

CHID

RCSL No: 31-D583

Edition: October 1979

Author: Edith Rosenberg

Title:

Corrections to RCSL No 28-D5, CF-System

1. INTRODUCTION

This manual is a complete list of corrections to the CF-System manual, collecting all changes since may 1972.

It contains minor changes of procedure descriptions as well as completely rewritten sections.

2. CORRECTIONS

2.1 Page 15-17, Protection of files:

In this chapter the subsections 'Security measures in the cf-system', 'Set-descr-cf', and 'Alarms' should be replaced by:

Security measures in the cf-system.

The cf-system has been provided with protection against the error cause 1, but not against 2 and 3.

The catalog entry of a file is used by the protection system in this way:

```
file = set <segments> <bs device> <shortclock> <update mark>
```

The last quantity is special for the cf-system:

<update mark> Either 0 or 1

- 0: The file is in read-only-state. This should be the state between runs, and this is the state accepted by open-cf.
- 1: The file is in update-state. This state must not occur between runs, and a file in update-state is rejected by open-cf.

Errors of type 1 are detected by means of the update-mark, which will be equal to one, if a processing in an update mode is terminated by an index alarm f.ex..

When a masterfile contains an update mark in the catalog entry, it also contains an internal update mark originating from the isq-system. The program 'recoveri' can be used to reestablish the contents of a destroyed masterfile, whereas the update mark in the catalog entry can be removed by the utility program 'backfile'.

The current value of shortclock is inserted in the catalog each time the processing mode is changed from read-only to an update mode, or when the procedure 'extendcf' is called.

Reactions on update mark errors:

If a masterfile contains an isq update mark when opened, the file can be read but not updated. As soon as the processing mode is changed to an update mode, the run will be terminated by the runtime alarm 'prep i 9'.

If a cf file contains an update mark in the catalog entry when opened, the file cannot even be read. The opencf procedure will terminate the run with the runtime alarm

```
***protectcf alarm:
file <file no> <file name> d.<shortclock>
upd.mark 1
```

2.2 Page 19, procedure buflengthcf:

The second parameter of buflengthcf has been changed to:

```
extendsegs_and_blocksincore (call value, integer)
```

The parameter has the meaning:

```
extend_segments shift 6 add blocks_in_core
```

Extend_segments may be 0, in which case the bufferlength is calculated without room for extension of the file.

If extend_segments \neq 0, the bufferlength is as large as needed when the file has been extended with this number of segments. The bufferlength will, however, never exceed the limit corresponding to the maximum size of the file. This value may be obtained in any case by using the parameter

```
(-1) shift 6 add blocks_in_core.
```

2.3 Page 25, procedure extendcf:

Add the following explanation:

If the value of the parameter 'segments' does not place the end of the file at a bucket or block limit, the extension will be decreased to the nearest such limit, but the extension will never be less than one bucket or block.

This means f.ex. that

```
extendcf (z, 1)
```

will extend a masterfile with one bucket and a listfile with one block.

2.4 Page 27, procedure get1:

The description of the parameter gmode should be replaced by:

gmode (call value, integer)

- =1 get the first record in the chain from current record in the motherfile
- =2 get the record next to last accessed record in the chain
- =3 get the last accessed record in the chain
- =4 get the record next to current record in the chain

The description of resultcf is:

resultcf	current record
1 found	the wanted
2 not found, end of chain	if gmode = 2 the last accessed else none
3 gmode = 3: last accessed undefined	unchanged
gmode = 4: current record not connected to this chain	

2.5 Page 29, procedure getnumbl:

resultcf	current record
1 found	the record with the given number
2 next, the specified record number did not identify an active record	the next record in ascending record number order
3 end of file. As 2, but no record exists with greater record number	none

recnocf is always returned as the record number of the current record. Dead records are not returned by the procedure.

The procedure getnumbl may thus be used to read a listfile sequentially:

```
for recnocf := 0, recnocf while result < 3 do
  getnumbl (z1, recnocf +1);
```

2.6 Page 31-32, procedures headl and headm:

After creation of the file head the size of the area will be cut down to the nearest bucket or block limit (as described for extendcf), so that a masterfile will contain at least one bucket and a masterfile at least one block.

Shortclock and contents fields are inserted in the catalog entry. Contents will be 22.1 for a masterfile and 23.1 for a listfile.

2.7 Page 35, procedure initrecm:

When resultcf = 1 the current record is now available as current record. This record is later on written to the file, so be careful not to change key or length fields as this will destroy the file.

- = 1 This call is used internally from `opencf`, to check that the catalog entry contains no update mark.
- = 2 This call is used internally from `readupdcf` and `updallcf` if the file was in read only mode. It will reserve the area process, and set the shortblock and the update mark in the catalog entry.
- = 3 This call is used internally from `readonlycf` and `closecf` if the file was in an updatemode. It removes the update mark from the catalog entry and releases the reservation of the area process.
- > 3 This call is used internally for an extended alarmprinting or from the user's program for testprinting.
The action parameter is interpreted as `cfprocno shift 12 add alarmno`
No internal states are modified by the call.

The printing from action > 3 is performed on the zone 'out', and contains selected parts of zone and share descriptors, current record, chain part, and chain table. The printing is performed for the zone given by the first parameter, as well as for the latest accessed mother zone and the latest accessed daughter zone, if any, in a recursive manner.

Warning: The procedure should never be called with action 1, 2, or 3 from the user's program.

2.10 Page 45, procedure setdescrcf:

Remove this procedure description, as the procedure is no longer contained in the cf-system.

2.11 Page 58-60, Alarms from code procedures:

Add the following description of how alarm printing is performed, and the explanation of the new alarms:

A runtime alarm from a cf code-procedure is handled in the following steps:

1. If the alarm is 'z.state', 'prep-cf', or 'prep i', then the normal short alarm text is printed, and the run terminated immediately.
2. If setjumpscf (z, alarmproc, 1,1) has been called previously, then this user specified alarm procedure is called. If this alarm procedure does not return through its final end, the next two steps will not be performed.
3. Now protectcf is called with the action parameter


```
cfprocno shift 12 add alarmno
```

 Protect performs the extended alarm printing of zone and share descriptors, current record, chain part and chain table as described for protectcf.
4. Finally the normal short alarm printing terminates the alarm actions.

The alarm address of this printing may be misleading (i.e. not cf-system 1 to 16) due to algol segmentation during the preceding alarm actions.

New or changed alarm causes:

<u>text</u>	<u>alarmno</u>	<u>explanation</u>	<u>error caused by</u>
cf-error	12	Error in block at insertion of a listfile record	file
prep i	8	Contents field of catalog entry of a masterfile < 22	file
prep i	9	Isq update mark found in a masterfile, i.e. the last updating has not been terminated.	program or file
rec.no.	19	The record number of a listfile record is <1. This may happen explicitly in a call of getnumbl as a program error, or implicitly in other procedures if the file has been destroyed.	program or file

2.12 Page 63, Alarms from external algol procedures:

The error messages

*** protectcf descrrec
*** protectcf setdescr
*** protectcf version

have been removed from the system.

2.13 Page 70, Appendix E:

This appendix is no longer relevant for the cf-system.

