules is running. From the point of view of the guest image running in the Hercules machine it appears to be an MPCPTP and/or MPCPTP6 ESCON CTC link to another image.

The PTP device is a “grouped” device that requires 2 (two) device addresses (device numbers) to be defined per PTP group, with the first device of the group being an even numbered device:

0460.2 **PTP** [arguments] .. note:: Hercules’s networking support requires privileged access to your host’s networking devices. If Hercules is not started with Administrative (root) privileges then initialization of your networking devices will fail and your guest’s networking will not work.

For \*nix systems, such as Linux, BSD, and OSX, you may use preconfigured interfaces or you may request Hercules to configure the interface for you. In the first case you must know and supply the name of the interface to use; in the second case the kernel supplies an interface name. See the Hercules TCP/IP page for more details.

Required for \*nix when using a preconfigured interface: [-x/-if] ifname Specifies the interface name of a TUN interface that is already configured.

Specify no host names or IP addresses or other arguments as the information is already configured into the interface.

Required for \*nix when not using a preconfigured interface, and for Windows: guest1 Specifies the host name or IP address of the guest operating system running under Hercules.

host1 Specifies the host name or IP address of the host (Linux or Windows) side of the point-to-point link.

guest2 Specifies the host name or IP address of the guest operating system running under Hercules.

host2 Specifies the host name or IP address of the host (Linux or Windows) side of the point-to-point link.

guest1 and host1 must both be of the same address family, i.e. both IPv4 or both IPv6. guest2 and host2, if specified, must both be of the same address family, i.e. both IPv4 or both IPv6, and must not be of the same address family as guest1 and host1. If a host name is specified for guest1, and the host name can be resolved to both an IPv4 and an IPv6 address, use either the -4/-inet or the -6/-inet6 option to specify which address family should be used; if neither the -4/-inet nor the -6/-inet6 option is specified, whichever address family the resolver returns first will be used. host1 or host2 can be followed by the prefix size expressed in CIDR notation, e.g. 192.168.1.1/24 or 2001:db8:3003:1::543:210f/48. For IPv4 the prefix size can have a value from 0 to 32; if not specified a value of 32 is assumed. For IPv6 the prefix size can have a value from 0 to 128; if not specified a value of 128 is assumed. For IPv4 the prefix size is used to produce the equivalent subnet mask; for example, a value of 24 produces a subnet mask of 255.255.255.0. If guest1, host1, guest2 or host2 are numeric IPv6 addresses, they can be between braces, e.g. [2001:db8:3003:1::543:210f]. Optional for \*nix: If these arguments are specified, they must precede the required arguments.

-t *size* or -mtu *size*

where *size* is the maximum transmission unit size. The default size is 1500.

-x *name* or -if *name*

Specifies the name of the TUN interface to use. There is no default for name.

Optional for Windows: If these arguments are specified, they must precede the required arguments.

-m *MAC address* or -mac *MAC address* where ‘MAC address’ is the optional hardware address for the virtual interface in the format: hh:hh:hh:hh:hh:hh. The default value is ‘02:00:5E:nn:nn:nn’ where the :nn:nn:nn portion is constructed from the last 3 octets of the specified guestip.

---

**Note** The MAC address you specify for this option MUST have the 02 locally assigned MAC bit on in the first byte, must NOT have the 01 broadcast bit on in the first byte, and MUST be unique as seen by all other devices on your network segment. It should never, for example, be the same as the host

adapter MAC address specified on the `-n` parameter.

---

`-k` kernel-capture-buffer-size

`-i` tuntap32-i/o-buffer-size

Refer to the Help file included with SoftDevLabs's CTCI-WIN product for further details and information.

Optional for both \*nix and Windows: If these arguments are specified, they must precede the required arguments:

`-n` *name* or `-dev` *name*

Specifies the name of the tunnel device to use. The default for this option is the value specified by the NETDEV configuration file statement.

`-4` or `-inet`

Indicates that when a host name is specified for `guest1`, the host name must resolve to an IPv4 address.

`-6` or `-inet6`

Indicates that when a host name is specified for `guest1`, the host name must resolve to an IPv6 address.

`-d` or `-debug`

Specifies that debugging output is to be produced on the Hercules control panel. This should normally be left unspecified.

#### CKD DASD devices

The argument specifies the name of a file containing the disk CKD DASD image or the INET address of a Hercules Shared Device server.

The file consists of a 512-byte device header record followed by fixed length track images. The length of each track image depends on the emulated device type, and is always rounded up to the next multiple of 512 bytes.

Volumes larger than 2GB (for example, the 3390 model 3) can be supported by spreading the data across more than one file. Each file contains a whole number of cylinders. The first file (which contains cylinders 0-2518 in the case of a 3390) usually has `_1` as the last two characters of its name. The `ckddasd` driver allocates the remaining files by replacing the last character of the file name by the characters 2, 3, etc.

---

**Note** When CKD DASD images are spread across multiple files, you must specify only the first file name (the file with suffix `_1`) in the configuration statement.

---

If your host operating system supports large file sizes (or 64-bit offsets) then volumes larger than 2G can be kept in a single file.

Alternatively, the argument may specify the name of a file containing a compressed CCKD DASD image. The CKD driver will automatically detect whether the file contains a regular CKD image or a compressed CCKD image.

Refer to "Creating an empty DASD volume" in the "Creating, formatting, and loading DASD volumes" section of the Creating DASD web page for information on using the 'dasdinit' command/utility to create compressed dasd files. Refer to the Compressed Dasd Emulation page for details on the actual CCKD emulation itself and additional information on the 'CCKD' initialization/-tuning control file statement, as well as information regarding the use of shadow files.

If you specify an INET address, the format is:

`ip-name-or-addr:port:devnum`

`ip-name-or-addr` specifies the internet name or address where the Hercules Shared Device server is running. `port` specifies the port number the Shared Device server is listening on. If

omitted, the default is 3990. `devnum` specifies the device number on the Shared Device server. If omitted, the default is the current device number.

In addition to the above, some additional optional arguments are also supported.

`sf=shadow-file-filename-template`

Shadow files are only supported for compressed dasd images.

A shadow file contains all the changes made to the emulated dasd since it was created, until the next shadow file is created. The moment of the shadow file's creation can be thought of as a snapshot of the current emulated dasd at that time, because if the shadow file is later removed, then the emulated dasd reverts back to the state it was at when the snapshot was taken.

Using shadow files, you can keep the base file on a read-only device such as cdrom, or change the base file attributes to read-only, ensuring that this file can never be corrupted.

Hercules console commands are provided to add a new shadow file, remove the current shadow file (with or without backward merge), compress the current shadow file, and display the shadow file status and statistics

For detailed information regarding shadow files and their use, please see the "Shadow Files" section of the Compressed Dasd Emulation web page.

`readonly`

`Readonly` returns "write inhibited" sense when a write is attempted. Note that not all of the sense bits may be getting set absolutely correctly however. (Some people have reported getting different error messages under Hercules than a real machine, but it really hasn't been an issue for a while now.)

`readonly` may be abbreviated as `rdonly` or `ro`

`fakewrite`

`Fakewrite` is a kludge for the `readonly` sense problem mentioned above. Here the disk is not intended to be updated (MVS updates the DSCB last referenced field for a `readonly` file) and any writes appear to be successful even though nothing actually gets written.

`fakewrite` may be abbreviated as `fakewrt` or `fw`

`[no]lazywrite [no]fulltrackio`

These options have been deprecated. They are still accepted, but they do absolutely nothing. `fulltrackio` may be abbreviated as `fulltrkio` or `ftio`

`cu=type`

Specifies the type of control unit to which this device is attached. The use of this parameter does not necessarily imply that all functions of the specified control unit are emulated, its only purpose is to force a particular control unit type to be indicated in the data returned by SENSE ID and similar CCW's.

The default value depends on the device type:

Device type	Default CU type
2311	2841
2314	2314
3330	3340
3350	3375
3380	3880
3390	3990
9345	9343

Other values which may be specified are: 3990-3 and 3990-6.

Normally the default value is appropriate and this parameter need not be specified.



`ser=nnnnnnnnnnnnnn`

Defines an optional overriding 12-digit serial number to be used for this device. The specified serial number will be used regardless of whatever permanent or randomly assigned serial number the device might have (if any).

### FBA DASD devices

The file consists of fixed length 512-byte records, each of which represents one physical block of the emulated disk.

The first argument specifies the name of a file which contains the FBA DASD image or the INET address of a Hercules Shared Device server.

If you specify a Shared Device server INET address, the format of the filename is:

`ip-name-or-addr:port:devnum`

`ip-name-or-addr` specifies the internet name or address where the Hercules Shared Device server is running. `port` specifies the port number the Shared Device server is listening on. If omitted, the default is 3990. `devnum` specifies the device number on the Shared Device server. If omitted, the default is the current device number.

To allow access to a minidisk within a full-pack FBA DASD image file, normal NON-compressed FBA dasds also support two additional arguments after the file name:

`origin`

Specifies the relative block number within the DASD image file at which the minidisk begins. The number must be less than the number of blocks in the file. The default origin is zero.

`numblks`

Specifies the number of 512-byte blocks in the minidisk. This number must not exceed the number of blocks in the file minus the origin. If omitted, or if specified as an \* asterisk, then the minidisk continues to the end of the DASD image file.

Compressed CFBA dasds also support an additional optional argument:

`sf=shadow-file-name`

The handling of shadow files for compressed CFBA devices is identical as that for CCKD devices. Please refer to the preceding CKD section for information regarding use of the `sf=shadow file` option.

### Communication Line - BSC

(Preliminary 2703 BSC Support)

Describes a BSC emulation line entry to either link 2 Hercules engines or a custom made program emulating a 2780, 3780 or 3x74, or a custom made program interfacing to a real BSC line. The line emulates a point-to-point BSC link. There is no point-to-multipoint handling.

The communication is emulated over a TCP connection. All bytes are transferred as-is (except for doubling DLE in transparent mode) just like it would over a real BSC link. Emulated EIA (DCD, DTR, CTS, etc..) or X.21/V.11 leads (C, T, etc..) are treated differently depending on the DIAL option selected.

The following options define the line emulation behaviour:

`DIAL=IN | OUT | INOUT | NO`

Specifies call direction (if any). If `DIAL=NO` is specified, the TCP outgoing connection is attempted as soon as an 'ENABLE' CCW is executed. Also, in this mode, an incoming connection will always be accepted. If `DIAL=IN|INOUT` is specified, a TCP incoming call is accepted ONLY if an 'ENABLE' CCW is currently executing on the device. If `DIAL=OUT`, the 'ENABLE' CCW is rejected. When `DIAL=IN|INOUT` is specified, a DIAL CCW allows the application to establish a TCP connection to a specific host. For other DIAL values, the DIAL CCW is rejected.

`LHOST=hostname | ip address | *`

Specifies which IP address to listen on. This also conditions the network interface from which incoming calls will be accepted. Specifying “\*” means all incoming TCP calls are accepted, regardless of the destination IP address or call origin. This is the default value. Specifying a specific IP address when DIAL=OUT is specified has no effect.

LPORT=service name | port number

Specifies the TCP port for which to listen to incoming TCP calls. This value is mandatory for DIAL=IN|INOUT|NO. It is ignored for DIAL=OUT.

RHOST=hostname | ip address RPORT=service name | port number

Specifies the remote host and port to which to direct a TCP connection on a DIAL=NO line when an ‘ENABLE’ CCW is executed. This option is mandatory when DIAL=NO is specified. It is ignored for other DIAL values.

The following options are tuning options. In most cases, using the default values give the best results

RTO=0 | -1 | nnn | 3000

Specifies the number of milliseconds before terminating a read on a timeout, when no read termination control character is received. Specifying 0 means the READ ends immediately. -1 specifies there is no timeout.

PTO=0 | -1 | nnn | 3000

Specifies the number of milliseconds before terminating a POLL on a timeout, when no ACK or NACK sequence is received. Specifying 0 means the POLL ends immediately. -1 specifies there is no timeout.

ETO=0 | -1 | nnn | 10000

Specifies the number of milliseconds before terminating an ENABLE operation on a timeout. the timeout applies when DIAL=NO|IN|INOUT is specified, the outgoing TCP call fails (DIAL=NO) and there is no previously or currently established TCP connection for this line. When DIAL=NO is specified, the timeout defaults to 10 seconds. For DIAL=IN|INOUT, the timeout defaults to -1.

### Communication Line - TTY

(Preliminary 2703 TELE2 TTY Support)

Describes a 2703 Telegraph Terminal Control Type II (TTY 33/35) stop/start line, providing access to the Host OS via a standard TELNET client. To the host OS the line emulates an asynchronous TELE2 connection. The communication is emulated over a TELNET connection.

The following options define the line emulation behaviour:

LPORT=port number

Specifies the TCPIP port to listen on for incoming TCP calls.

DIAL=IN

Specifies that this line is for in-bound calls. Required.

TERM=TTY

Specifies that this definition is for a TTY port. Required

### Additional 2703 Communication Line options

The following are some additional options that may also be specified for 2703 devices:

APPEND=hh... PREPEND=hh...

Specifies up to four bytes (in S/370 channel format, not ASCII) to be prepended or appended to input line data received from terminals before they are sent to the guest OS. Typical use is to add Circle D and C around each input transmission (2741's for APL360).

BINARY=NO | YES

Negotiate to telnet binary mode if TERM=RXVT4APL.

BS=DUMB BREAK=DUMB

Backspace and break key handling option.

When using windows telnet it is recommended to always use BS=DUMB and BREAK=DUMB.

CODE=EBCD | CORR | NONE

Specify code=ebcd for EBCD, code=corr for correspondence code, or code=none to disable all translation. The code= option applies to 2741 mode only.

CRLF=YES | NO

Option to map 2741 NL to TTY CRLF sequence.

CRLF2CR=YES | NO

Remove LF that immediately follow CR.

EOL=hh

Specifies the ASCII byte value that, when received, marks the end of the input line. The default is EOL=0D.

ISkip=hh...

Specifies up to four ASCII bytes that are to be suppressed during input processing.

KA=NO | (idle, intv, count)

Defines the TCP/IP keepalive settings for this line's connections. Refer the the CONKPALV statement for details.

LNCTL=TELE2 | IBM1 | BSC

Specifies the type of communications line being defined.

For asynchronous communications lines specify LNCTL=TELE2 if TERM=TTY or LNCTL=IBM1 if TERM=2741. For binary synchronous (BSC) lines specify LNCTL=BSC.

SKIP=hh...

Specifies "garbage" code points (either byte-reversed ASCII for TERM=TTY or correspondence code/EBCD for TERM=2741) that are to be suppressed in output processing, thereby allowing distinct lists to be used for different terminal types.

SEND CR=NO | YES

Send CR back to terminal when input line received.

SWITCHED=IN | OUT | INOUT | NO

Switched is just a synonym for the DIAL option.

TERM=TTY | 2741 | RXVT4APL

Specifies the terminal type. Use RXVT4APL for 8-bit and character translation in 2741 mode.

UCTRANS=YES | NO

Enable automatic translation to uppercase.



---

## Operation

---

### 3.1 Overview

Overview section goes here.

### 3.2 Other information

etc etc etc etc



---

## System Messages

---

### 4.1 Overview

This page describes the system messages for the Hercules S/370, ESA/390, and z/Architecture emulator.

### 4.2 Message Format

All Hercules-issued messages are of the form `HHCmmnnns text` where:

- `HHC` is the message prefix for Hercules. All Hercules messages will have this prefix.
- `mm` specifies the function that issued the message, from the below list
- `nnn` Specific message number, assigned more or less sequentially.
- **s Message severity**
  - `S` Severe error. Causes immediate termination of Hercules.
  - `E` Error. The function being requested did not execute correctly, but Hercules should continue running.
  - `W` Warning. Not necessarily an error, but something to take note of and possibly correct.
  - `I` Information. General messages that do not require any further action.
  - `A` Action. You need to do something.
- `text` Message text.

#### 4.2.1 CA Communication Adapter emulation

##### HHCCA001I

HHCCA001I CCUU:Connect out to ipaddr:port failed during initial status : System Cause Text Explanation HERCULES attempted to make an outgoing TCP connection to ipaddr:port, but the system indicated that there was an error while processing the request. System Action The DIAL or ENABLE CCW that caused the connection attempt ends with Unit Check and Intervention Required. The reason for the failure is indicated in the System Cause Text field Operator Action None Programmer Action Correct the RHOST/RPORT configuration statements in the configuration file. If this message occurred during a program initiated DIAL, correct the dial data. Module commadpt.c Function commadpt\_connout

**HHCCA002I**

HHCCA002I CCUU:Line Communication thread thread id started Explanation The thread responsible for asynchronous operations for the BSC emulated line CCUU has been started. System Action The system continues Operator Action None. This is an informational message Programmer Action None. This is an informational message Module commadpt.c Function commadpt\_thread

**HHCCA003E**

HHCCA003E CCUU:Cannot obtain socket for incoming calls : System Cause Text Explanation A system error occurred while attempting to create a socket to listen for incoming calls. System Action The device creation is aborted. Operator Action None. Programmer Action Check the System Cause Text for any information relating to the host system. Notify support. Module commadpt.c Function commadpt\_thread

**HHCCA004W**

HHCCA004W CCUU:Waiting 5 seconds for port port to become available Explanation While attempting to reserve port port to listen to it, the system indicated the port was already being used. System Action The system waits 5 seconds and then retries the operation Operator Action Terminate the device if the port is in error Programmer Action Determine the program holding the specified port. If the port cannot be made available, use a different port.

**HHCCA005I**

HHCCA005I CCUU:Listening on port port for incoming TCP connections Explanation The system is now listening on port port for incoming a tcp connection. System Action The system continues Operator Action None. This is an informational message Programmer Action None. This is an informational message

**HHCCA006T**

HHCCA006T CCUU:Select failed : System Cause Text Explanation An error occurred during a 'select' system call. System Action The BSC thread is terminated Operator Action None. Programmer Action Check the System Cause Text for any indication of where the error might come from. Notify Support.

**HHCCA007W**

HHCCA007W CCUU:Outgoing call failed during ENABLE|DIAL command : System Cause Text Explanation The system reported that a previously initiated TCP connection could not be completed System Action The I/O operation responsible for the TCP outgoing connection is ended with Unit Check and Intervention Required. Operator Action If the error indicates that the error is temporary, retry the operation. Programmer Action Check that the destination for this line is correctly configured. If the operation was a DIAL attempt, check in the application configuration or operation data.

**HHCCA008I**

HHCCA008I CCUU:cthread - Incoming Call Explanation The BSC thread has received an incoming call. System Action Depending on configuration and operational status, the call is either accepted or rejected. Eventually, an ongoing I/O operation may complete. Operator Action None. This is an informational message Programmer Action None. This is an informational message

**HHCCA009I**

HHCCA009I CCUU:BSC utility thread terminated Explanation The BSC thread has ended System Action the system continues. Operator Action Refer to any previous error message if this message was not unexpected Programmer Action Refer to any previous error message if this message was not unexpected

**HHCCA010I**

HHCCA010I CCUU:initialisation not performed Explanation The Device initialisation process has failed. System Action the system terminates or continues, depending on the reason for which the device was initialised. Operator Action Refer to any previous error message



Programmer Action Refer to any previous error message

#### HHCCA011E

HHCCA011E CCUU:Error parsing Keyword Explanation The device keyword parser found an error while parsing a known keyword. System Action The system continues. The device initialisation routine turns on a NOGO flag. Operator Action for a runtime initialisation, correct the device initialisation parameters, otherwise notify the programmer. Programmer Action For an engine initialisation, correct the device configuration parameters in the configuration file.

#### HHCCA012E

HHCCA012E CCUU:Unrecognized parameter Keyword Explanation The device keyword parser found an unknown keyword in the device parameter list. System Action The system continues. The device initialisation routine turns on a NOGO flag. Operator Action for a runtime initialisation, correct the device initialisation parameters, otherwise notify the programmer. Programmer Action For an engine initialisation, correct the device configuration parameters in the configuration file.

#### HHCCA013E

HHCCA013E CCUU:Incorrect local port|remote port|local host|remote host specification value Explanation The device initialisation routine could not correctly parse a parameter value. System Action The system continues. The device initialisation routine turns on a NOGO flag. Operator Action for a runtime initialisation, correct the device initialisation parameters, otherwise notify the programmer. Programmer Action For an engine initialisation, correct the device configuration parameters in the configuration file.

#### HHCCA014E

HHCCA014E CCUU:Incorrect switched/dial specification value; defaulting to DIAL=OUT Explanation The device initialisation routine found an incorrect DIAL value. System Action The system continues. The device initialisation routine turns on a NOGO flag. Operator Action for a runtime initialisation, correct the device initialisation parameters, otherwise notify the programmer. Programmer Action For an engine initialisation, correct the device configuration parameters in the configuration file.

#### HHCCA015E

HHCCA015E CCUU:Missing parameter : DIAL=NO|IN|OUT|INOUT and LPORT|RPORT|LHOST|RHOST not specified Explanation The device initialisation routine found that a mandatory parameter was not provided for a specific DIAL Value. System Action The system continues. The device initialisation routine turns on a NOGO flag. Operator Action for a runtime initialisation, correct the device initialisation parameters, otherwise notify the programmer. Programmer Action For an engine initialisation, correct the device configuration parameters in the configuration file. Note For DIAL=NO , LPORT, RPORT and RHOST are needed For DIAL=IN , LPORT is required For DIAL=OUT None of LPORT,LHOST,RPORT,RHOST are required For DIAL=INOUT, LPORT is required

#### HHCCA016W

HHCCA016W CCUU:Conflicting parameter : DIAL=NO|IN|OUT|INOUT and LPORT|RPORT|LHOST|RHOST=value specified Explanation The device initialisation routine found that a parameter was provided for a parameter that is not relevant for a specific DIAL Value. System Action The parameter is ignored. The system continues Operator Action for a runtime initialisation, correct the device initialisation parameters, otherwise notify the programmer. Programmer Action For an engine initialisation, correct the device configuration parameters in the configuration file. Note For DIAL=IN , RPORT and RHOST are ignored For DIAL=OUT , LPORT, LHOST, RPORT and RHOST are ignored For DIAL=INOUT, RPORT and RHOST are ignored

#### HHCCA017I

HHCCA017I CCUU:LPORT|RPORT|LHOST|RHOST parameter ignored Explanation The system indicates that the parameter specified is ignored. This message is preceded by message HHCCA016W System Action The system continues Operator Action None Programmer Action None

#### **HHCCA018E**

HHCCA018E CCUU:Bind failed : System Cause Text Explanation While attempting to bind a socket to a specific host/port, the host system returned an uncorrectable error. System Action BSC Thread terminates Operator Action None Programmer Action Check that the LHOST parameter for this device is indeed a local IP address. Otherwise, notify support.

#### **HHCCA019E**

HHCCA019E CCUU:BSC comm thread did not initialise Explanation The BSC communication thread reported that it terminated while the device was initialising. System Action The device is not initialised. Operator Action Check for any previously issued error message. Programmer Action Check for any previously issued error message.

#### **HHCCA020E**

HHCCA020E CCUU:Memory allocation failure for main control block Explanation A memory allocation failure occurred while attempting to reserve memory for the Communication Adapter control block System Action The device is not initialised. Operator Action None Programmer Action Contact support

#### **HHCCA021I**

HHCCA021I CCUU:Initialization failed due to previous errors Explanation The initialisation process for device CCUU did not complete successfully System Action The device is not initialised Operator Action None Programmer Action Refer to any previous error message

#### **HHCCA300D**

HHCCA300D Debug Message Explanation This is a debug message. CCW Tracing has been turned on for this device and the Line Handler issues debug messages to help diagnose interface, conformance and protocol issues. System Action The system continues Operator Action If the debug messages are no longer necessary, turn off CCW tracing (panel command : 't-CCUU'). Programmer Action None

### **4.2.2 CF Configuration file processing**

#### **4.2.3 CP CPU emulation**

#### **4.2.4 CT Channel-to-channel adapter emulation**

#### **4.2.5 CU CCKD utility**

#### **4.2.6 DA DASD emulation (both CKD and FBA)**

#### **4.2.7 DC dasdcopy**

#### **4.2.8 DG dyngui.dll**

#### **4.2.9 DI dasdinit**

#### **4.2.10 DL dasdload**

#### **4.2.11 DS dasdisup**

#### **4.2.12 DT dasdcat**

#### **4.2.13 DU DASD utilities common functions**

**4.2.14 HD Hercules Dynamic Loader****4.2.15 HE hetinit****4.2.16 HG hetget****4.2.17 HM hetmap****4.2.18 HT HTTP server****4.2.19 HU hetupd****4.2.20 IF hercifc (Network interface configuration handler)****4.2.21 IN Hercules initialization****4.2.22 LC LCS emulation****4.2.23 LG System Log functions****4.2.24 PN Hercules control panel command messages****4.2.25 PR Printer emulation****4.2.26 PU Card punch emulation****4.2.27 RD Card reader emulation****4.2.28 SD Socket devices common functions****4.2.29 TA Tape device emulation****4.2.30 TC tapecopy****4.2.31 TE 1052 and 3270 terminal emulation****4.2.32 TM tapemap****4.2.33 TS tapesplt****4.2.34 TT TOD Clock and Timer Services****4.2.35 TU TUN/TAP driver support****4.2.36 VM VM/CP emulation facility**

- genindex
- modindex

- search



