RELEASE NOTES

Subject: TurboDOS 1.43+16a.00

TurboDOS 1.43+16a.00 Release Notes

Publication No. 440-0095A

06 JUN 1986

PRODUCT DESCRIPTION

TurboDOS 1.43+16a.00 has been released to update the previous 16-bit TurboDOS versions. TurboDOS 1.43+16a.00 has a major improvement in the 8086 memory model. The improvement relaxes the 64KB memory size restriction imposed by prior versions. increasing the memory space available to the 8086 operating system enables the new version to support:

- more memory space available for users
- more open files
- more drivers, and
- more-complex drivers

See Appendix B for more information on the 8086 memory model.

The memory model enhancements do not affect Z80 TurboDOS.

The TurboDOS 1.43+16a.00 release is contained on four disks:

1. GEN-16: 16-bit GEN and old SYS
2. LIB-16: 16-bit LIB
3. SYS-16: 16-bit SYS and CMD
4. SYS-8: 8-bit SYS, COM, GEN and LIB

TurboDOS 1.43+16a.00 Changes

1. The COPY command supplied by L/F Technologies has been corrected to allow it to operate from a DO loop. The command also leaves the attributes (such as "global", "FIFO", and so on) intact the same way the Software 2000 COPY does. (Remember: when creating a DO file, do not end the file with a <CR> (Carriage Return)).

2. TLINK has been changed to support the memory model changes. See Appendix A for details.

3. TurboGEN has been changed to support both the 8080 and the small memory model.

TurboDOS 1.43+16a.00 Enhancements

1. The COPY command supplied by LFT now has an **Append** option for concatenating files. The command format follows:

   COPY <SRCFILE> <DESTFILE> ;S

   Note: DESTFILE must not be a wildcard filename. The command appends the SRCFILE to DESTFILE. However, the SRCFILE may be a wildcard filename.

2. There is now a **RESET** command for the 16-bit master systems to reset dead user processors. See Appendix D for RESET details.

TurboDOS 1.43+16a.00 Notes

- SUBMIT relies on the OS being configured as an 8080 model; it will not operate under the new small model. To use SUBMIT, generate TurboDOS using the TLINK −8 option to produce an 8080 model system.

- Some compilers will not operate under the 1.43 small model because of the way the compilers manage their stack segments. These compilers will operate under the 8080 model. If you find that a compiler will not operate under the small memory model, recompile the OS to the 8080 model (using the -8 option; see the TurboDOS Utility Enhancements in Appendix A of this document).

- Warning:When using programs that use multisector I/O, such as the LFT COPY program, a maximum number of message buffers, MAXMBS, must be set or all available dynamic memory will be used for message buffers. This will leave no memory for opening files. Also, any program that uses multisector I/O for writing may lock up the system on a Write error.

- Warning:On systems using spooling, **every assigned printer must be assigned a different queue.** The following example illustrates the assignments for three printers:

  Printer A assigned to **Queue A**
  Printer B assigned to **Queue B**
  Printer C assigned to **Queue C**

  The system will not operate if all printers despool from the same queue.

- STDxxxxx - The GEN and PAR files support the A1100 Winchester controller and the A1250 Serial-Parallel I/O boards. The Printer Assignment Table has been changed so that TurboGEN correctly configures parallel printers. The number of buffers has been reduced from 32 to 4. The buffer size has been increased from 1024 to 8192 bytes. The buffer size change increases performance of I/O operations. (Note that if frequent Compile operations are done, a greater number of smaller buffers may be preferable.) A limit has been placed on the maximum number of message buffers and reply packets, allowing multiuser systems to use multisector I/O programs without locking up the system.

- OLDxxxxx - The GEN and PAR files support the A1100 Winchester board and the A480 and A630 boards. The buffer size, number of buffers, and maximum number of message and reply buffers have been modified in the same manner as those in the STDxxxxx files.

- **SLV1240.PAR, SLV1081.PAR, SLV1270.PAR, and SLV862.PAR** - These files have been modified to correct a problem with random writes that cross extents when the Multisector Count is greater than 1.

- The current versions of the LFT utilities are as follows:

RESTRK 1.4	READPC 2.1
FMTWIN 3.6	WRITEPC 2.1
FMTF 3.3	COPYD 2.1
TURBOGEN 3.9	DATE 3.0
COPY 1.1	RESET 1.0

Appendix A

TurboDOS 1.43 Changes

T-Function 44 – Open File

This function opens the file specified by an FCB who address is passed in DE (or DX). It differs from C-function 15 (Open File) as follows:

1. The S2 byte of the FCB is not zeroed.
2. FCB interface attributes F5 & F6 are ignored.
3. File sharing is determined by a sharing vector passed in B (or CH).

The sharing vector is defined as follows:

7	6	5	4	3	2	1	0
DR	DW	DC	DP	AR	AW	AC	AP
Sharing				Access			

Sharing bits:

```
DR  =  deny read
DW  =  deny write
DC  =  deny compatibility
DP  =  deny permissive
```

Access bits:

```
AR  =  access read
AW  =  access write
AC  =  access compatibility
AP  =  access permissive
```

TurboDOS/PC maps the MS-DOS sharing and access modes into a sharing vector as follows:

MS-DOS sharing modes:

```
0  -  compatibility    →    AC
1  -  deny read/write  →    DR+DW+DC
2  -  deny write       →    DW+DC
3  -  deny read        →    DR+DC
4  -  deny none        →    DC
```

MS-DOS access modes:

```
0  -  read access        →    AR
1  -  write access       →    AW
2  -  read/write access  →    AR+AW
```

TurboDOS maps the MP/M-style open modes into a sharing vector as follows:

```
exclusive mode        →    DR+DW+DC+AR+AW
shared mode           →    DC+AR+AW
permissive mode       →    AR+AP
read-only mode (MP/M) →    DW+DC+AR
read-only mode (mixed) →   DC+AR
```

T-Function 45 – Make File

This function creates the file specified by an FCB whose address is passed in DE (or DX). It differs from C-function 22 (Make File) as follows:

1. The S2 byte of the FCB is not zeroed.
2. FCB interface attributes F5 and F6 are ignored.
3. File sharing is determined by a sharing vector passed in B (or CH).

The sharing vector is defined exactly the same as for T-function 44.

T-Function 46 – Lock File Region

This function locks a region of the file specified by an FCB whose address is passed in DE (or DX). The region to be locked is specified by a 32-bit byte-offset and a 32-bit byte-length passed at the current DMA address:

current DMA address →

(lsb)	byte offset	(msb)
(lsb)	byte length	(msb)
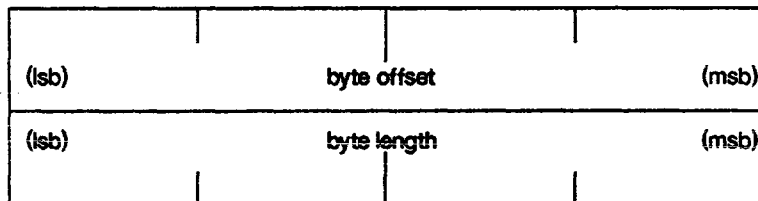
MP/M-style record locks (C-function 42) have the effect of locking regions whose length is always 128 and whose offset is always a multiple of 128. An "all-inclusive lock" locks the maximal region whose offset is 0 and whose length is FFFFFFFF hex.

T-Function 47 – Unlock File Region

This function unlocks a region of the file specified by an FCB whose address is passed in DE (or DX). The region to be locked is specified by a 32-bit byte-offset and a 32-bit byte- length passed at the current DMA address. The region is specified in the same fashion as described above, and must be exactly the same region specified in a previous lock.

C-Function 159 – Detach List Device

This MP/M function is now supported by TurboDOS if the module "MPMSUP" is installed. It performs exactly the same function as T-function 28 (Signal End-of-Print). The purpose of this enhancement is to improve the action of spooled printing with certain applications written in RM-COBOL.

New Compatibility Flag: Global Inhibit

Bit 2 of the TurboDOS compatibility flags has been defined as the "global inhibit" flag. If this bit is set to 1 (by means of the COMPAT patch point or T-function 13), global files in user number zero are not accessible from non-zero user numbers.

The principal purpose of this enhancement is to make it possible to inhibit global file access from TurboDOS/PC.

TurboDOS Utility Enhancements

TLINK Changes: To support the memory model changes in 8086 TurboDOS 1.43, TLINK has been changed so that the following linker options are all independent:

    -8 = 8080 model
    -B = no base page
    -H = no .CMD header

(In pre-1.43 versions of TLINK, the -H option implied both -8 and -B options.)

When using TLINK for system generation of 8086 TurboDOS, the options -B and -H are automatically enabled. The operating system is linked as a small model by default. However, the operating system may be linked as an 8080 model by including the -8 option explicitly in the TLINK command.

SET and SHOW Commands: These TurboDOS commands now honor a new "D" option for manipulating and displaying the "Directory" (F2) attribute, which TurboDOS/PC uses to identify MS-DOS directories. The D option is used to identify MS-DOS subdirectories as directory files in networks containing both LFT and IBM PCs.

Appendix B

TurboDOS 1.43 8086 Memory Models

Before version 1.43, 8086 TurboDOS was implemented using an "8080 memory model"; all code and data used by the operating system (including drivers) was limited to a single memory segment that could not exceed 64KB. (See Figure 1.) A typical master OS configuration requires about 32KB for code and static data, leaving only about 32KB for the dynamic data space that TurboDOS uses to keep track of users, open files, locked records, and so forth. (Unusually large drivers leave even less dynamic space.) The amount of memory available for dynamic data space limits the number of users and open files that a single TurboDOS server may support.

The TurboDOS 1.43 8086 version uses a "small memory model". In this model the OS uses two memory segments: one containing code, and the other containing data. Each segment may be up to 64KB long (see Figure 2). The small memory model nearly doubles the amount of dynamic data space available in a typical configuration. The model also permits large driver modules to be used without decreasing the amount of dynamic data space available.

Figure 1. — 8080 Memory Model (prior to 1.43).

Figure 2. — Small Memory Model (1.43 & beyond).
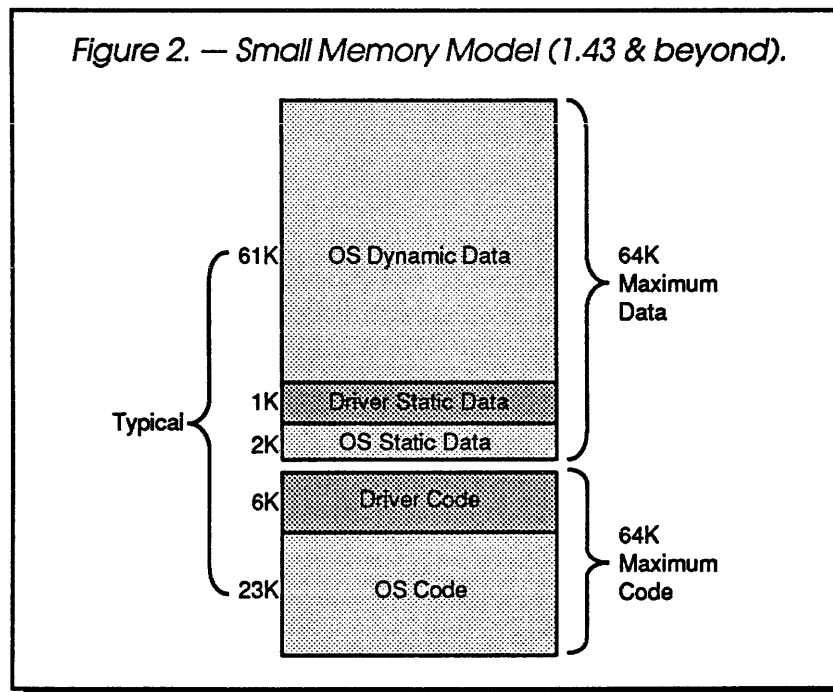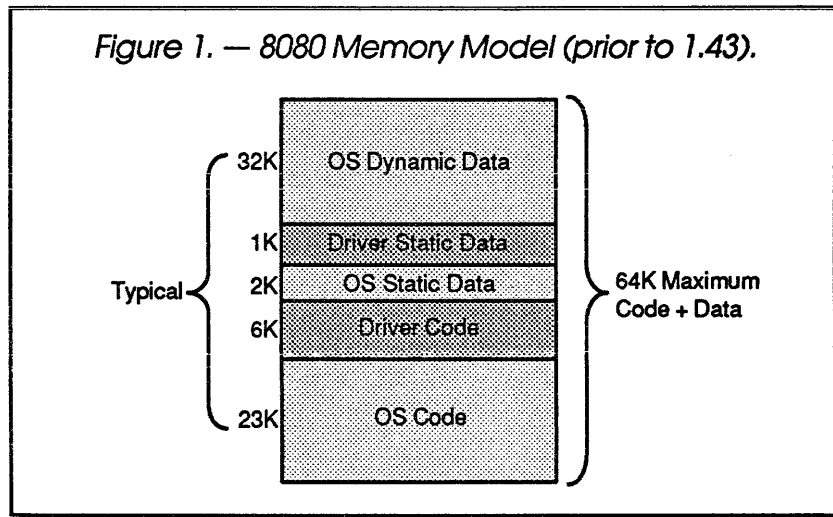
Appendix C

TurboDOS 1.43 8086 Programmer Implementation Information

8086 Driver Changes

Programmers must make minor changes to some existing driver modules to operate with release 1.43. The changes are made to poll routines.

Prior to 1.43, the code of each poll routine was immediately preceded by a two-word linkage (for example, PL in Figure 3). The OS uses this linkage to maintain a linked list of active poll routines.

```
Figure 3. — Old poll routine coding (prior to 1.43).

          LOC  Data#      ;data segment
     SPH: WORD 0,.+2,.+2   ;event semaphore

          LOC  Code#      ;code segment
          ...
          MOV  DX,&PL      ;get linkage addr
          CALL LNKPOL#     ;activate poll rtn
          CALL PR          ;optional pre-test
          MOV  BX,&SPH     ;get semaphore addr
          CALL WAIT#       ;wait for event
          ...
     PL:  WORD 0,0         ;poll rtn linkage
     PR:  IN   AL,=STAT    ;get device status
          TEST AL,=MASK    ;did event occur?
          JZ   PX          ;if not, exit
          MOV  BX,&SPH     ;get semaphore addr
          CALL SIGNAL#     ;signal event
          MOV  BX,&PL      ;get linkage addr
          CALL UNLINK#     ;deactivate poll rtn
     PX:  RET              ;done
```

```
Figure 4. — New poll routine coding (1.43 & beyond).

          LOC  Data#      ;data segment
     SPH: WORD 0,.+2,.+2   ;event semaphore
     PL:  WORD 0,0,&PR     ;poll rtn linkage

          LOC  Code#      ;code segment
          ...
          MOV  DX,&PL      ;get linkage addr
          CALL LNKPOL#     ;activate poll rtn
          CALL PR          ;optional pre-test
          MOV  BX,&SPH     ;get semaphore addr
          CALL WAIT#       ;wait for event
          ...
     PR:  IN   AL,=STAT    ;get device status
          TEST AL,=MASK    ;did event occur?
          JZ   PX          ;if not, exit
          MOV  BX,&SPH     ;get semaphore addr
          CALL SIGNAL#     ;signal event
          MOV  BX,&PL      ;get linkage addr
          CALL UNLINK#     ;deactivate poll rtn
     PX:  RET              ;done
```

Starting with 1.43, this poll routine linkage must be moved to the data segment, and a pointer to the poll routine code must be appended to the linkage (see Figure 4).

Programmers must also find and eliminate any code sequences in the driver code which rely implicitly on the 8080 memory module (for example, code which assumes that CS = DS). If you have trouble bringing up TurboDOS 1.43, the most likely cause is a driver coding problem of this sort.

Note: Z80 driver modules require no such changes.

Appendix D

TurboDOS 1.43 - RESET Command Operation

The **RESET** utility will cause the reset of a named user processor. The command is

RESET

The user will be prompted with:

Enter circuit number (0 - 1, CR exits)?

If the response is **0**, the user will be prompted with:

Enter slave number (1 - 16, CR exits)?

If the response is **1** the prompt is:

Enter slave number (1 - 30, CR exits)?

If the number of a non-existent user processor (slave) is entered, RESET will appear to function, but actually nothing will happen.

RESTRICTIONS:

- This utility must be run from the master.

- Only local user processors may be reset.

REMARKS:

If a Circuit 0 user attempts to reset a Circuit 1 user processor, the reset may or may not succeed. The Circuit 1 user processor will wait for the Reset to occur, until a Circuit 1 user issues a command (any command). Once the command is passed over the network and obtains the master's attention, the Reset will proceed.