COMPUTATION CENTER

TO: Time-Sharing Users

FROM: Janet Allen

SUBJECT: MEMØ, MØDIFY, and DITTØ Commands for the MAC Time-Sharing System

The MEMØ, MØDIFY, and DITTØ commands are now available, and are to be used as described in the CTSS Programmer's Guide, with the changes noted below. This system note also contains some clarifications and suggestions in using these commands, lists all printouts which may occur under error or normal conditions, and concludes with a brief program description.

I. Changes

a. The brief code for edit preprint is now

.EP XX YY

If YY is omitted, lines will be printed beginning with line XX up to and including the edit preprint line.

b. The edit mode begun by control word EDIT FROM (or .ED) may be terminated only by the control word .END EDIT (or .EE), not by another edit control word.

c. New CONTROL WORD .ERASE X (brief mode .ER X)
New control word specifies that a character X be interpreted, on input, to mean - delete the previous character. If no character is mentioned in the control word, no character will be used to delete the previous. At least one blank must follow ERASE or ER. The program will pick up the first non-blank character after that and use it to mean delete the previous. Normally, the character " until otherwise specified by this control word, will be interpreted as a delete mark. This has been implemented because the program runs in 12-bit character mode, and there is no automatic delete character (in 6-bit mode " means delete previous character and ? means delete line).

II. Clarifications

A. Footnotes

1. When there is no character | (as on the teletype), the character $ is to be used.

2. When the footnote is being typed, the user should remember to leave blank the first three characters in the first line for the footnote number to be inserted.

3. At the end of each page containing footnotes, DITTØ will first print a line of dashes before the footnotes.

4. Care must be taken, when using the resequence control word, that references to lines by the footnote control word are not made obsolete. If a line number so mentioned no longer refers to the correct line, the user should MØDIFY the memo, correcting the line either with the edit or manual mode.

B. Manual Mode

1. MAN. is printed at the beginning of each line. The line number should then be typed. A tab or blank following the line number is omitted and the line is stored beginning with the next character. If any other character immediately follows the line number, the stored line will be begin with it.

2. Manual mode may be entered while in the edit mode, but normal mode must be re-entered before the edit is ended.

C. Change Type Ball

1. Type ball names may be any of the following: UNIVER, SCRIPT, DELEGA, MANIFØ, ADJUTA. Since the first six characters after each blank in the line are examined, the user upon misspelling the name should skip a space and try again.

2. This control word supplies a convenient method of underlining. To do this the user should, after the line to be underlined, use this control word with any legal type name, and then in the following line type the underline characters. When the memo is being typed by DITTØ, the stop will occur and the user may turn the carriage back two turns and strike the carriage return DITTØ will then type the line containing the underlining.

D. The program occupies approximately 5300$_8$ locations. The rest of core is thus available for the buffers (see program description). Since the smaller the program the faster the response time, it is suggested that very long memos be written in several sections.

III. Printouts

A. Before typing out the memo, DITTØ types a preliminary page containing instructions for the user.

1. Printout when page number specified to DITTØ MEMØ WILL BEGIN WITH PAGE XX. WHEN THIS MESSAGE ENDS, LØAD PAPER AND HIT RETURN.

2. Normal Printouts

a. All comments are printed

b. For each change type ball control word (where XX is FIRST, SECØND, ..., YYYYY is the type ball)
AT THE XX STØP ØN PAGE X, CHANGE TYPE BALL TØ YYYYY, RUN CARRIAGE BACK 2 TURNS AND STRIKE A CARRIAGE RETURN.

c. Instructions to User
AT LAST STØP ØN EACH PAGE LØAD PAPER AND HIT RETURN TØ CØNTINUE

THIS MEMØ CONTAINS XX PAGE(S).
WHEN THIS MESSAGE ENDS LØAD PAPER AND HIT RETURN TØ BEGIN.

B. Error Printouts

1. LINE NUMBER NØT FØUND. LAST LINE DELETED.
This occurs when the delete control word specifies a line number not found by the program.

2. CØNTRØL WØRD ERRØR. DELTE LAST LINE.
If the line is not deleted, it will appear in the final memorandum.

3. MEMØ MAY NØT BE ENDED DURING EDIT. USE END EDIT. LAST LINE DELETED.
User has attempted to use END MEMØ OR RESEQUENCE while he is editing. That line is deleted, so that it will not destroy the previous line in the group being edited.

4. FAILURE TØ MATCH CHARACTER ØR LINE NUMBER. LAST LINE DELETED.
This printout occurs under any of the following conditions:

a) Immediately after an edit preprint line if the specified line number cannot be found.

b) Immediately after a change type ball control word line when no legal type ball name is found.

c) Immediately after a footnote control word if both $ are not found.

5. FAILURE TØ MATCH CHARACTER ØR LINE NUMBER.
Printed at the beginning of the preliminary page by DITTO when the characters specified by the footnote control word line cannot be found in the specified line.

6. FØØTNØTE REFERENCE LINE XX NØT FØUND. FØØTNØTE WILL BE IGNORED.
Line number referred to by the footnote control word line is not found. This is printed by DITTØ.

7. BEGIN EDIT LINE REACHED. EDIT MODE IS TERMINATED.
The line number of the edit from control word line has been reached. The program terminates the edit mode.

IIV. Program Description

A. General

MEMØ, MØDIFY, and DITTØ are entry points to the same program with MEMØ/MØDIFY one routine and DITTØ the other. Supervisor subroutines used are GETCOM, RDFLXA, WRFLXA, SETMEM, .LØAD, SETFUL, SETBCD, .DLETE, DØRMNT, and .DUMP.

Both routines operate in 12-bit character mode, keeping all information stored in 12-bit. The 6-bit mode may be used by typing 6BIT as the last parameter in the string after the command MEMØ, MØDIFY, or DITTØ. This was included for debugging purposes and will probably be removed. When in 6-bit mode the routines expand all input by supplying leading zeroes and just before output, discard the high order six bits.

The user specifies the name of the memo, and the MEMØ/ MØDIFY routine uses this for the first name, with MEMØ being the class name. The memo is filed by the .DUMP subroutine in binary form.

Both the MEMØ/MØDIFY and the DITTØ routines go to DØRMNT when finished.

Sixty-three lines per page are written. This includes four lines at the top of the page (two blank, a header line, and one blank) and one blank line at the bottom.

B. MEMØ/MØDIFY

MØDIFY reads in the memo specified using .LØAD and begins with the last line used to end the memo. The MEMØ/ MØDIFY routine issues six digit line numbers, incrementing by 100. Control words identified by a leading period are interpreted, some being stored for DITTØ to act upon. The following control words are interpreted by MEMØ/ MØDIFY and are then written over by the next line: EDIT FROM, END EDIT, DELETE, EDIT PREPRINT, ERASE and carriage returns indicating begin and end of manual mode.

Subroutine RDF (using RDFLXA) reads in each line expanding 6-bit data to 12-bit and checking for an erase mark if one is in effect. The line is first stored in a temporary buffer allowing a maximum of 120 characters (40 12-bit words) including the carriage return. It is then stored in the buffer by another subroutine UNPAK, which increases memory size if necessary and right adjusts the return character inserting blanks. If the line number is the same as the previous line it is stored over the previous line. In this buffer each line has in addition to the words input, seven words - line number, a pointer to the previous and to the next lines, a code word, and four others. Each line is examined for initial carriage return, indicating begin of manual mode, or for the leading period of a control word. When in manual mode another routine first prints MAN. and then using RDF, reads the line, searches for the line number and adjusts the line which is then handled as if it had been read in normally.

The buffer of lines is sorted whenever an edit (EDIT FROM or EDIT PREPRINT), resequence, or end memo is given. Whenever, in the sort, two lines with the same number are found, the later one is kept. The lines are not moved in the buffer but each pointer word is adjusted. Thus, whenever an edit or manual line replaces another the old one remains in the buffer, but the pointer words of the previous and the following lines refer to the new line.

The memo is ended by the resequence or end memo control word. Resequence causes new line numbers to be stored beginning with 000100 and sequencing by 100, and the lines to be printed. The routine tries to delete any file with the same name and then writes the new file using .DUMP. This file contains the memory size word, a word indicating which is the first line in the memo, and the buffer of lines.

C. DITTØ

DITTØ prepares the memo for output by setting up the entire memo in a buffer from which it is printed. This buffer begins with the preliminary page which contains all comments, a two-sentence instruction for each change type ball, and instructions for the user.

The specified memo file is read in, and the DITTØ routine begins a preliminary search for comments, footnotes, headers and change type ball control words. Comments are stored in the output buffer to be printed on the preliminary page. The code word of each comment is set to one which will be ignored later. The header is stored in a headline buffer. For each change type ball control word the routine stores in the output buffer the instructions which contain the name of the type ball and saves a pointer to the location so that the page number may be inserted later. For each footnote control word, the reference line is searched for and its location stored.

When the preliminary search is ended, the routine stores the final instructions for the preliminary page and begins storing each line of the memo and interpreting control words. As each change type ball is found, the current page number is stored in the appropriate instruction in the preliminary page. For each blank line on the page a one word carriage return is stored. The routine always stores a full sixty three line page so that if a page is partly blank, the routine will space to the end of the page. If one line is left on the last page it is added to the previous page.

The location in the output buffer of the end of each page and of the end of each line after which a change type ball control word was given are saved for the printing. The print routine will stop at each and expect a carriage return to continue.

When the entire memo has been stored in the output buffer, the total page count is stored in the instruction on the preliminary page and also in the headline of each page if there is one.

The routine now prepares to print but first checks to see if a page number with which to begin was specified. If so, each page before that specified is skipped, including the preliminary page and different instructions are printed.

When the entire memo has been printed and the user has hit a carriage return, the routine goes to D@RMNT.