



MTC QUE CARD™

Quick User Education

NEWDOS/80

by Apparat, Inc.

This card is intended as a quick reference guide for the documentation provided by Apparat for the NEWDOS/80 Disk Operating System. The user of this guide is assumed to be familiar with that publication and is encouraged to look there for more complete explanations and detailed information. This guide is produced under agreement with Apparat, Inc. and is proprietary to MTC.

IMPORTANT NOTICE DISCLAIMER OF EXPRESS AND IMPLIED WARRANTIES

EXPRESS WARRANTY AND IMPLIED WARRANTIES, INCLUDING WARRANTIES OF MERCHANTABILITY AND FITNESS FOR PARTICULAR USE INTENDED ARE HEREBY DISCLAIMED BY METATECHNOLOGIES CORPORATION, INC. (MTC).

MTC shall have no liability for consequential damages including, but not limited to, loss of use, business, or profits alleged to have been caused by the MTC QUE Card.

QUE CARD is a TM of Metatechnologies Corporation, Inc.
Copyright © 1980 Metatechnologies Corporation, Inc.
Radio Shack ® and TRSDOS are trademarks of Tandy Corp.

EXPLANATION OF TERMS:

Most terms used in this guide are explained where they are used. Some common ones include:

file—The name of a disk file, its password and the drive it is on, in the form *name/ext.ipwd.dd*, where:

The file *name* is up to 8 characters, the first must be a letter; *ext* is the name extension, up to 3 characters, the first must be a letter; (optional)

ipwd is the file password, up to 8 characters, the first must be a letter; (optional)

dd is the drive number where the file is located, from 0 to 3. (optional)

dname—The name of a disk, up to 8 characters, the first must be a letter

dpwd—A disk's master password, same form as a file password

yn—Yes or no specification. The words (Y), Y, (ON) or ON generally mean "enable" or "turn on." The words (N), N (OFF) or OFF generally mean "disable" or "turn off." A null (no entry at all) usually implies "ON". A *yn* specification is used to turn on or off various DOS features.

DOS

COMMANDS

- Shaded commands are not executable under MINI-DOS
- All commas shown can be replaced by spaces
- The word "TO" is always optional
- The colon (:) before a drive number is sometimes optional
- NEWDOS/80 comes configured with password protection DISABLED. Therefore, whenever the use of passwords (*ipwd* or *dpwd*) is mentioned in a DOS command, it is NOT needed.

APPEND, *file1*, TO, *file2*

Add the contents of *file1* to the end of *file2*.
APPEND NEWPART/DAT:1 TO OLDPART/DAT:2

ATTRIB, *file1*, options

Change the attributes of *file1*. Options are: (at least 1 is needed)

- INV—Make the file invisible to a normal DIR command
- VIS—Make the file visible.
- PROT = *xxx*—Assign protection level *xxx* to the file. Levels are:
 - LOCK—no access
 - EXEC—execute only
 - READ—execute or read
 - WRITE—executed, read or write
 - RENAME or NAME—execute, read, write, rename
 - KILL—execute, read, write, rename, KILL
 - FULL—anything
- ACC = *ipwd*—Assign *ipwd* as file's access password.
- UPD = *ipwd*—Assign *ipwd* as the file's update password.

ATTRIB SECRET: 1 INV PROT = EXEC UPD = PASS

AUTO, *doscmd*

Execute the DOS command *doscmd* when the system is re-booted. Holding down the <ENTER> key when re-booting will prevent execution of the automatic command. AUTO alone disables any previous AUTO command.

AUTO BASIC

BACKUP

Not used with NEWDOS/80. Replaced by third form of COPY command.

BASIC, *bufnum*, *memsize*, command

Enter Disk BASIC. Allow *bufnum* I/O buffers (HOW MANY FILES?) and set the high memory limit to *memsize* (MEMORY SIZE?). *command* can be any valid BASIC command or statement, which will be executed immediately. All parameters are optional (*bufnum* defaults to 3 and *memsize* to all available memory).

```
BASIC  
BASIC,4,50000,RUN"MYPRG/BAS"  
BASIC,32767
```

BASIC2

Enter Level-II BASIC. DOS is no longer active.

BASIC2

BOOT

Re-boot the system.

BOOT

BREAK, *yn*

Enable or disable the BREAK key.

BREAK (OFF)

CHAIN, *file1*, section

Begin CHAIN mode, using the contents of *file1*. If the optional *section* is specified, CHAINING will begin with the portion of the file named *section*. If no extension is specified in *file1*, JCL is assumed. (See also the description of CHAIN files under SPECIAL DOS FUNCTIONS.)

```
CHAIN COMFIL/JCL:1 PARTA
```

CLOCK, *yn*

Turn on or off the on-screen clock display.

CLOCK Y

COPY, *file1*, TO, *file2*

Copy the contents of *file1* into *file2*. If *file1* was created by a non-NEWDOS/80 system, put a \$ (no space following) before *file1*.

```
COPY BIGFILE/DAT:1 TO BACKUP/DAT:2
```

COPY, :*dd*, *file1*, TO, *file2*

Single-drive copy: Copy the contents of *file1* on drive *dd* to *file2* on another disk, but using the same disk drive. If *file1* was created by a non-NEWDOS/80 system, put a \$ before *file1*.

```
COPY :0 MYFILE/BAS TO NEWFILE/BAS
```

COPY, :*dd1* = *tc*, TO, :*dd2* = *tc*, date, options

Copy the entire contents of the disk on drive *dd1* to the disk on drive *dd2*. If either disk has an unusual number of tracks, = *tc* is used to specify that number. (*dd1* can be the same as *dd2*.) Options are:

- Y—The user doesn't care what was on the destination disk.
- N—The destination disk has been bulk-erased.
- NDMW—No Disk Mount Waits: Both disks must already be mounted; there will be no prompts to put them in.
- NFMT—No ForMaT: Do not format the destination disk (formatted already.)
- SPW = *dpwd*—The source disk's master password must be *dpwd* or COPY will halt.
- NDPW = *dpwd*—Make the destination disk's master password *dpwd*.
- DDND—Display Destination disk's old Name and Date on the screen and ask the operator whether or not to continue.
- ODN = *dname*—If the destination disk's name is not *dname*, the operator is asked whether or not to continue.

KDN—Keep Destination disk's old Name.
KDD—Keep Destination disk's old Date.
NDN = *dname*—Make the destination disk's name *dname*.
USD—Use Source disk's Date as the date for the destination disk.
SN = *dname*—If the source disk's name is not *dname*, the operator is asked whether or not to continue.
BDU—Bypass Directory Update: Do not specially modify the directory on the destination disk. (rarely used)
UBB—Use Big Buffer: Copy disks with as large a buffer as possible.
CBF—Copy By File: Copy only those files listed in the source disk's directory.

Additional options available with CBF are:

CFWO—Check Files With Operator: The operator is asked whether or not to copy each file.
ODPW = *dpwd*—If the destination disk's password is not *dpwd*, the operator is asked whether or not to continue.
DDST = *tc*—Start the destination disk's directory at track *tc*.
DDGA = *gn*—Assign *gn* granules to the destination disk's directory.

COPY 0 TO 0 06/01/80 USD KDN ODN = NEATO SN = GOSH
COPY 2 TO 3 = 77 06/05/81 NDMW CBF CFWO DDST = 29 DDGA = 4
COPY 0 1 09/25/80

DATE, *date*

Set computer's date to *date*.
DATE 09/05/08

DEBUG, *yn*

Enable or disable DEBUG for use with the BREAK key. Invocation using 1-2-3 is unaffected (see SPECIAL KEYBOARD COMMANDS).
DEBUG OFF

DEVICE

Not supported under NEWDOS/80.

DIR :*dd*, *options*

Display the directory of the disk in drive *dd*. If *dd* is not specified, drive 0 is assumed. Options are:

S—display system files
I—display files with the INV attribute
A—display each file's space allocation
P—send the listing to the printer

DIR 1 S I A

DUMP, *file*, *start-addr*, *end-addr*, *entry-addr*

Take the contents of memory locations *start-addr* through *end-addr* and place them in a disk file named *file*. When that file is later executed, execution will begin at *entry-addr* (if *entry-addr* is not specified, then *start-addr* will be used).
DUMP PROG/CMD 2 5200H 9ABCH 54EDH

FORMAT, *dd* = *tc*, *dname*, *date*, *dpwd*, *options*

Format disk on drive *dd* with name *dname*, date *date* and password *dpwd*. The optional track count *tc* specifies the number of tracks to be formatted, if different than normal. Options are:

N, Y, DDND, NDMW, ODN = *dname*, KDN, DDST = *tc*, DDGA = *gn*
(See COPY command for explanations.)
FORMAT 1 MTC 09/04/80 MTC Y NDMW

FREE

Display number of free granules and available directory entries on all drives.
FREE

HIMEM, *addr*

Set DOS's high memory boundary to *addr*. Used by COPY, BASIC, EDTASM, DISASSEM and LMOFFSET as their default upper memory limit.
HIMEM FFFFF

JKL

Copy contents of screen to printer.
JKL

KILL, *file1*

Delete *file1* and free the disk space it occupies.
KILL BADFILE/BAS:3

LIB

List DOS library commands.
LIB

LIST, *file1*, *startline*

List *file1* on the display *startline* optionally specifies the first text line to be listed.
LIST MYPRG/ASC 25

LOAD, *file1*

Load the machine-language disk file *file1* into memory, but do not execute it.
LOAD INLINE/OB1:1

MDBORT

Terminate MINI-DOS and go to DOS READY.
MDBORT

MDCOPY, *file1*, TO, *file2*

Copy a file under MINI-DOS. Similar to first COPY format.
MDCOPY FILEA TO FILEB

MDRET

Exit MINI-DOS and resume execution of main program (BASIC program, assembly program, etc.)
MDRET

PDRIVE, *dpwd*:*dd1*;*dd2*, *options*

Tell the NEWDOS/80 system disk mounted on drive *dd1* what kind of drive *dd2* is. Options are:

TI = *type*—type of interface used: A = TRS-80 Model I Expansion Interface; B = OMIKRON 8" Interface.
TD = *type*—type of disk drive: A = 5-1/4" single-sided, single-density; B = 8" single-sided, single-density.
DTC = *tc*—number of tracks on a disk (1-96).
DDST = *tc*—disk directory starts at track *tc*.
DDGA = *gn*—disk directory is allocated *gn* granules.

NOTE: The system must be re-booted in order for the specifications to take effect.

PDRIVE :0 :1 TI = A TD = A DTC = 35 DDST = 17 DDGA = 2 (the system comes configured this way)

PROT, *dpwd*:*dd*, *options*

Alter master information of disk on drive *dd*. Options are (at least 1 must be specified):

NAME = *dname*—Give disk name *dname*.
DATE = *date*—Give disk date *date*.
LOCK—Give all files on the disk password *dpwd* as their access and update passwords.
UNLOCK—Give all files on the disk blanks as their access and update passwords.
PW = *dpwd*—Give the disk master password *dpwd*.
PROT :0 NAME = MTC UNLOCK

PURGE, *dpwd*:*dd*

Selectively KILL all files from the disk on drive *dd*. The operator is asked whether or not to delete each file on the disk (except BOOT/SYS and DIR/SYS).

PURGE :1

RENAME, *file1*, TO, *file2*

Change the name of *file1* to *file2*.
RENAME OLDFILE TO NEWFILE

SYSTEM, *dpwd*:*dd*, *options*

Change the system options of the NEWDOS/80 system disk on drive *dd*. Options are: (bold items indicate the way the system originally is configured)

AA = *aa*—Passwords: Y = enabled, N = disabled.
AB = *ab*—Run-only mode: Y = enabled, N = disabled.
AC = *ac*—NEWDOS/80 keyboard debounce routine: Y = active, N = inactive.
AD = *ad*—'J-K-L' screen copy: Y = enabled, N = disabled.
AE = *ae*—'1-2-3' DEBUG: Y = enabled, N = disabled.
AF = *af*—'D-F-G' MINI-DOS: Y = enabled, N = disabled.
AG = *ag*—BREAK key: Y = enabled, N = disabled.
AH = *ah*—Clock lost time fix: Y = 12ms lost, N = 70ms lost.
AI = *ai*—Lowercase mod: Y = present, N = not present.
AJ = *aj*—Keyboard Intercept routine: Y = active, N = inactive.
AK = *ak*—'J-K-L' graphics code handling: Y = leave alone, N = convert to ""

AL = *al*—Number of drives (1-4): 4

AM = *am*—Number of disk read retries: 10

AN = *an*—Default drive # for DIR command: 0

AO = *ao*—Drive # of first drive searched for new file (0-3): 0

AP = *ap*—Default HIMEM address: 0

AQ = *aq*—CLEAR key: Y = enabled, N = disabled.

AR = *ar*—Full disk copy w/o passwords: Y = allowed, N = disallowed.

AS = *as*—Inputted lowercase chars: Y = convert to uppercase, N = leave alone.

AT = *at*—Single character input from CHAIN files: Y = available, N = entire lines only.

NOTE: The system must be re-booted in order for the changes to take effect.
SYSTEM :0 AA = N AL = 3

TIME, *time*

Set computer's clock to *time*.
TIME 14:25:17

TRACE, *yn*

Turn on or off the screen display of the Z-80 PC Register.
TRACE N

VERIFY, *yn*

Turn on or off VERIFY mode for all disk writes.
VERIFY (Y)

DOS AND BASIC

SPECIAL DOS FUNCTIONS

MINI-DOS

There are always situations where it would be convenient to be able to stop execution of a program, execute a DOS library command, and resume execution—at a moment's notice—without disturbing the program or variables. MINI-DOS offers just such a facility. When the D, F and G keys are pressed simultaneously, MINI-DOS is entered. This can occur during execution of a DOS command, in the middle of a BASIC program, even (under most circumstances) in an assembly-code program. The commands executable under MINI-DOS are subset of the normal DOS commands (those not shaded in the DOS command summary). Typing MDRET returns you to your main program; MDBORT takes you to DOS READY.

CHAIN FILES

NEWDOS/80 allows the use of CHAIN files, which are sequential files containing a series of statements, commands, replies, etc. which are used instead of input from the keyboard. When the DOS command CHAIN is entered, the file specified (default extension: /JCL) is opened and records from it are substituted for keyboard input until the end of that file is reached. Each CHAIN file record is terminated by a carriage return (ODH). Special functions are performed when a CHAIN file record begins with one of the following special characters:

- 80H—What follows is a section identifier (for use with the section parameter of the DOS CHAIN command)
- 81H—What follows is to be displayed; the system then waits for the <ENTER> key to be pressed (pause function)
- 82H—What follows is a comment (ignore it)
- 83H—What follows is to be displayed, but not used as substitute input

RUN-ONLY MODE

For the purposes of program security, NEWDOS/80 can be configured to operate in a "Run-Only" mode. With system option AB=Y, the system will boot up with the BREAK key disabled. The operator is able to type in data, but cannot enter any direct commands to either BASIC or DOS (an illegal direct error occurs in BASIC, DOS asks that the system be reset). It is therefore assumed that there is an AUTO command active which will invoke the necessary commands (including CHAIN files) to execute the desired program(s).

DEBUG FACILITY

A DEBUG facility is available as an aid to machine-language programmers. DEBUG can be entered by pressing the 1, 2 and 3 keys simultaneously, or after enabling DEBUG with the appropriate DOS command, pressing the BREAK key, beginning execution of a program, or executing an RST 30H or CALL 440DH instruction.

DEBUG, when entered, saves all CPU registers and displays the contents of memory where it was invoked, in one of two modes. DEBUG commands (all followed by <ENTER>) include:

- X—Change to X display mode (display contents of all registers plus 3 separate 64 byte memory areas).
- S—Change to S display mode (display one 256 byte memory area).
- Daddr—Begin memory display at address addr.
- ;- Advance display 256 bytes (S mode) or 64 bytes (X mode).
- - Decrement display.
- The commands D, ; or - may be preceded by 2 or 3 to indicate the middle or lower areas of the X display.
- Maddr—Change to S mode and begin modifying displayed memory (<ENTER> terminates modifications).
- Faddr,byte1,byte2—Find the specified series of bytes (up to 4) after address addr and begin X display at that location - in rows 20H.
- I—Single-step through the program.
- C—Same as I, but CALLS execute entire routines.
- Rreg val—Put val into register pair reg.
- Ldn,tn,sn—Get sector sn, track tn from disk dn and put it in the DOS buffer at 4200H.
- Wdn,tn,sn—Write DOS buffer to disk dn, track tn, sector sn.
- Gaddr1,addr2,addr3—Restore registers and continue execution at addr1; optionally set breakpoints at addr2,addr3.

SPECIAL KEYBOARD COMMANDS

While in DOS or BASIC, simultaneously pressing any of the following groups of three keys will perform the indicated special functions:

- [J][K][L] Copy the contents of the video display to the line printer (system will halt until reset if none is connected)
- [1][2][3] Enter DEBUG
- [D][F][G] Enter MINI-DOS

BASIC

IMMEDIATE MODE EXTENSIONS

When the word "READY" and a prompt (">") appear on the screen, the computer is in "immediate mode", and NEWDOS/80 allows several new commands to be executed:

LISTING/EDITING COMMAND KEYS

When in immediate mode, certain keys have special functions. They are:

- [SR][↑][=][f] List first line
- [↑] List previous line
- [.] List current line
- [↓] Edit current line
- [↓] List next line
- [SR][↓][=][f] List last line

COMMAND ABBREVIATIONS

The BASIC commands LIST, EDIT, DELETE and AUTO can be abbreviated to one letter if they are the first character on a line and are followed by a period (.) or one or more line numbers. For example:

LIST50 can be entered L50

EDIT. can be entered E.

DELETE26-98 can be entered D26-98

AUTO10.5 can be entered A10.5

DI and DU COMMANDS

The commands DI (meaning Delete and Insert) and DU (meaning DUPLICATE) have been added to BASIC. They are entered:

DI line1,line2—line1 will be deleted, then inserted as line2.

DU line1,line2—line1 will be duplicated as line2, but not deleted.

NOTE: A period (.) may be substituted for line1 or line2, meaning "use the 'current' line."

DI 10,25

DU 15.

REF COMMAND

The command REF may be entered to obtain a list of the line numbers of all statements that refer to any or all variables, line numbers or constants. Forms of REF are:

- REF vn—Display on screen all references to variable vn.
- REF line—Display on screen all references to line number line or integer constant line.
- REF *—Display a complete list of all references.
- REF \$—Print a full reference list.
- REF *vn or REF *line—Display on screen a complete list of all references, starting with vn or line.
- REF \$ vn or REF \$ line—Print a complete reference list starting with vn or line.
- REF—If REF alone is entered after REF vn or line, the next text line containing vn or line will be listed. REF alone can be used repeatedly to list all lines which reference vn or line.

NOTE: Pressing <BREAK> will pause the display of a list, <ENTER> will continue the list and <UP-ARROW> will terminate it. Do not include type declarations (\$,!,%) in vn.

REF A

REF 1008

RENUM COMMAND

The RENUM command has been added to facilitate the renumbering of programs in memory. All or part of a program can be renumbered, and one section of a program can be relocated to another area of the program, so long as the section fits between existing lines or is placed at the beginning or end of the program. Forms of RENUM are:

RENUM U—Check the program in memory for errors in references. Errors indicated are:

line1/U—line1 is referenced, but does not exist

line1/X—there is a syntax error in line1

line1/S—line1 has a bad line number

RENUM newline,increment,startat,endwith—start renumbering lines numbered startat through endwith, beginning with new line number newline, and incrementing by increment. Any or all of the parameters can be omitted, by replacing them with a comma (at least 1 comma must follow RENUM). When they are omitted, the parameters are assumed to be:

newline = 10

increment = 10

startat = first line

endwith = last line

RENUM , would renumber all text lines, starting with new line 10, incrementing by 10

RENUM 5,5 would renumber all text lines, starting with new line 5, incrementing by 5

RENUM 110,1,1008,2036 would renumber lines 1008 through 2036 starting with new line 110, incrementing by 1 (if there were a line 500, then the new lines would come before it, instead of after it as they did before.)

DISK I/O EXTENSIONS

NEW FILE TYPES

NEWDOS/80 supports the sequential and random file types available with Radio Shack Disk Basic. In addition, several new file types are available:

- Fixed-item files—just like TRSDOS random files but no conversions (MK\$, CVI, etc.) are necessary. Individual data items are not marked in any way as to their type—it is up to the programmer to assign the proper sections.
- Marked-item files contain one byte in the file before each item to identify the item as a string, integer, and so on. Type mismatch errors may occur if incorrect access is attempted.

Either file type can be record-segmented (there are distinct records) or not record-segmented (breaking the file into records is the programmer's responsibility). A record can be from 1 to 4095 bytes long.

The new types available are:

- "FI"—a file of unmarked (fixed) items, without individual records (essentially a continuous stream of data)
- "FF"—a file of unmarked items, but with clearly defined records of the same length
- "MI"—a file of marked items, without individual records (cannot be updated)
- "MF"—a file of marked items in specific records of the same length
- "MU"—a file of marked items, with specific records of differing lengths (see manual)

MODIFICATIONS TO OPEN STATEMENT

The OPEN statement has been extended to include the form:

OPEN *mode,bufnum,file,type,length*

mode—Any new file type can be OPENed in any of the following modes:

- "I"—Open for input only
- "O"—Open for output only
- "E"—Open for output, starting at the end of the file
- "R"—Open for direct access; create if necessary; add to end if needed
- "D"—Open for direct access; file must exist; cannot add to end

bufnum—I/O buffer number (1-15); max. is specified when BASIC is initialized

file—Name of file

type—see discussion above

len—Length of a logical record (for type FF and MF files only)

OPEN "R",2,"NEWSTUF/DAT,3","MF",35

OPEN "E",1,"DATAFIL","MU"

ADDITIONS TO GET AND PUT STATEMENTS

The GET and PUT statements have been extended to include the forms:

GET/PUT *bufnum,position,IGELSN* and

GET/PUT *bufnum,position,,IGEL;*

bufnum—I/O buffer number (1-15); max set when BASIC is initialized

position—Position in file to get or put record.

Forms are:

- (null)—next record
- *—same record
- #—Set to REMRA - REMembered Record Address: The file is positioned to the point where the last record was read or written.
- \$—Set to REMBA - REMembered Byte Address: The file is positioned to the last place that was read or written.
- %—Pseudo FIELD function: No data is transferred; the string variables in the IGEL are merely assigned a set length (just like a FIELD statement) and LSET or RSET can then be used to move data into them.
- !*rba*—position to *rba* bytes from beginning of file
- !%—position to end of file (PUT only)
- !\$*rba*—position to *rba*, but transfer no data
- !\$%—position to EOF, but transfer no data
- !#*rba*—set file's EOF to *rba* (PUT only)
- !*rn*—logical record number for type FF and MF files only.

IGEL—Item Group Expression List—a series of variables, expressions or special specifiers from which or to which data from the file is directly transferred. IGEL lists must end with a semicolon (;). Items can include:

expression—usually a simple variable name on a GET, a variable or an expression on a PUT

(*len*)*expression*—*expression* is a string variable or expression, preceded by the number of characters (*len*) to be written from it or read into it.

(*len*)\$—Skip *len* characters.

(*len*)#—usually the same as (*len*)\$ (see manual)

IGELSN—The line number of a separate statement that contains an IGEL or the first of a series of such statements. (These statements are NOT executable!)

GET2,!512,A#, (25)B\$,C%, (10)\$,D!;
Would start reading file 2 at the 512th byte from the beginning of the file and read a value into A# (8 bytes), 25 characters into B\$, a value into C% (2 bytes), skip 10 bytes and read a value into D! (4 bytes).

LOC FUNCTION

The LOC function is available to perform the following functions:

- LOC(*bufnum*) = logical record number of last record read
- LOC(*bufnum*)\$ = -1 if the current position in the file is >= EOF, 0 if it is not
- LOC(*bufnum*)% = position of file's EOF (number of bytes from beginning of file)
- LOC(*bufnum*)! = current position in file (sometimes = position of next record—see manual)
- LOC(*bufnum*)# = REMRA

LOC(1)

LOC(1)%

DISK BASIC ERROR CODES (ERR/2)

- 50 Field Overflow
- 51 Internal Error
- 52 Bad File Number
- 53 File Not Found
- 54 Bad File Mode
- 55 File Already Open
- 57 DOS Error
- 58 File Already Exists
- 61 Disk Full
- 62 Input Past End
- 63 Bad Record Number
- 64 Bad File Name
- 65 Mode Mismatch
- 66 Direct Statement in File
- 67 Too Many Files
- 68 Disk Write Protected
- 69 File Access Denied
- 70 Sequence Number Overflow
- 71 Record Overflow
- 72 Illegal to Extend File
- 74 Previously Displayed Error
- 75 Can't Process Line 0
- 76 Bad File Type
- 77 IGEL Syntax Error
- 78 IGEL Item Syntax Error
- 79 Bad/Illegal/Missing IGEL Item Prefix
- 80 IGEL Item Requires Type
- 81 Bad Record Length
- 82 Statement Uses 2 File Areas
- 83 Bad File Positioning Parameter

ADDITIONAL UTILITY PROGRAMS ON THE NEWDOS/80 DISK

SUPERZAP/CMD—Program to examine, alter, or copy any sector on a disk.

DISASSEM/CMD—Disassembler for machine-code programs on disk or in memory.

LMOFFSET/CMD—Utility for transferring machine-code programs from tape to disk and for relocating any machine-code program to another area of memory.

DIRCHECK/CMD—Utility for testing and verifying a disk's directory.

EDTASM/CMD—The TRS-80 Z-80 Tape Editor/Assembler adapted for use with disk.

LEVEL1/CMD—A slightly modified version of Level 1 BASIC with facilities for loading and saving programs on disk.

LV1DSKSL/CMD—Interface program to allow Level 1 disk saves and loads.

LCDVR/CMD—A general purpose driver for lower-case modified machines.

ASPOOL/MAS—Automatic spooler facility (see manual).

Additional copies available from:

Meta Technologies Corporation
26111 Brush Avenue
Euclid, Ohio 44132
1-800-321-3552