RC5000 File System

LOOKWD

EM Reference Manual

Keywords:
RcPAX Network System, NCP, LCP, EM, LOOKWD, File System, WD.

Abstract:
This manual describes all the Entity Management (EM) Operations for the LOOKWD software component of the RC5000 Network Switch.

Date:
November 1991

Author:
Henning Jakobsen

PN: 99001412

Table of Contents

1. Introduction

This manual describes all the Entity Management (EM) Operations for the LOOKWD software component of the RC5000 Network Switch. The LOOKWD is included in the RC5000 File System package (SW50220).

The EM Operations described in this manual are those supported by release 1.5 of the SW50220.

2. Survey of all EM Operations

Operations

Number	Name
0	Lookwd: lcp operations
1	Create and start
9	Lookup file
13	List catalog, old version
17	Pliblookup
20	Create directory
21	Free slices
24	Rename file
28	Link file
32	Copy file
33	List directory
36	Delete file

Events

Number	Name

3. Sense Operations

3. 1. Lookup file (9)

Request data

Field name	Datatype
Filename pattern	text-fix

Answer data

Field name	Datatype
Filename	text-fix
File type	text-fix
Authorization (r,w,a,d,au)	bit8
Inheritable (r,w,a,d,au)	bit8
Protections (r,w,a,d,au)	bit8
Written (in bytes)	num32
Allocated (in bytes)	num32
Length (in bytes)	num32
Creation time	hex-df20
Last update time	hex-df20
Access state or error code	text-fix

Function

Find file and return entry and descriptor.

Parameters

Filename pattern
Filename must include the whole pathname starti
ng from the root and starting with a '/'
e.g. /pax7cat/mega*

Max chars: 64 Std: /

Filename
The filename inclusive the pathname.
e.g. /pax7cat/plib2

File type
"data" for datafile.
"dir " for directory (catalog).

"????" for others.
If error then "c=x", where x is the routine call no for that operation.

Authorization (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorization.

Allocated (in bytes)
Allocation is done in slices, so allocated migth be greater than written.

Length (in bytes)
Amount of bytes in the file.

Creation time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

Last update time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

Access state or error code
State may be: closed/open/******
Error code =result from a file system routine.
5: name exists 6: name not found
11: disk offline 16: disk full
29: file limit 30: unauthorized

3. 2. Pliblookup (17)

Request data

Field name	Datatype
Filename pattern	text-fix
First wanted entry no	num16

Answer data

Field name	Datatype
Entry no	num16
Module kind	num8
Module name	text-fix
Source date	hex-df20
Object date	hex-df20
Release no	num8
Subrelease no	num8
Entry no	num16
Module kind	num8
Module name	text-fix
Source date	hex-df20

Object date	hex-df20
Release no	num8
Subrelease no	num8
Entry no	num16
Module kind	num8
Module name	text-fix
Source date	hex-df20
Object date	hex-df20
Release no	num8
Subrelease no	num8
Entry no	num16
Module kind	num8
Module name	text-fix
Source date	hex-df20
Object date	hex-df20
Release no	num8
Subrelease no	num8
Entry no	num16
Module kind	num8
Module name	text-fix
Source date	hex-df20
Object date	hex-df20
Release no	num8
Subrelease no	num8
Entry no	num16
Module kind	num8
Module name	text-fix
Source date	hex-df20
Object date	hex-df20
Release no	num8
Subrelease no	num8
Entry no	num16
Module kind	num8
Module name	text-fix
Source date	hex-df20
Object date	hex-df20
Release no	num8
Subrelease no	num8
Entry no	num16
Module kind	num8
Module name	text-fix
Source date	hex-df20
Object date	hex-df20
Release no	num8
Subrelease no	num8
Entry no	num16
Module kind	num8
Module name	text-fix
Source date	hex-df20
Object date	hex-df20
Release no	num8
Subrelease no	num8
Entry no	num16
Module kind	num8
Module name	text-fix
Source date	hex-df20
Object date	hex-df20
Release no	num8
Subrelease no	num8

Function

List up to 10 modules in a program library file.

Parameters

Filename pattern
Filename must include the whole pathname starti
ng from the root and starting with a '/'
e.g. /pax7cat/mega*

Max chars: 64 Std: /

First wanted entry no
Number of first entry to lookup.

Min-max: 1-32767 Std: 1

Entry no
The modules entry number in the plib.

Module kind
0: checksum 1: program
2,4: procedure
3,5: function
6: data

Module name
Only the first 12 characters.

Source date
year month day hour min in bcdcode
(easy to read as hexnumbers).

Object date
Date for compilation.
year month day hour min in bcdcode.

Release no
1st byte of version

Subrelease no
2nd byte of version.

Entry no
The modules entry number in the plib.

Module kind
0: checksum 1: program
2,4: procedure

3,5: function
6: data

Module name
Only the first 12 characters.

Source date
year month day hour min in bcdcode
(easy to read as hexnumbers).

Object date
Date for compilation.
year month day hour min in bcdcode.

Release no
1st byte of version

Subrelease no
2nd byte of version.

Entry no
The modules entry number in the plib.

Module kind
0: checksum 1: program
2,4: procedure
3,5: function
6: data

Module name
Only the first 12 characters.

Source date
year month day hour min in bcdcode
(easy to read as hexnumbers).

Object date
Date for compilation.
year month day hour min in bcdcode.

Release no
1st byte of version

Subrelease no
2nd byte of version.

Entry no
The modules entry number in the plib.

Module kind
0: checksum 1: program
2,4: procedure
3,5: function
6: data

Module name
Only the first 12 characters.

Source date
year month day hour min in bcdcode
(easy to read as hexnumbers).

Object date
Date for compilation.
year month day hour min in bcdcode.

Release no
1st byte of version

Subrelease no
2nd byte of version.

Entry no
The modules entry number in the plib.

Module kind
0: checksum 1: program
2,4: procedure
3,5: function
6: data

Module name
Only the first 12 characters.

Source date
year month day hour min in bcdcode
(easy to read as hexnumbers).

Object date
Date for compilation.
year month day hour min in bcdcode.

Release no
1st byte of version

Subrelease no
2nd byte of version.

Entry no
The modules entry number in the plib.

Module kind
0: checksum 1: program
2,4: procedure
3,5: function
6: data

Module name
Only the first 12 characters.

Source date
year month day hour min in bcdcode
(easy to read as hexnumbers).

Object date
Date for compilation.
year month day hour min in bcdcode.

Release no
1st byte of version

Subrelease no
2nd byte of version.

Entry no
The modules entry number in the plib.

Module kind
0: checksum 1: program
2,4: procedure
3,5: function
6: data

Module name
Only the first 12 characters.

Source date
year month day hour min in bcdcode
(easy to read as hexnumbers).

Object date
Date for compilation.
year month day hour min in bcdcode.

Release no
1st byte of version

Subrelease no
2nd byte of version.

Entry no
The modules entry number in the plib.

Module kind
0: checksum 1: program
2,4: procedure
3,5: function
6: data

Module name
Only the first 12 characters.

Source date
year month day hour min in bcdcode
(easy to read as hexnumbers).

Object date
Date for compilation.
year month day hour min in bcdcode.

Release no
1st byte of version

Subrelease no
2nd byte of version.

Entry no
The modules entry number in the plib.

Module kind
0: checksum 1: program
2,4: procedure
3,5: function

6: data

Module name
Only the first 12 characters.

Source date
year month day hour min in bcdcode
(easy to read as hexnumbers).

Object date
Date for compilation.
year month day hour min in bcdcode.

Release no
1st byte of version

Subrelease no
2nd byte of version.

Entry no
The modules entry number in the plib.

Module kind
0: checksum 1: program
2,4: procedure
3,5: function
6: data

Module name
Only the first 12 characters.

Source date
year month day hour min in bcdcode
(easy to read as hexnumbers).

Object date
Date for compilation.
year month day hour min in bcdcode.

Release no
1st byte of version

Subrelease no
2nd byte of version.

3. 3. Free slices (21)

Request data

None.

Answer data

Field name	Datatype
Volume name	text-fix
Free slices	num16
Locations	num16
Bad slices	num16
Volume size in slices	num32
Slice size in bytes	num32
Disk space used, in pct	num16
Systemversion of file system	num16
Version date	hex-df20
Date of disk formatting	hex-df20
Ship page	num32
Opcode for disk parameters	num8
Cntrlcode for disk parameters	num8
Paramsize for disk parameters	num8

Function

Read volume name and compute amount of free slices, holes, a
nd bad slices.

Parameters

Volume name
Set by the FSFORMAT program.

Free slices
Amount of free slices.

Locations
Amount of holes (areas) with free slices.

Bad slices
Amount of slices registrated as bad areas.

Slice size in bytes
=slicesize * pagesize,
pagesize =512 (1991)

Systemversion of file system
Changed from 2 to 3, 1991-05-14

Version date
year month day hour min in bcdcode (easy to read as hexnumbers).

Date of disk formatting
year month day hour min in bcdcode (easy to read as hexnumbers).

Ship page
Shipping zone of the disk, for transport and removal.

Opcode for disk parameters
Used when the disk parameters are written to the scsi controller after reset.

Cntrlcode for disk parameters
Used when the disk parameters are written to the scsi controller after reset.

Paramsize for disk parameters
Used when the disk parameters are written to the scsi controller after reset.
Amount of bytes in the parameters.

3. 4. List directory (33)

Request data

Field name	Datatype
Filename pattern	text-fix
First wanted entry no	num16

Answer data

Field name	Datatype
File entry no	num16
Filename	text-fix
File type	text-fix
Authorization (r,w,a,d,au)	bit8
Inheritable (r,w,a,d,au)	bit8
Protections (r,w,a,d,au)	bit8
Written (in bytes)	num32
Allocated (in bytes)	num32
Length (in bytes)	num32
Creation time	hex-df20
Last update time	hex-df20
File entry no	num16
Filename	text-fix
File type	text-fix
Authorization (r,w,a,d,au)	bit8
Inheritable (r,w,a,d,au)	bit8
Protections (r,w,a,d,au)	bit8
Written (in bytes)	num32

Allocated (in bytes)	num32
Length (in bytes)	num32
Creation time	hex-df20
Last update time	hex-df20
File entry no	num16
Filename	text-fix
File type	text-fix
Authorization (r,w,a,d,au)	bit8
Inheritable (r,w,a,d,au)	bit8
Protections (r,w,a,d,au)	bit8
Written (in bytes)	num32
Allocated (in bytes)	num32
Length (in bytes)	num32
Creation time	hex-df20
Last update time	hex-df20
File entry no	num16
Filename	text-fix
File type	text-fix
Authorization (r,w,a,d,au)	bit8
Inheritable (r,w,a,d,au)	bit8
Protections (r,w,a,d,au)	bit8
Written (in bytes)	num32
Allocated (in bytes)	num32
Length (in bytes)	num32
Creation time	hex-df20
Last update time	hex-df20
File entry no	num16
Filename	text-fix
File type	text-fix
Authorization (r,w,a,d,au)	bit8
Inheritable (r,w,a,d,au)	bit8
Protections (r,w,a,d,au)	bit8
Written (in bytes)	num32
Allocated (in bytes)	num32
Length (in bytes)	num32
Creation time	hex-df20
Last update time	hex-df20

Function

List selected filenames from the given directory.
Works like UNIX "ls".

Parameters

Filename pattern

Filename must include the whole pathname starti ng from the root and starting with a '/'
e.g. /pax7cat/mega*

Max chars: 64 Std: /

First wanted entry no

Number of first entry in directory to lookup.

Min-max: 1-4095 Std: 1

File entry no
The files entry no in the directory.
If error then error code from a file system routine is shown.

Filename
The filename exclusive the pathname.

File type
"data" for datafile.
"dir " for directory (catalog).
"????" for others.
If error then "c=x", where x is the routine call no for that operation.

Authorization (r,w,a,d,au)
The fields are: read, write, adjust, delete, change authorizations

Inheritable (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorization.

Protections (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorizations.

Allocated (in bytes)
Allocation is done in slices, so allocated migth be greater than written.

Length (in bytes)
Amount of bytes in the file.

Creation time
Year month day hour min in bcdcode.
(easy to read as hexnumbers)

Last update time
Year month day hour min in bcdcode.
(easy to read as hexnumbers)

File entry no
The files entry no in the directory.

Filename
The filename exclusive the pathname.

File type
"data" for datafile
"cat " for catalog (directory)
"udef" for others

Authorization (r,w,a,d,au)
The fields are: read, write, adjust, delete, change authorizations

Inheritable (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorization.

Protections (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorizations.

Allocated (in bytes)
Allocation is done in slices, so allocated migth be greater than written.

Length (in bytes)
Amount of bytes in the file.

Creation time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

Last update time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

File entry no
The files entry no in the directory.

Filename
The filename exclusive the pathname.

File type
"data" for datafile
"cat " for catalog (directory)
"udef" for others

Authorization (r,w,a,d,au)
The fields are: read, write, adjust, delete, change authorizations

Inheritable (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorization.

Protections (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorizations.

Allocated (in bytes)
Allocation is done in slices, so allocated migth be greater than written.

Length (in bytes)
Amount of bytes in the file.

Creation time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

Last update time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

File entry no
The files entry no in the directory.

Filename
The filename exclusive the pathname.

File type
"data" for datafile
"cat " for catalog (directory)
"udef" for others

Authorization (r,w,a,d,au)
The fields are: read, write, adjust, delete, change authorizations

Inheritable (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorization.

Protections (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorizations.

Allocated (in bytes)
Allocation is done in slices, so allocated migth be greater than written.

Length (in bytes)
Amount of bytes in the file.

Creation time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

Last update time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

File entry no
The files entry no in the directory.

Filename
The filename exclusive the pathname.

File type
"data" for datafile
"cat " for catalog (directory)
"udef" for others

Authorization (r,w,a,d,au)
The fields are: read, write, adjust, delete, change authorizations

Inheritable (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorization.

Protections (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorizations.

Allocated (in bytes)
Allocation is done in slices, so allocated migth be greater than written.

Length (in bytes)
Amount of bytes in the file.

Creation time
Year month day hour min in bcdcode.
(easy to read as hexnumbers)

Last update time
Year month day hour min in bcdcode.
(easy to read as hexnumbers)

4. Contol Operations

4. 1. Create and start (1)

Request data

Field name	Datatype
run family 1	num8
process name part 1	text-fix
process name part 2	text-fix
LCP ident, incarnation number	num8
LCP ident, module type	num8
stack size	num16
dummy mini family	text-fix

Answer data

Field name	Datatype
run family 1	num8
process name part 1	text-fix
process name part 2	text-fix
LCP ident, incarnation number	num8
LCP ident, module type	num8
stack size	num16
dummy mini family	text-fix

Function

Used to create and start incarnation 0 of LOOKWD.

Parameters

run family 1

Min-max: 0-255 Std: 56

process name part 1
the name as registered in the linker catalog of the process to create

Max chars: 6 Std: lookwd

process name part 2
the name as registered in the linker catalog of the process to create

Max chars: 6 Std:

LCP ident, incarnation number
the incarnation number of the process

Min-max: 0-127 Std: 0

LCP ident, module type
the lcp number of the process

Min-max: 0-255 Std: 136

stack size

Min-max: 0-32000 Std: 0

dummy mini family

Max chars: 8 Std:

process name part 1
the name as registered in the linker catalog of the process to create

process name part 2
the name as registered in the linker catalog of the process to create

LCP ident, incarnation number
the incarnation number of the process

LCP ident, module type
the lcp number of the process

4. 2. Create directory (20)

Request data

Field name	Datatype
Directory name	text-fix

Answer data

Field name	Datatype
Filename	text-fix
File type	text-fix
Authorization (r,w,a,d,au)	bit8
Inheritable (r,w,a,d,au)	bit8
Protections (r,w,a,d,au)	bit8
Written (in bytes)	num32
Allocated (in bytes)	num32

Length (in bytes)	num32
Creation time	hex-df20
Last update time	hex-df20
Access state or error code	text-fix

Function

Works like UNIX "mkdir".

Parameters

Directory name
The name must include the whole pathname starti ng from the root and starting with a '/'
e.g. /demo/subcat

Max chars: 64 Std: /

Filename
The filename inclusive the pathname.
e.g. /pax7cat/plib2

File type
"data" for datafile.
"dir " for directory (catalog).
"????" for others.
If error then "c=x", where x is the routine call no for that operation.

Authorization (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorization.

Allocated (in bytes)
Allocation is done in slices, so allocated migth be greater than written.

Length (in bytes)
Amount of bytes in the file.

Creation time
Year month day hour min in bcdcode.
(easy to read as hexnumbers)

Last update time
Year month day hour min in bcdcode.
(easy to read as hexnumbers)

Access state or error code
State may be: closed/open/******
Error code =result from a file system routine.

5: name exists 6: name not found
11: disk offline 16: disk full
29: file limit 30: unauthorized

4. 3. Rename file (24)

Request data

Field name	Datatype
Old filename	text-fix
New filename	text-fix

Answer data

Field name	Datatype
Filename	text-fix
File type	text-fix
Authorization (r,w,a,d,au)	bit8
Inheritable (r,w,a,d,au)	bit8
Protections (r,w,a,d,au)	bit8
Written (in bytes)	num32
Allocated (in bytes)	num32
Length (in bytes)	num32
Creation time	hex-df20
Last update time	hex-df20
Access state or error code	text-fix

Function

Let a file get a new name. Works like UNIX "mv".

Parameters

Old filename
Filename must include the whole pathname starti
ng from the root and starting with a '/'

Max chars: 64 Std: /

New filename
Filename must include the whole pathname starti
ng from the root and starting with a '/'

Max chars: 64 Std: /

Filename
The filename inclusive the pathname.
e.g. /pax7cat/plib2

File type
"data" for datafile.
"dir " for directory (catalog).
"????" for others.
If error then "c=x", where x is the routine
call no for that operation.

Authorization (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorization.

Allocated (in bytes)
Allocation is done in slices, so allocated migth be greater than written.

Length (in bytes)
Amount of bytes in the file.

Creation time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

Last update time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

Access state or error code
State may be: closed/open/******
Error code =result from a file system routine.
5: name exists 6: name not found
11: disk offline 16: disk full
29: file limit 30: unauthorized

4. 4. Link file (28)

Request data

Field name	Datatype
Original filename	text-fix
New filename	text-fix

Answer data

Field name	Datatype
Filename	text-fix
File type	text-fix
Authorization (r,w,a,d,au)	bit8
Inheritable (r,w,a,d,au)	bit8
Protections (r,w,a,d,au)	bit8
Written (in bytes)	num32
Allocated (in bytes)	num32
Length (in bytes)	num32
Creation time	hex-df20
Last update time	hex-df20
Access state or error code	text-fix

Function

Let a directory entry be a physical link to an existing file
Works like UNIX "ln".

Parameters

Original filename
Filename must include the whole pathname starti
ng from the root and starting with a '/'
Filename of an existing file.

Max chars: 64 Std: /

New filename
Filename must include the whole pathname starti
ng from the root and starting with a '/'
the alternative filename for the file.

Max chars: 64 Std: /

Filename
The filename inclusive the pathname.
e.g. /pax7cat/plib2

File type
"data" for datafile.
"dir " for directory (catalog).
"????" for others.
If error then "c=x", where x is the routine
call no for that operation.

Authorization (r,w,a,d,au)
The fields are: read, write, adjust, delete,
authorization.

Allocated (in bytes)
Allocation is done in slices, so allocated
migth be greater than written.

Length (in bytes)
Amount of bytes in the file.

Creation time
Year month day hour min in bcdcode.
(easy to read as hexnumbers)

Last update time
Year month day hour min in bcdcode.
(easy to read as hexnumbers)

Access state or error code
State may be: closed/open/******
Error code =result from a file system routine.
5: name exists 6: name not found
11: disk offline 16: disk full

29: file limit 30: unauthorized

4. 5. Copy file (32)

Request data

Field name	Datatype
Source filename	text-fix
Destination filename	text-fix

Answer data

Field name	Datatype
Filename	text-fix
File type	text-fix
Authorization (r,w,a,d,au)	bit8
Inheritable (r,w,a,d,au)	bit8
Protections (r,w,a,d,au)	bit8
Written (in bytes)	num32
Allocated (in bytes)	num32
Length (in bytes)	num32
Creation time	hex-df20
Last update time	hex-df20
Access state or error code	text-fix

Function

Make a duplicate of a file. Works like UNIX "cp".

Parameters

Source filename
Filename must include the whole pathname starti
ng from the root and starting with a '/'

Max chars: 64 Std: /

Destination filename
Filename must include the whole pathname starti
ng from the root and starting with a '/'

Max chars: 64 Std: /

Filename
The filename inclusive the pathname.
e.g. /pax7cat/plib2

File type
"data" for datafile.
"dir " for directory (catalog).
"????" for others.
If error then "c=x", where x is the routine
call no for that operation.

Authorization (r,w,a,d,au)
The fields are: read, write, adjust, delete, authorization.

Allocated (in bytes)
Allocation is done in slices, so allocated migth be greater than written.

Length (in bytes)
Amount of bytes in the file.

Creation time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

Last update time
Year month day hour min in bcdcode. (easy to read as hexnumbers)

Access state or error code
State may be: closed/open/******
Error code =result from a file system routine.

4. 6. Delete file (36)

Request data

Field name	Datatype
Filename pattern	text-fix

Answer data

Field name	Datatype
Filename	text-fix
File type	text-fix
Authorization (r,w,a,d,au)	bit8
Inheritable (r,w,a,d,au)	bit8
Protections (r,w,a,d,au)	bit8
Written (in bytes)	num32
Allocated (in bytes)	num32
Length (in bytes)	num32
Creation time	hex-df20
Last update time	hex-df20
Result of deletion	text-fix

Function

Delete a file. Works like UNIX "rm", but for one file only.

Parameters

Filename pattern
Filename must include the whole pathname starti
ng from the root and starting with a '/'
e.g. /pax7cat/mega*

Max chars: 64 Std: /

Filename
The filename inclusive the pathname.
e.g. /pax7cat/plib2

File type
"data" for datafile.
"dir " for directory (catalog).
"????" for others.
If error then "c=x", where x is the routine
call no for that operation.

Authorization (r,w,a,d,au)
The fields are: read, write, adjust, delete,
authorization.

Allocated (in bytes)
Allocation is done in slices, so allocated
migth be greater than written.

Length (in bytes)
Amount of bytes in the file.

Creation time
Year month day hour min in bcdcode.
(easy to read as hexnumbers)

Last update time
Year month day hour min in bcdcode.
(easy to read as hexnumbers)

Result of deletion
State may be: del ok/*r=xx
where xx is result of a file operation

PN: 99001412