

REQUIRE file for COPY utility, COPY.REQ
Version: 'V04-000'

```

*****
*
* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
* ALL RIGHTS RESERVED.
*
* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
* TRANSFERRED.
*
* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
* CORPORATION.
*
* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*
*****

```

Created by Carol Peters, April 14, 1978, 14:23:00

Modified by

- V03-005 TSK0004 Tamar Krichevsky 6-Mar-1984
Add FIRST WILD INFILE bit to qualifier structure. This bit indicates that the current input file is the first one found for a wildcard specification.
- V03-004 TSK0003 Tamar Krichevsky 9-Feb-1984
Change severity of the the BADVALUE message from error to severe.
- V03-003 LMP0150 L. Mark Pilant, 9-Sep-1983 9:46
Correct alignment of the protection fields.
- V3-002 TSK0002 Tamar Krichevsky 12-Aug-1983
Fix /PROTECTION qualifier so that unspecified fields will be undisturbed.
- V3-001 TSK0001 Tamar Krichevsky 17-Jan-1983
Convert bit locations to field definitions for the global variables. COPY\$CLI STATUS and COPY\$SEM STATUS are now BBLOCKS, instead of BITVECTORS. Many of the field names have been changed. Others, specifically those for COPY\$SEM_STATUS, have remained.

Remove the external declarations from this module and place them in the modules which need them.

Remove the call_cli, explicit_qual, explicit_cop_qual macros.
They are unnecessary with the new CLI interface.

V203 KRM0037 Karl Malik 12-Jan-1982
add CREATEDSTM message.

V202 WMC0031 Wayne Cardoza 22-Dec-1981
Add ILLDIRCOPY message.

V201 KRM0006 K. Malik 11-Feb-1981
Add MSG\$ DIRNOTCRE for use when unable to create
a network directory.

V104 TMH0004 T. Halvorsen 17-Nov-1979
Change LIBRARY VMSMAC.L32 to REQUIRE VMSMAC.REQ.

V103 TMH0003 T. Halvorsen 16-Aug-1979
Remove fixed_overhead and move to copymain where calculation
is more relevant. Also, changed cbt message to warning.

V102 TMH0002 T. Halvorsen 26-Jul-1979
Add /VOLUME qualifier for relative volume placement
Add msg\$ cbt for contiguous best try copy
Remove tfy_timeout value since no longer used

V101 TMH0001 T. Halvorsen 14-Jul-1979
Increase working set FIXED OVERHEAD to allow for some mag
tape copying which was failing.

Include files

REQUIRE 'SRC\$:VMSMAC.REQ'; ! General purpose macro definitions

Equated symbols

LITERAL			
copy_id	= 103,	!	COPY facility identifier
append_id	= 113,	!	APPEND facility identifier
yes	= 1,	!	Used to set a single status bit on
no	= 0,	!	Used to set a single status bit off
no_file	= 0,	!	Invalid file specification routine return code
true	= 1,	!	Success code
false	= 0,	!	Failure code
ok	= 1,	!	Normal routine return code
error	= 2,	!	General error routine return code
no_more_files	= 3,	!	End of input routine success return code
no_wild_open	= 5,	!	Wildcard file cannot be opened success code
default_alloc	= 0,	!	Default output allocation size if input size is unknown
default_protect	= -1,	!	Special RMS default protection indicator

```

max_name_size      = nam$c_maxrss,      ! Maximum length of a file specification
page_size          = 512,              ! Size of a VAX page (bytes)
disk_block_size    = 512,              ! Size of a disk block (bytes)
max_io_length      = 65535,           ! Maximum length of an I/O request
single_buffer      = 1,                ! Level of I/O buffering
double_buffer      = 2,                ! Level of I/O buffering
;
```

```
! External literals for command line parsing.
```

```
EXTERNAL LITERAL
CLIS_PRESENT,      ! Entity is present on the command line
CLIS_NEGATED,      ! Entity has been explicitly negated
CLIS_LOCPRES,      ! Entity is locally present
CLIS_LOCNEG,       ! Entity has been locally negated
;
```

```
! The message codes for COPY and APPEND.
```

```

$shr_messages (msg, 0,
  (appendedb, success),      ! "<file-name> appended to <file-name> (<nn> blocks)"
  (appendedr, success),     ! "<file-name> appended to <file-name> (<nn> records)"
  (atpc, error),            ! "at PC=<location>"
  (baddelim, error),       ! "invalid delimiter"
  (badkey, error),         ! "invalid keyword"
  (badlogic, severe),      ! "internal logic error detected"
  (badlogicpc, severe),    ! "internal logic error detected at PC=<PC>"
  (badvalue, severe),      ! "invalid keyword value"
  (clib, error),           ! "CLI control block at <address>"
  (closein, error),        ! "Error closing <file-name> as input"
  (closeout, error),       ! "Error closing <file-name> as output"
  (copiedb, success),      ! "<file-name> copied to <file-name> (<nn> blocks)"
  (copiedr, success),     ! "<file-name> copied to <file-name> (<nn> records)"
  (created, info),         ! "<file-name> created"
  (dirnotcre, warning),    ! "<file-name> directory file not created"
  (hashconcat, error),     ! "Hashed file cannot be concatenated"
  (highver, warning),      ! "higher version of <file-name> already exists"
  (idxconcat, error),      ! "Indexed file cannot be concatenated"
  (incompat, warning),     ! "<file-name> and <file-name> have incompatible attr"
  (invquaval, severe),     ! "invalid value <value> for qualifier/<qualifier>"
  (newfiles, success),     ! "<number> files created"
  (notcopied, warning),    ! "<file-name> not copied"
  (notcmplt, warning),     ! "<file-name> not completely copied"
  (novalue, error),        ! "keyword requires a value"
  (openin, error),         ! "Error opening <file-name> as input"
  (openout, error),       ! "Error opening <file-name> as output"
  (overlay, info),         ! "<file-name> being overwritten"
  (readerr, error),        ! "Error reading <file-name>"
  (relconcat, error),     ! "Relative file cannot be concatenated"
  (replaced, info),        ! "<file-name> being replaced"
  (syntax, severe),        ! "syntax error"
  (wildconcat, error),     ! "Wildcard specification cannot be concatenated"
  (writeerr, error),      ! "Error writing <file-name>"
);
```

```
(cvt, warning),
(illdircopy, warning),
(createdstm, info));
```

```
! "File copied contiguous best try"
! "illegal director copy of <file-name> attempted"
! "<file-name> has been created in stream format"
```

```
! Two SEVERE error message codes.
```

LITERAL

```
msg$_openinx = msg$_openin -
               sts$_k_error + sts$_k_severe,
msg$_openoutx = msg$_openout -
               sts$_k_error + sts$_k_severe;
```

```
! "Error opening <file-name> as input"
! "Error opening <file-name> as output"
```

```
! A COPY specific message code:
```

```
$shr_messages (cpy, copy_id,
               (wildoutver, error));
```

```
! "Explicit wildcard version required for new file"
```

LITERAL

```
!++
```

```
! Bit offsets for flags in COPY$SEM STATUS. Note that the first 16 bits
! are checked by COPY$SEMANTICS to determine the characteristics of the
! output file specification. Only 10 of these bits are currently in use.
! The flags which describe the state of the input and output files begin
! at the 16th bit.
```

```
!--
```

```
concat_qual_bit      = 0,
exp_inp_ver_bit      = 1,
exp_out_ver_bit      = 2,
no_output_spec_bit  = 3,
wild_input_bit       = 4,
wild_inp_ver_bit     = 5,
wild_output_bit      = 6,
wild_out_ver_bit     = 7,
multiple_output_bit  = 8,
multiple_input_bit   = 9,
```

```
! /CONCATENATE was explicitly given
! Explicit input version number specified
! Explicit output version number specified
! No output file name, type or version specified
! Wildcard in input specification
! Input version number is wild
! Wildcard in output specification
! Output version number is wild
! Multiple output files being produced
! Multiple input files specified
```

```
quiet_slip_bit       = 16,
outfile_open_bit     = 17,
infile_open_bit      = 18,
concat_follows_bit   = 19,
concat_active_bit    = 20,
wildcard_active_bit  = 21,
record_mode_bit      = 22,
extend_outfile_bit   = 23,
no_exp_out_fields_bit = 24,
first_wild_infile_bit = 25;
```

```
! Slip lower version numbers without reporting them
! Output file is currently open
! Input file is currently open
! Input files will be concatenated
! Input files are being concatenated to an output file
! Input file spec contained a wildcard
! Record mode I/O required
! Append input file to current output file
! Output file spec fields all contained wild cards or were missing
! Curr. file is first to match wildcard spec
```

```
! Macros
```

MACRO

```

!++
: Field definitions for COPY$SEM_STATUS, which describes the conditions
: that effect the creation of the output file ( for example, an explicit
: version number was given for the output file ) and the current status of
: the output file. (Is it open? Are there multiple output files? etc.)
!--

```

```

concat_qual      = COPY$SEM_STATUS[ 0, concat_qual_bit,    1, 0]%,
exp_inp_ver      = COPY$SEM_STATUS[ 0, exp_inp_ver_bit,    1, 0]%,
exp_out_ver      = COPY$SEM_STATUS[ 0, exp_out_ver_bit,    1, 0]%,
no_output_spec   = COPY$SEM_STATUS[ 0, no_output_spec_bit, 1, 0]%,
wild_input       = COPY$SEM_STATUS[ 0, wild_input_bit,     1, 0]%,
wild_inp_ver     = COPY$SEM_STATUS[ 0, wild_inp_ver_bit,   1, 0]%,
wild_output      = COPY$SEM_STATUS[ 0, wild_output_bit,    1, 0]%,
wild_out_ver     = COPY$SEM_STATUS[ 0, wild_out_ver_bit,   1, 0]%,
multiple_output  = COPY$SEM_STATUS[ 0, multiple_output_bit,1, 0]%,
multiple_input   = COPY$SEM_STATUS[ 0, multiple_input_bit, 1, 0]%,
quiet_slip       = COPY$SEM_STATUS[ 0, quiet_slip_bit,     1, 0]%,
outfile_open     = COPY$SEM_STATUS[ 0, outfile_open_bit,   1, 0]%,
infile_open      = COPY$SEM_STATUS[ 0, infile_open_bit,    1, 0]%,
concat_follows   = COPY$SEM_STATUS[ 0, concat_follows_bit, 1, 0]%,
concat_active    = COPY$SEM_STATUS[ 0, concat_active_bit,  1, 0]%,
wildcard_active  = COPY$SEM_STATUS[ 0, wildcard_active_bit,1, 0]%,
record_mode      = COPY$SEM_STATUS[ 0, record_mode_bit,    1, 0]%,
extend_outfile   = COPY$SEM_STATUS[ 0, extend_outfile_bit, 1, 0]%,
no_expl_out_fields = COPY$SEM_STATUS[ 0, no_expl_out_fields_bit, 1, 0]%,
first_wild_infile = COPY$SEM_STATUS[ 0, first_wild_infile_bit, 1, 0]%,
;

```

MACRO

```

!++
: Field definitions for COPY$CLI_STATUS, the block which contains the
: results of the command line parse. If a bit is set, then the qualifier
: was given on the command line.

```

```

: Because the command lines should produce identical results:
: COPY/QUAL a,b/NOQUAL,c * <--> COPY a,b/NOQUAL,c */QUAL
: COPY has to pretend that all qualifiers on the output file are global
: instead of positional. This means that extra information about
: the command line needs to be kept around. So, for each positional
: qualifier (those qualifiers which effect the output file in some
: way) there are three bits in COPY$CLI_STATUS... one for the global
: presence of the qualifier, one for the local presence and the third
: for local negations of the qualifier. The longwords which contain
: values given on the command line hold only the global values. The
: local values are stored elsewhere.

```

```

: *****

```

COPY\$CLI_STATUS has the following format:

32	16	15	0	Byte offset:	
-----				0	- Global qualifier flags; 2 - Positional qualifier flags
-----				4	- Positional qualifier flags
-----				8	- /ALLOCATION value (global)
-----				12	- /EXTENSION value (global)
-----				16	- /FILE_MAXIMUM value (global)
-----				20	- /PROTECTION mask (global)
-----				24	- /VOLUME value (global)

```

append_command      = COPY$CLI_STATUS[ 0, 0, 1, 0]%,      ! APPEND command
log_msg_qual        = COPY$CLI_STATUS[ 0, 1, 1, 0]%,      ! /LOG
explicit_concat_qual = COPY$CLI_STATUS[ 0, 2, 1, 0]%,      ! /CONCATENATE
negated_concat_qual = COPY$CLI_STATUS[ 0, 3, 1, 0]%,      ! /NOCONCATENATE
new_version_qual    = COPY$CLI_STATUS[ 0, 4, 1, 0]%,      ! /NEW_VERSION

alloc_qual          = COPY$CLI_STATUS[ 2, 0, 1, 0]%,      ! /ALLOCATION
loc_alloc_qual      = COPY$CLI_STATUS[ 2, 1, 1, 0]%,      ! /ALLOCATION (locally)
neg_alloc_qual      = COPY$CLI_STATUS[ 2, 2, 1, 0]%,      ! /ALLOCATION (locally negated)

contig_qual         = COPY$CLI_STATUS[ 2, 3, 1, 0]%,      ! /CONTIGUOUS
contig_negated      = COPY$CLI_STATUS[ 2, 4, 1, 0]%,      ! /NOCONTIGUOUS
loc_contig_qual     = COPY$CLI_STATUS[ 2, 5, 1, 0]%,      ! /CONTIGUOUS (locally)
neg_contig_qual     = COPY$CLI_STATUS[ 2, 6, 1, 0]%,      ! /CONTIGUOUS (locally negated)

extend_qual         = COPY$CLI_STATUS[ 2, 7, 1, 0]%,      ! /EXTENSION
loc_extend_qual     = COPY$CLI_STATUS[ 2, 8, 1, 0]%,      ! /EXTENSION (locally)
neg_extend_qual     = COPY$CLI_STATUS[ 2, 9, 1, 0]%,      ! /EXTENSION (locally negated)

file_max_qual       = COPY$CLI_STATUS[ 2, 10, 1, 0]%,     ! /FILE_MAXIMUM
loc_file_max_qual   = COPY$CLI_STATUS[ 2, 11, 1, 0]%,    ! /FILE_MAXIMUM (locally)
neg_file_max_qual   = COPY$CLI_STATUS[ 2, 12, 1, 0]%,    ! /FILE_MAXIMUM (locally negated)

protect_qual        = COPY$CLI_STATUS[ 2, 13, 1, 0]%,     ! /PROTECTION
loc_protect_qual    = COPY$CLI_STATUS[ 2, 14, 1, 0]%,    ! /PROTECTION (locally)
neg_protect_qual    = COPY$CLI_STATUS[ 2, 15, 1, 0]%,    ! /PROTECTION (locally negated)

read_chk_qual       = COPY$CLI_STATUS[ 4, 0, 1, 0]%,      ! /READ_CHECK
loc_read_chk_qual   = COPY$CLI_STATUS[ 4, 1, 1, 0]%,      ! /READ_CHECK (locally)
neg_read_chk_qual   = COPY$CLI_STATUS[ 4, 2, 1, 0]%,      ! /READ_CHECK (locally negated)

write_chk_qual      = COPY$CLI_STATUS[ 4, 3, 1, 0]%,      ! /WRITE_CHECK
write_chk_negated   = COPY$CLI_STATUS[ 4, 4, 1, 0]%,      ! /NOWRITE_CHECK
loc_write_chk_qual  = COPY$CLI_STATUS[ 4, 5, 1, 0]%,      ! /WRITE_CHECK (locally)
neg_write_chk_qual  = COPY$CLI_STATUS[ 4, 6, 1, 0]%,      ! /WRITE_CHECK (locally negated)

overlay_qual        = COPY$CLI_STATUS[ 4, 7, 1, 0]%,      ! /OVERLAY
loc_overlay_qual    = COPY$CLI_STATUS[ 4, 8, 1, 0]%,      ! /OVERLAY (locally)
neg_overlay_qual    = COPY$CLI_STATUS[ 4, 9, 1, 0]%,      ! /OVERLAY (locally negated)

```

```

volume_qual      = COPY$CLI_STATUS[ 4, 10, 1, 0]%,      ! /VOLUME
loc_volume_qual  = COPY$CLI_STATUS[ 4, 11, 1, 0]%,      ! /VOLUME (locally)
neg_volume_qual  = COPY$CLI_STATUS[ 4, 12, 1, 0]%,      ! /VOLUME (locally negated)

truncate_qual    = COPY$CLI_STATUS[ 4, 13, 1, 0]%,      ! /TRUNCATE
truncate_negated = COPY$CLI_STATUS[ 4, 14, 1, 0]%,      ! /NOTRUNCATE
loc_truncate_qual = COPY$CLI_STATUS[ 4, 15, 1, 0]%,      ! /TRUNCATE (locally)
neg_truncate_qual = COPY$CLI_STATUS[ 4, 16, 1, 0]%,      ! /TRUNCATE (locally negated)

replace_qual     = COPY$CLI_STATUS[ 4, 17, 1, 0]%,      ! /REPLACE
loc_replace_qual = COPY$CLI_STATUS[ 4, 18, 1, 0]%,      ! /REPLACE (locally)
neg_replace_qual = COPY$CLI_STATUS[ 4, 19, 1, 0]%,      ! /REPLACE (locally negated)

allocation_value = COPY$CLI_STATUS[ 8, 0, 32, 0]%,      ! /ALLOCATION value
extension_value  = COPY$CLI_STATUS[ 12, 0, 32, 0]%,     ! /EXTENSION value
file_max_value   = COPY$CLI_STATUS[ 16, 0, 32, 0]%,     ! /FILE MAXIMUM value
protection_or    = COPY$CLI_STATUS[ 20, 0, 16, 0]%,     ! /PROTECTION mask
protection_and   = COPY$CLI_STATUS[ 22, 0, 16, 0]%,     ! /PROTECTION mask
volume_value     = COPY$CLI_STATUS[ 24, 0, 32, 0]%,     ! /VOLUME value
;
```

```

:
: End of REQUIRE file, COPY.REQ
:
```

This image displays a grid of 100 terminal window screenshots, arranged in 10 rows and 10 columns. Each window shows a different system utility or data table. The windows are titled as follows:

- Row 1: COPY, COPYMSG REQ, COPY, COPYMAP, COPYREQ, COPYCLI LIS, COPYMAIN LIS, COPYSEMAN LIS
- Row 2: COPY, COPYMSG REQ, COPY, COPYMAP, COPYREQ, COPYCLI LIS, COPYMAIN LIS, COPYSEMAN LIS
- Row 3: COPY, COPYMSG REQ, COPY, COPYMAP, COPYREQ, COPYCLI LIS, COPYMAIN LIS, COPYSEMAN LIS
- Row 4: COPY, COPYMSG REQ, COPY, COPYMAP, COPYREQ, COPYCLI LIS, COPYMAIN LIS, COPYSEMAN LIS
- Row 5: COPY, COPYMSG REQ, COPY, COPYMAP, COPYREQ, COPYCLI LIS, COPYMAIN LIS, COPYSEMAN LIS
- Row 6: COPY, COPYMSG REQ, COPY, COPYMAP, COPYREQ, COPYCLI LIS, COPYMAIN LIS, COPYSEMAN LIS
- Row 7: COPY, COPYMSG REQ, COPY, COPYMAP, COPYREQ, COPYCLI LIS, COPYMAIN LIS, COPYSEMAN LIS
- Row 8: COPY, COPYMSG REQ, COPY, COPYMAP, COPYREQ, COPYCLI LIS, COPYMAIN LIS, COPYSEMAN LIS
- Row 9: COPY, COPYMSG REQ, COPY, COPYMAP, COPYREQ, COPYCLI LIS, COPYMAIN LIS, COPYSEMAN LIS
- Row 10: COPY, COPYMSG REQ, COPY, COPYMAP, COPYREQ, COPYCLI LIS, COPYMAIN LIS, COPYSEMAN LIS

The screenshots show various data tables, command prompts, and system status information. Some windows contain text like "COPYMSG REQ", "COPYCLI LIS", "COPYMAIN LIS", and "COPYSEMAN LIS". The data is presented in a structured, tabular format typical of VAX/VMS system utilities.