

1987

Introduction to ms-dos and watfor-77, August 1987

Celal N. Kostem

Frederick Chapman

Follow this and additional works at: <http://preserve.lehigh.edu/engr-civil-environmental-fritz-lab-reports>

Recommended Citation

Kostem, Celal N. and Chapman, Frederick, "Introduction to ms-dos and watfor-77, August 1987" (1987). *Fritz Laboratory Reports*. Paper 2139.
<http://preserve.lehigh.edu/engr-civil-environmental-fritz-lab-reports/2139>

This Technical Report is brought to you for free and open access by the Civil and Environmental Engineering at Lehigh Preserve. It has been accepted for inclusion in Fritz Laboratory Reports by an authorized administrator of Lehigh Preserve. For more information, please contact preserve@lehigh.edu.

400.33

INTRODUCTION

TO

MS-DOS AND WATFOR-77

FRITZ ENGINEERING
LABORATORY LIBRARY

by

Celal N. Kostem

Frederick W. Chapman

Fritz Engineering Laboratory
Department of Civil Engineering
Lehigh University
Bethlehem, Pennsylvania

August, 1987

Fritz Engineering Laboratory Report No. 400.33

1. INTRODUCTION

The material contained in this document was assembled to assist the beginning users of microcomputers with MS-DOS operating system (disk operating system by Microsoft), WATFOR-77 FORTRAN compiler and WEDIT editor by WATCOM. The material presented herein was extracted from voluminous publications by Microsoft and WATCOM products. For advanced applications of MS-DOS and WATFOR-77, it is suggested that the readers refer to these parent publications.

This document is subject to periodic modifications, as the versions of MS-DOS and WATFOR-77 used at Lehigh University are updated. The descriptions contained in the document are based on the diskettes employed by ENG-1: Engineering Computations course in the College of Engineering and Applied Science of the University. If there are any doubts regarding the compatibility of the document versus the ENG-1 diskettes, it is strongly recommended that the prospective user check with the User Services of the Lehigh University Computing Center (LUCC) or the course leader of ENG-1.

This document has evolved from various handouts prepared by LUCC staff and ENG-1 course leader. These handouts were prepared primarily by Drs. G. Oberlander and S. K. Tarby.

2. MS-DOS COMMANDS FOR WATFOR-77 USERS

2.1 Starting Your MS-DOS Computer

1. You will need a "boot diskette" to start a floppy disk-based computer (a later step will explain how this diskette is created). For your convenience, the ENG-1 WATFOR-77 diskette No.1 is a boot diskette. Place WATFOR-77 diskette No.1 in the uppermost drive (if the floppy drives are placed vertically) or in the leftmost drive (if the drives are arranged horizontally). This is the "A" drive, from which the computer is usually started. The other floppy drive is called the "B" drive. Place WATFOR-77 diskette No. 2 in the "B" drive. If there is a hard disk present, it is designated the "C" drive, whether or not there is a "B" drive present. The floppy diskette is inserted into the drive with the label side uppermost, and with the label on the trailing edge of the diskette (the side away from the computer). Be sure to close the "door" on each floppy disk drive. Turn on the power to the computer and power the monitor.

In order for the WATFOR-77 software to work as intended, it is ABSOLUTELY ESSENTIAL that the computer be booted with WATFOR-77 disk 1 in drive "A" and WATFOR-77 disk 2 in drive "B".

2. After a few moments, prompts for date and time will appear. You can bypass the requested date and time input by hitting the ENTER key at each prompt. However, it is strongly recommended that you enter both date and time, since a file's directory entry includes date and time fields which are useful in determining when a file was last modified or created.

3. After dealing with the date and time input, a message will appear instructing you to type "WATFOR77" and press the RETURN key, to start the WATFOR-77 compiler. Below this message you will see the MS-DOS prompt "B:\>". This prompt tells you that "B" drive is the default drive, and the ">" symbol tells you that MS-DOS is loaded and running.

4. To change the default drive, type the letter of the desired drive, immediately followed by a colon, e.g. "A:". Hit the "RETURN" key. All MS-DOS commands are case insensitive so you

may use upper or lower case at will. You may even mix the two. Note the new prompt indicating the new default drive.

When using WATFOR-77, you will want the "B" drive to be the default disk drive, as a general rule. The reason for this is that WATFOR-77 disk 2 contains approximately 250,000 characters of free storage space (which is a fair amount of space), whereas WATFOR-77 disk 1 contains about 30,000 characters of free space (which is relatively little space). FORTRAN programs which you write during the semester can be stored on disk 2. Change the default disk drive back to "B" before continuing.

2.2 Some Keyboard Pointers

1. Viewing the keyboard from left to right, three sections can be distinguished. The first section consists of two columns of function keys. The second section consists of all alphabetic and punctuation characters, as well as the RETURN, ESCape, CTRL, SHIFT, and ALT keys. The last section comprises (among others) the numeric keypad, the INSert, DELeTe, NUMBER LOCK, PRT SC (print screen) and SCROLL LOCK/BREAK keys.

2. The 10 function keys are all programmable and WEDIT, the WATFOR-77 program editor, makes use of all of them. When MS-DOS is running, only the first five function keys have assigned functions. Of these five, the F1 and F3 keys are most useful. Both keys allow you to repeat the previous MS-DOS command. Each time you hit the F1 key, a single character from the previous command is entered at the current prompt. Hence, you can repeat a command up to a certain point, correct an error or change a parameter, and then finish the command by using either F1 or F3 keys. Pressing the F3 key causes the entire previous command to be re-entered.

3. The ESCape, CTRL, and ALT keys can take on special functions in a given applications program. In MS-DOS, the key sequence CTRL ALT DEL (all pressed simultaneously) will reboot the computer, and when a WATFOR-77 program is running, the sequence CTRL BREAK (pressed simultaneously) will terminate that program.

4. The keys of the numeric keypad are dual function keys. If the SHIFT or NUMBER LOCK keys are not active, then the function labeled on the lower part of each key can be executed, e.g. END,

HOME, PGUP (page up) and PGDN (page down) have no functions when MS-DOS is running, though they are all employed by WEDIT.

5. The backspace key will rub out the character to the left of the cursor in MS-DOS and in most applications programs. It should not be confused with the left arrow key on the numeric keypad which, in most application programs, moves the cursor left without any rub out function.

2.3 Preparing Floppy Diskettes for Use

1. New floppy diskettes must first be formatted before they can be used. You must decide when formatting a diskette whether you also want to make it a boot diskette. Diskettes used only for data storage should probably be formatted without a boot option, which will save you around 60,000 characters of storage space (this being the space required by the system boot programs). In order to format a diskette, you need WATFOR-77 Diskette No.1. The latter has, in addition to its boot capability, a utility program which formats diskettes.

2. To format an ordinary non-bootable diskette, place WATFOR-77 diskette No.1 in drive "A" and a blank diskette to be formatted in drive "B". Make sure that the default drive is "A." Type the command "FORMAT B:". You will then be prompted with a few simple instructions. Then you will be asked if you want to give the diskette a volume label. This is optional; the volume label serves as an internal diskette label and appears whenever a directory command is given.

3. To format a boot diskette, follow the instructions immediately above, using the command "FORMAT B:/S". Do not include spaces from the "B" onward. REMEMBER: When a diskette is formatted, all the contents will be LOST!

4. After you are finished formatting floppy diskettes, be sure to change the default disk drive back to "B", and be sure to insert WATFOR-77 diskette No. 2 back into drive "B" before using WATFOR-77.

2.4 The Directory

The basic command to display a listing of files from a diskette is "DIR" (for the default drive) or "DIR d:" for a non-default drive, where "d" is the desired drive. For example, "DIR" gets a directory of the default drive and "DIR A:" will get a directory of drive "A:". Note that the latter command will not change the default drive. Often, the file listing will scroll off the screen. You can compensate for this in several ways: if you hold down the "CTRL" key and press the "S" key, the display will stop scrolling - pressing any key will cause the scrolling to resume; type "DIR/P" to cause the listing to pause after each screen of new information; or type "DIR/W" to display the directory in a wide format which leaves off some information so that many more entries can be displayed.

2.5 Filenames

1. Filenames in MS-DOS consist of a primary file name up to 8 characters followed by an optional extension consisting of period and up to 3 characters. Virtually any character accessible from the keyboard is allowed in file names, except non-printing or special function characters (mathematical symbols, the asterisk, slash, backslash, colon, etc.). Note that embedded blanks are not allowed.

2. In addition to the primary file name and extension, a file may also be designated by the drive and directory in which it resides. This compound file designator is called the file specification. For example, "B:\WATFOR\MATRIX.FOR" designates the file "MATRIX.FOR" on drive "B", in subdirectory "WATFOR".

2.6 Copying Files

The basic format of the copy command is: "COPY d1:filename d2:filename", where "d1" is the source drive and "d2" is the target drive. For example, "COPY A:STUFF.FOR B:STUFF.FOR". This is the long form of the copy command, and it can be shortened in several ways. First, if the target filename is to remain the same as the source filename, it need not be repeated. The above example would then become "COPY A:STUFF.FOR B:". Note where there is a blank between characters and where there is no blank. If the default drive was drive "A", then the example could be further shortened to "COPY STUFF.FOR B:". It is recommended,

4. To move up one level in the subdirectory structure, type `:CD..`". The two dots stand for the parent directory of the current subdirectory.

5. To go directly to the root directory, type `"CD\"` (remember the backslash is the symbol for the root directory).

6. To change to a subdirectory more than 1 level below the current directory, you must indicate each intervening subdirectory, each separated by a backslash, e.g., `"CD\subdirectory1\subdirectory2"`.

7. To remove a subdirectory, all files in that subdirectory must be deleted. Then, the current directory must be at least one level above the subdirectory in question. Finally, the command `"RD subdirectoryname"` can be given.

8. When referencing a file not in the current directory, the subdirectory which contains it must be explicitly specified, e.g., `"COPY C:\subdir1\subdir2\filename A:"`. The `COPY`, `DELeTe`, `REName`, and `TYPE` commands (to be treated in the next section) all come under this requirement.

2.9 Other Important Commands

1. *Deleting files:* `"DEL filename"` will remove a file from a diskette, e.g., `"DEL C:STUFF.BAS"` will remove the file `"STUFF.BAS"` from the `"C"` drive. The command `"DEL B:*.*)" will remove all files from the current directory of drive "B".`

2. *Renaming files:* `"REN oldfilename newfilename"` will change the indicated filename. If a file with `"newfilename"` already exists, the command will be aborted with an error message.

3. *Printing files:* A "quick and dirty" method to print files is the command `"COPY filename LPT1"`, where `LPT1` is a parallel printer on port #1. Another way to print a file will be to issue the command: `COPY filename PRN"`. If multiple files need to be printed, or if line spacing is important, use `PRINT` command (you will need the MS-DOS floppy disk-1 in drive `"A:"`) `"PRINT filename1 filename2 filenameN"`, where `N` may be at most 32. Many options, like page length, auto form feed, etc., can be specified (see MS-DOS Reference Manual, pp. 11.270-11.284).

however, to fully specify source and target drives until you become experienced with this command. Otherwise, there is a potential to accidentally reverse the intended direction of the copying, i.e., possibly copying an out-of-date file on top of a new file.

2.7 Wildcard Characters

1. MS-DOS employs two wildcard characters, the "*" and the "?". The latter symbol represents any single character. So, for example, the command "DIR ?BC" will list any files on the default drive which have file names of three characters, the first character being anything and the remaining characters being "BC". The "*" stands for any number of characters up to eight. For example, "COPY ABC.* C:" will copy all files on the default drive which have a primary file name "ABC" and any characters in their extension. A common use of the "*" wildcard is the sequence " *.*", e.g., "COPY *.* A:", which will copy every file in the current directory of the default drive into the current directory of drive "A". "COPY *.FOR A:" IS AN EASY WAY TO BACK UP FORTRAN PROGRAMS. ALWAYS BACK UP YOUR WORK.

2.8 Handling Subdirectories

1. MS-DOS allows the use of subdirectories on both hard disk and floppy diskette. The root directory is symbolized by the backslash "\". This directory is created when a diskette is formatted.

2. If we give the command "MD directoryname", a subdirectory will be created one level below the current directory. The directory name may be up to 8 characters long. To change the current directory to this new subdirectory, type "CD directoryname". If we give another "MD" command in this directory, we will create a new subdirectory two levels below the root directory.

3. The command "CD" by itself will show the current directory. If the current directory is the root directory, then a backslash will be the response to this command. Note that the current directory is automatically displayed as part of the MS-DOS prompt.

4. *Viewing text files:* "TYPE filename" will display the contents of a text file on the screen. This is useful for a fast look at a FORTRAN source code file. Wildcard characters ("*" or "?") are not allowed with this command.

3. WATFOR-77 ON THE MICROCOMPUTER

3.1 General Considerations

1. There are two versions of the WATFOR-77 compiler: WATFOR77.EXE and WATFOR87.EXE. The latter version requires the presence of an Intel 8087 numeric coprocessor, which is used to perform both integer and floating point arithmetic. The former version uses software routines to implement these arithmetic functions. WATFOR77.EXE may be run with or without the numeric coprocessor; however, this chip (if it was present) would not be used by WATFOR77.EXE.

2. All of the WATFOR-77 software is under a campus site license. Accordingly, it may be copied by Lehigh University faculty, staff, and students for use on campus only.

3. A two disk set containing the essential components of WATFOR-77 has been prepared to meet the needs of Engineering-1. The first disk contains the WATFOR-77 compiler, WATCOM editor (WEDIT), and associated files. This disk is bootable. The second disk contains the WATFOR-77 and GKS (Graphical Kernel System) library files, as well as some utility programs which allow graphics output to be printed on a dot-matrix or laserjet printer. The autoexec file on the first diskette tells the user how to start WATFOR-77.

5. A version of WATFOR-77 for microcomputers with hard disks is also available. This three disk set contains detailed on-line installation instructions and an automatic installation utility to help the user install WATFOR-77 onto a hard disk system. This set of disks is available for copying at the following public microcomputing sites: the central site user's area, the Fairchild-Martindale campus library, the Media Center, the Linderman Library, and the Educational Technology Center. (For assistance with installing WATFOR-77 onto a hard disk system, contact User Services, room 194 of the Fairchild-Martindale Computing Center, building 8B, phone 83990.) Also available at these locations are two documents, written by the Computing Center, entitled "Installation and Use of WATFOR-77" and "Description of WATFOR-77 Files." Included in the first of these documents is a complete description of the various versions of WATFOR-77 available at Lehigh for faculty, staff, and student use.

5. WATFOR-77 includes a library of graphics subroutines, namely WATCOM GKS. These subroutines adhere to the Graphical Kernel System (GKS) Standard established by the International Standards Organization (ISO). WATCOM GKS allows you to write graphics programs that are device independent; i.e. are compatible with a large variety of hardware.

7. Documentation for WATFOR-77 from WATCOM publications includes:

* WATFOR-77 User's Guide for the IBM PC with DOS (Third Edition), and Supplement. This book covers the basic operation of WATFOR-77, file and device handling, data types supported, compiler options, the WATFOR-77 debugger and special configuration options.

* WATFOR-77 Language Reference Manual (Second Edition). This book gives a detailed description of WATFOR-77 implementation of FORTRAN. It is organized by topics. However, one chapter, "FORTRAN Statements," is organized alphabetically.

* WATCOM Editor User's Guide for the IBM PC with DOS, (Third Edition or Second Edition plus Supplement). This book is a complete description of the general purpose editor that accompanies all WATCOM language products. It consists of a tutorial guide, followed by a reference section.

* WATCOM GKS Graphics Tutorial and Reference Manual (Second Edition), and Addendum. This book combines a tutorial on WATCOM GKS with complete description of all graphics routines.

3.2 FORTRAN Programming and WATFOR-77

A few comments are in order concerning computer programming in general and the operation of WATFOR-77 in particular.

1. The level at which a computer functions internally is quite primitive in comparison with the level at which human beings think about and solve problems. Forcing people to work at the level of the machine would generally interfere with rather than aide the problem-solving process. In order to bridge the gap be-

tween the level at which people think about problems and the primitive level at which the machine operates, various software tools have been developed, among them "high level languages" such as FORTRAN. A language such as FORTRAN allows one to express the method to be used to solve a problem (i.e., the "algorithm") in a relatively abstract fashion, using statements which resemble a mixture of natural language (e.g., English) and mathematical notation. Given such a language for describing algorithms, two problems remain: one needs a way to enter the description of the algorithm (e.g., FORTRAN program) into the computer, and one needs to "translate" statements written in the high-level language into the more primitive statements which the machine is able to carry out (i.e., into "machine instructions"). Entering the program is accomplished by a piece of software called an "editor," and the translation of the program into machine instructions is accomplished by a piece of software called a "compiler." WATFOR-77 includes both a general-purpose text editor (WEDIT) and a FORTRAN compiler. (Note that the term "FORTRAN" is an acronym for "FORMula TRANslator" -- how appropriate!)

2. At Lehigh, WATFOR-77 is set up so that the editor and compiler function as a single entity rather than as two separate pieces of software; this makes for a more convenient environment in which to write, test, and debug FORTRAN programs. Entering "WATFOR77" at the DOS prompt will cause both the FORTRAN compiler and text editor to be loaded into the computer's memory and made available for use. FORTRAN programs can then be entered and/or modified using the text editor; once a program has been entered, it can be made to execute using the "RUN" command (to be described later).

3. Getting a FORTRAN program to run actually involves three separate steps. The program must first be compiled into machine instructions. If the program makes use of any functions or subroutines contained in FORTRAN "libraries" (for example, GKS graphics subroutines), these library routines must be "linked" into the set of machine instructions which resulted from compilation. This linking results in a largely self-contained list of instructions which can be carried out by the machine. When the machine carries out these instructions, the FORTRAN program is said to "execute." This compile-link-execute sequence is automatically performed by WATFOR-77 when the "RUN" command is given.

4. The WATCOM text editor, WEDIT, can be used to edit any kind of text file; the file need not be a FORTRAN program. Since WEDIT is a general purpose text editor, it can be used apart from the WATFOR-77 compiler; in order to do so, enter "WEDIT" at the DOS prompt. The WATFOR-77 compiler (which compiles, links, and executes FORTRAN programs) can be used apart from the WATCOM text editor by entering "COMPILE" at the DOS prompt (enter "COMPIL77" or "COMPIL87" on hard disk systems). Consult chapter two of the "WATFOR-77 User's Guide" for instructions on using the compiler apart from WEDIT.

NOTE THAT IN ENGINEERING-1, THE COMPILER AND EDITOR WILL BE USED TOGETHER; THUS, AS A GENERAL RULE "WATFOR77" RATHER THAN "WEDIT" OR "COMPILE" WILL BE ENTERED AT THE DOS PROMPT.

3.3 Using the WATCOM Text Editor

1. Recall that entering "WATFOR77" at the DOS prompt will load both the WATFOR-77 compiler and WATCOM text editor into the computer's memory, whereas entering "WEDIT" at the DOS prompt will load only the editor. In either case, after the editor starts you will see a "beginning of file" marker and an "end of file" marker, and the cursor will be at the bottom of the screen on the editor's command line. To edit an existing file, type "Edit filename" at the editor's command line, and press the RETURN key. Entering "WATFOR77 filename" (or "WEDIT filename") at the DOS prompt will start the editor and begin editing the specified file, thereby combining the above two steps into one.

Note that when the editor is used in conjunction with the WATFOR-77 compiler, a file extension of "FOR" is assumed if no extension is specified; thus, entering "WATFOR77 TEST" at the DOS prompt or entering "Edit TEST" at the editor's command line will result in the editing of a file named "TEST.FOR". When WEDIT is used apart from the WATFOR-77 compiler, no file extension is assumed.

2. Type "EXit" and press RETURN at the command line in order to leave the editor and return to DOS, saving any additions or modifications made to the file being edited. If the file being edited does not already have a name, a name can be specified with "EXit filename". To leave the editor and return to DOS without saving the file, use the "QUIT" command. "Put" or "Put filename"

will save the file to disk without leaving the editor. When the editor is used in conjunction with the compiler, a file extension of "FOR" is assumed if none is given when specifying the name under which the file is to be saved.

3. Enter "Help" at the command line to see a list of valid editor commands. Most editor commands can be abbreviated; for example, the "Edit" command can be shortened to "Edi", "Ed", or "E". The shortest acceptable abbreviation is indicated by capital letters, both in this document and in the list obtained with the "Help" command; thus, the "EXit" command can be abbreviated to "EX" but not to "E" -- after all, "E" is an abbreviation for "Edit".

In order to obtain help on a particular editor command, enter "Help command" at the command line; thus, "Help EXit" will provide information on the "EXit" command. When "<more>" or "<hold>" appears in the lower right corner of the screen, simply press the RETURN key in order to see more information or to continue editing, respectively.

Some of the editor's functions are performed via commands entered at the command line; other functions are performed via the computer's function keys. Press function key F10 to obtain a list of the functions which the editor assigns to each function key; then press the RETURN key to continue editing.

4. Pressing the function key F9 will move the cursor from the editor's command line to the editor's full screen area, where the contents of the file being edited are displayed; pressing F9 again will move the cursor back to the command line. Every time F9 is pressed, the cursor moves alternately between these two areas of the screen.

If the "Input" command is given at the command line, the cursor will move into the full screen area and a new, blank line will appear in the file being edited. The user can then enter text on this new line. Pressing RETURN will cause another blank line to appear below the line just entered. This mode of entering new lines into the file is referred to as "Input" mode. Pressing F9 will terminate input mode and return the cursor to the command line. Input mode is also terminated upon moving to a different line in the file via the up- or down-arrow keys. When this happens, the cursor remains in the full screen area, but pressing

RETURN no longer creates a new line; it simply moves the cursor to the beginning of the next existing line. Should input mode be terminated accidentally, simply press F9 to return to the command line, and give the "Input" command again.

When the cursor is in the full screen area, the line on which the cursor is positioned is referred to as the "current line." When the cursor is on the command line, the current line is highlighted. Although input mode is well suited for entering a group of new lines into a file, a single new line is more easily entered by pressing the function key F5. Pressing F5 inserts one new, blank line after the current line and positions the cursor on this new line, regardless of the cursor's original position.

5. Note that when editing an existing file, the editor works with a copy of the file rather than the original. This copy is stored in an area of the computer's memory which is referred to as an "editor workspace." When the file is saved to disk via the "EXit" or "Put" commands, the contents of this workspace replace the version of the file stored on disk; thus, the editor does not alter the disk version of the file being edited until the file is actually saved.

It is sometimes useful to edit several different files simultaneously; the WATCOM editor has this capability. Simultaneous editing is accomplished by storing a copy of each file in a different workspace; however, only one workspace is displayed at one time. The workspace whose contents are currently displayed in the editor's full screen area is called the "current workspace"; if no more than one file is edited at a time, the phrase "current workspace" necessarily refers to this one file.

In order to distinguish between different workspaces, each workspace is given a name, which is usually the same as the name of the file stored in the workspace. An arbitrary name can be assigned to the current workspace with the command "Name workspacename". A list of the names of all the editor's workspaces can be obtained with the "SHow" command; the current workspace is indicated by the ">" character. The function key SHIFT/F10 (which is equivalent to entering "Edit" at the command line and pressing RETURN) will cause the workspace following the current workspace on the above list to become the current workspace; repeatedly pressing SHIFT/F10 will display each

workspace in turn in the editor's full screen area.

6. Entering "Get filename" at the command line will cause the file specified to be inserted into the file currently displayed in the editor's full screen area (i.e., into the current workspace). The specified file is inserted immediately after the current line.

Note that there are fundamental differences between the "Edit" command and the "Get" command. Suppose that during a single editing session the command "Edit filename" is given several times, each time with a different filename. Each time this command is given, a new editor workspace will be created; the workspace will be given the name of the specified file and will contain a copy of that file. Suppose, on the other hand, that during a single editing session the command "Get filename" is given several times, each time with a different filename. Each time this command is given, no new work space will be created; rather, the specified file will be inserted into the current workspace, and the name of the current workspace will remain unchanged. "Edit filename" thus creates and names a new workspace, whereas "Get filename" simply inserts a file into the current workspace.

7. Each function key is used by the editor for at least two purposes. For example, F10 provides a help screen which summarizes the uses of the function keys, whereas SHIFT/F10 (i.e., hold down the SHIFT key and press F10) is equivalent to entering "Edit" at the command line and pressing RETURN. In WATCOM's documentation, functions SHIFT/F1 through SHIFT/F10 are referred to as F11 through F20; thus, Fn+10 denotes SHIFT/Fn, where n = 1, 2, ..., 10. Some of the more useful function keys are briefly described below.

F1: Page Up

F2: Page Down

F3: Line Up

F4: Line Down

F5: Line Insert (Inserts a new line immediately after the current line.)

F6: Line delete (Deletes the current line.)

F7: Select/deselect (in preparation for a cut and paste operation).

F8: Cut

F9: Toggle Full Screen Area/Command Line

F10: Function Key Help Screen

F16: Undelete Line (Brings back the last line deleted by F6.)

F18: Paste

F20: Edit

8. WEDIT automatically maintains a line numbering system. The line numbers are not displayed during editing. However, they may be easily referenced from the command line. For example, to locate line 48, and make it the current line, input "48" on the command line.

9. WEDIT uses special symbols for certain lines. The current line is referenced by a "."; the last line is referenced by a "\$". A "+" indicates the line after the current one and a "-" indicates a line before the current one. Numbers may be appended to these last two symbols; for example, "+10" refers to the tenth line after the current line.

10. Line number ranges are indicated by "i,j", where "i" and "j" may be numbers or special symbols. ".,\$" indicates the range from the current line to the end of the file. WEDIT also employs the symbol "*" to refer to all lines in the current workspace.

11. "i,j Delete" will delete all the lines in the range specified by "i,j". Omission of the range will cause the current line to be deleted. The command "*Delete" will delete all lines in the current workspace.

12. The command `"/string/` will search the current workspace for the first line containing the specified string of characters; for example, entering `"/FORTRAN Compiler/` at the command line will cause the editor to search for the first line which contains "FORTRAN Compiler". Note that the editor's string-searching is "case sensitive," meaning that whether the search string contains upper or lower case characters will effect the results of the search; thus, the command given above would not find the strings "fortran compiler" or "Fortran Compiler". The command `"/+string/` causes the editor to search for the first line after the current line which contains the specified string; the command `"/-string/` causes the editor to search backwards through the current workspace starting with the line immediately before the current line.

When used as a search string, the symbol `"/` refers to the last search string specified; thus, one can give the command `"/string/` to find the first occurrence of the specified string and then repeatedly give the command `"/+//` to find all subsequent occurrences. In this context, it is useful to know that the commands `"?"` and `"="` will recall and re-execute, respectively, the last command given at the editor's command line.

Note that if the `"/` character is itself part of the search string, it must be preceded by a `"%"`. For example, the command `"/1%/2/` will search for the first occurrence of the string `"1/2"`. The construction `"%/"` is one of several "meta-characters" employed by the WATCOM editor.

13. `"i,j Change/string1/string2/` will change the first occurrence of "string1" in the designated range to "string2". If no range is indicated, only the current line will be affected. `"ChangeN/string1/string2"` will change the "Nth" occurrence of "string1". The use of `"*"` instead of a whole number will result in all occurrences of "string1" in the designated range being changed. Lastly, `"*Change*/string1/string2"` will change every occurrence of "string1" throughout the file. For example, `",$ Change/sat/isat"` will change the first occurrence of "sat" to "isat" in the range from the current line to the end of the file. The command `",$ Change*/sat/isat"` will change every occurrence of "sat" to "isat" in the designated range.

14. "Directory" will list the files on the current drive. This command takes all the usual MS-DOS modifiers; wildcards, drive designations, and path specifications.

15. "ERase filename" deletes the selected file(s) from the disk. Wildcards, drive designations, and path specifications are allowed.

16. The "SYStem" command will temporarily leave the editor and return to MS-DOS; however, the WATCOM editor, WATFOR-77 compiler (where applicable), and all editor workspaces are retained in the computer's memory. Any number of DOS commands may then be entered at the DOS prompt. The DOS command "EXIT" will return the user to the editor and (where applicable) the compiler, with all of the editor's workspaces intact.

Note that if only one DOS command is to be executed, it is more convenient to enter "SYStem doscommand" at the editor's command line. After the specified DOS command is executed, press the RETURN key to continue editing.

17. It is possible (to some extent) to tailor the WATCOM editor to suit the user's personal needs. Whenever the editor is started up in conjunction with the compiler (via entering "WATFOR77" at the DOS prompt), any editor commands contained in the file "PROFILE.FOR" are automatically performed, just as though they had been entered at the editor's command line. Whenever the editor is started up apart from the compiler (via entering "WEDIT" at the DOS prompt), editor commands contained in the file "PROFILE.EDI" are automatically performed. Any commands which the user gives at the beginning of every editing session can be put in the appropriate profile file, and these commands will be executed automatically. For example, the command "TABset i, j, k, ...", which sets the editor's tab-stops to the specified columns, has already been put into both profile files; the tabs have been set to columns 7, 12, 17, 22, ..., 72. The command "BELL 72" has also been put into both profile files; this command causes the computer's bell to ring whenever the cursor moves from column 72 to column 73. These commands have been used to set up the editor for entering FORTRAN programs (FORTRAN statements must be contained in columns 7 - 72).

It is also possible for the user to assign functions to the function keys with the command "PF n text", where "n" is a whole number from 1 to 10, and "text" is the text to be produced whenever the user enters ALT/Fn; for example, giving the command "PF 10 HELP<CR>" will cause the text "HELP" followed by a carriage return (just as if the user had pressed the RETURN key) to be inserted after the cursor whenever the user holds down the ALT key and presses function key F10. After giving this command, pressing ALT/F10 at the command line will produce the same results as typing "HELP" and pressing the RETURN key. If the command "PF 10 HELP<CR>" is placed in the editor profile files, then ALT/F10 will automatically be defined to perform as described above.

3.4 Using WATFOR-77 Compiler

1. Suppose that "WATFOR77" has been entered at the DOS prompt, and the WATCOM editor and WATFOR-77 compiler have been loaded into the computer's memory. Suppose also that a FORTRAN program has been newly entered during the current editing session; the workspace containing the program (presumably the current workspace) will not yet have a name. A name can be given by entering "Name workspacename" at the editor's command line. After naming the workspace, it is wise to save the workspace to disk with the "Put" command. (Note that the "Put" command does not name the workspace.)

Once the current workspace has been named and saved to disk, the FORTRAN program contained in it can be compiled, linked, and executed by entering "RUN" at the editor's command line. (Recall that compiling, linking, and executing the FORTRAN program are accomplished by the WATFOR-77 compiler.) If the program contains no syntax errors (i.e., if there are no mistakes in FORTRAN "grammar"), the program will be executed immediately after it has been compiled and linked. Any output which is written to Unit 6 (the default unit number for output) will appear on the screen, and any input which is read from Unit 5 (the default unit number for input) may be entered at the keyboard. When the program has finished executing, "<hold>" will appear in the lower right corner of the screen; press the RETURN key to return to the editor and resume editing the FORTRAN program.

Note that if one attempts to "RUN" a program contained in an unnamed workspace, the message "no file name" will appear in the

line above the command line, and the program will not be compiled. This message indicates that a workspace must be named before the program contained in it can be run.

In addition to compiling, linking, and executing a FORTRAN program, the "RUN" command creates what is called a "listing file". This file is stored in the current directory of the default disk drive. It has the same primary file name as the workspace, but has an extension of "LST"; for example, if the workspace is named "TEST.FOR", the "RUN" command will create a listing file named "TEST.LST". The listing file contains a copy of the original FORTRAN program, as well as some additional information which is generated during the compilation, linking, and execution of the program. All non-blank, non-comment lines in the original program are numbered; these line numbers appear to the left of each effected line in listing file. At the end of the listing file are some statistics concerning how much time was needed to compile and execute the program, how much memory was used, etc.

There are at least two ways to examine this listing file while using the WATFOR-77 compiler/editor. If the workspace containing the FORTRAN program is named "name.FOR", the command "SYSTem TYPE name.LST" will type the listing file, just as if the "TYPE" command had been given at the DOS prompt; press RETURN in order to resume editing. Here is a more advanced way to examine the listing file: enter "Edit name.LST" at the editor's command line; this will create a new editor workspace containing the listing file, which can then be easily examined using the PgUp, PgDn, up-arrow, and down-arrow keys. Note that the original FORTRAN program is still in the computer's memory in a separate workspace. Repeatedly pressing SHIFT/F10 will alternately display the program file and the listing file in the editor's full screen area. Be sure to "QUIT" from the workspace containing the listing file before running the program again; otherwise, the next "Edit name.LST" command will display an outdated copy of the listing file.

2. If the FORTRAN program contains syntax errors, error messages will be sent to the screen during compilation, and the program will not be executed. The compiler tries to the best of its ability to locate and describe all syntax errors contained in the program; however, this requires a level of intelligence which

most compilers are not yet able to muster. For this reason, compiler error messages most often give one a "feel" for where the true problem lies, rather than a precise description of the nature and location of the problem.

Note that the editor numbers every line in the original program, whereas the listing file numbers only lines which contain FORTRAN statements; thus, the editor's line numbers are different from the line numbers which appear in the listing file! The WATFOR-77 error messages which are sent to the screen may refer to line "n", where "n" is some whole number; this line number refers to the editor line number in the original program and not the listing file line number. Recall that entering "n" at the editor command line will make line "n" become the current line; this provides a convenient way to examine the lines specified in the compiler's error messages.

When compiling a program which contains syntax errors, the compiler also sends error messages to the listing file. The compiler tries to place these error messages immediately below the line which it believes to be the cause of the problem, and points (with a "\$") to what it believes to be the offending portion of the line. This feature makes the listing file particularly valuable when trying to debug longer programs containing many syntax errors.

3. Entering "RUN" at the editor's command line is the most basic way to compile, link, and execute the FORTRAN program contained in the current workspace. For the convenience of the user, several other ways to run a program have been provided with the WATFOR-77 software distributed at Lehigh. WHEN THE CURSOR IS ON THE COMMAND LINE, the function keys ALT/F1, ALT/F2, and ALT/F3 will also compile, link, and execute the program in the current workspace, with the following differences. ALT/F1 causes Unit 6 output to appear on the screen, as does "RUN", but creates no listing file; thus, ALT/F1 provides a slightly faster way to run, a way which does not clutter up the disk with "LST" files. ALT/F2 causes Unit 6 output to appear on the screen and does generate a listing, but sends the listing to the screen rather than

to disk; thus, ALT/F2 provides a convenient way to examine the listing produced by the compiler (and also does not clutter up the disk). ALT/F3 sends both Unit 6 output and the listing to the listing file rather than to the screen; ALT/F3 is best suited for creating a file which can be printed and turned in to the instructor for an assignment.

These function key definitions are automatically made whenever the editor/compiler starts up (the file "PROFILE.FOR" contains the appropriate "PF" commands). The function key F10 displays the functions assigned to user-programmable function keys as well as the standard editor function keys.

4. SOME HELPFUL HINTS FOR ENGINEERING-1

The microcomputer user must be aware of a number of command statements and what they do. These are listed below according to operational categories.

4.1 MS-DOS Commands

These commands may be given when the MS-DOS prompt "B:\>" is on the screen.

DIR	lists the names of the files on the (directory) on the disk in the default drive (B).
DIR A: /P	lists the names of the files on the disk in drive A, and pauses after every screen of output.
DIR B:	lists the names of the files on the disk in drive B.
TYPE SAMPLE1.LST	lists the contents of the file names "SAMPLE.LST" on the disk in the default drive.
DEL SAMPLE1.LST	erases the file names "SAMPLE1.LST" from the disk in the default drive.
REN DATA CIVIL.DTA	renames the file named "DATA" on the disk in the default drive to "CIVIL.DTA" (you may have little use for this command)
FORMAT B:	formats the disk in your drive B. You <u>must</u> format a new disk before using it for the first time. (IMPORTANT: An ENG-1 WATFOR-77 diskette No.1 must be in drive A for executing this command.)

COPY A:PROGA.FOR B:PROGB.FOR copies the file called "PROGA.FOR" on the disk in drive A as a file called "PROGB.FOR" to the disk in drive B.

COPY B:PROGR2.FOR LPT1:
COPY B:PROGR2.FOR PRN gets the file called "PROG2.FOR" on the disk in drive B and prints the contents on a dot-matrix printer. (You realize, of course, that the microcomputer used to do this must be physically wired to the printer.)

To copy the entire contents of one diskette onto another diskette, you can go through the following steps.

- 1) Get an MS-DOS (#1) disk from the consultant on duty at the site.
- 2) Place it in drive A and boot the system.
- 3) Type "DISKCOPY" and remove the MS-DOS disk from drive A.
- 4) Place the source diskette in drive A and the target diskette in drive B and depress the return key. (The diskette in drive B will first be formatted and then all of the files on the disk in drive A will be copied onto the disk in drive B. REMEMBER: When a DISKCOPY command is invoked as described above, all the contents, if any, of the diskette in drive B will be lost.

4.2 Additional Scenario

The following situations apply when the Engineering-1 WATFOR-77 Diskette No.1 is located in the "A" drive, and WATFOR-77 Diskette No.2 is in "B" drive.

Situation: Have a program called TEST1.FOR with
OPEN(6,FILE='RESULT.LST')

and either

```
WRITE(*,16)
```

or

```
WRITE(6,16)
```

Then if you do

- i) ALT/F2 at the editor's command line
Compilation will appear on the screen, but output will be in a file called RESULT.LST.

- ii) ALT/F3 at the editor's command line
Nothing will appear on the screen.
Compilation will be in a file called TEST1.LST, and output will be in the file called RESULT.LST.i