Tektronix

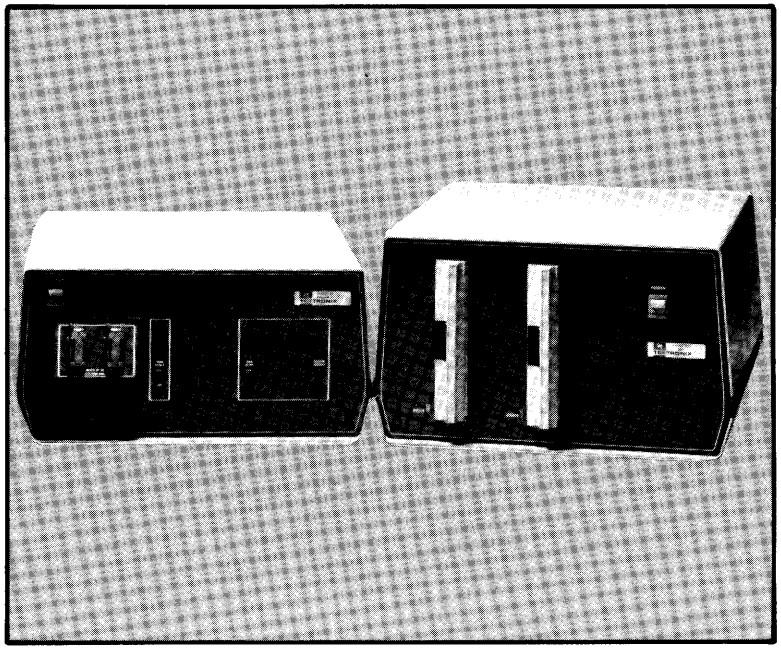
COMMITTED TO EXCELLENCE

8002A

μPROCESSOR LAB

System Reference Booklet

TEKDOS VERSION 3



070-2706-01 REV A, FEB 1979

TABLE OF CONTENTS

Section 7 Tables

Section 8 Error Explanations

Section 1
OVERVIEW

The 8002A μProcessor Lab is a tool for developing and testing microprocessor-based products. The block diagram in Fig. 1-1 illustrates the relationship between the 8002A and the various system components. The minimum 8002A system components are highlighted. All other components are optional.

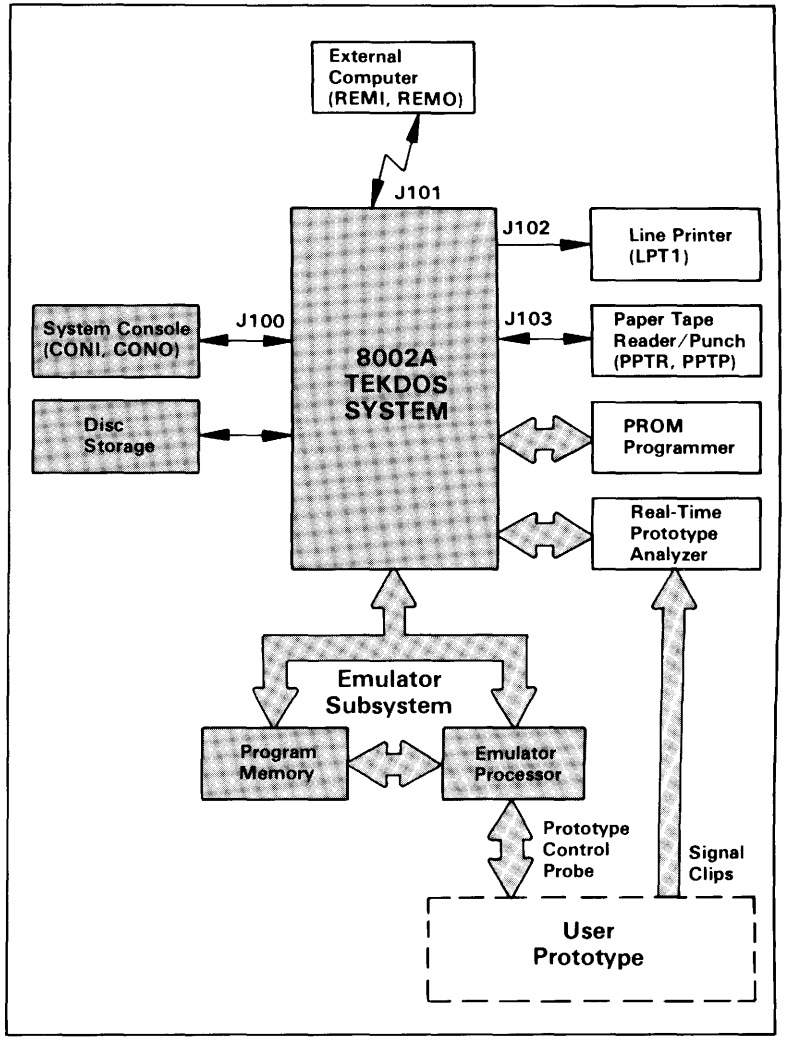


Fig. 1-1. 8002A block diagram.

The 8002A communicates with the various peripheral devices through jacks J100, J102, and J103. Communication with the external computer takes place through J101. The 8002A communicates with the Flexible Disc Unit through a cable attached to the Debug and Front Panel I/O module. This module resides within the 8002A mainframe.

The optional PROM Programmer and RTPA modules also reside within the 8002A mainframe. The PROM Programmer sockets on the 8002A Front Panel are connected via a cable to the PROM Programmer module. These sockets allow you to read from, compare, and write to, three different types of PROMs: 1702, 2704, and 2708. The RTPA permits the 8002A to store and monitor Emulator Processor operations at full processing speed.

Table 1-1

Device Names, Jacks, and Functions

Device Name	Jack	Function
CONI	J100	Console input
CONO	J100	Console output
LPT1	J102	Line printer
PPTP	J103	Paper tape punch
PPTR	J103	Paper tape reader
REMI	J101	Remote input (external computer)
REMO	J101	Remote output (external computer)
TTYR	J100	Teletypewriter/reader

File Name Specifications

File names may not exceed eight characters. File names must begin with a letter. The rest of the characters in the file name may be alphanumeric, or may include the following special characters:

? % " = ; * ( ) # & ! @

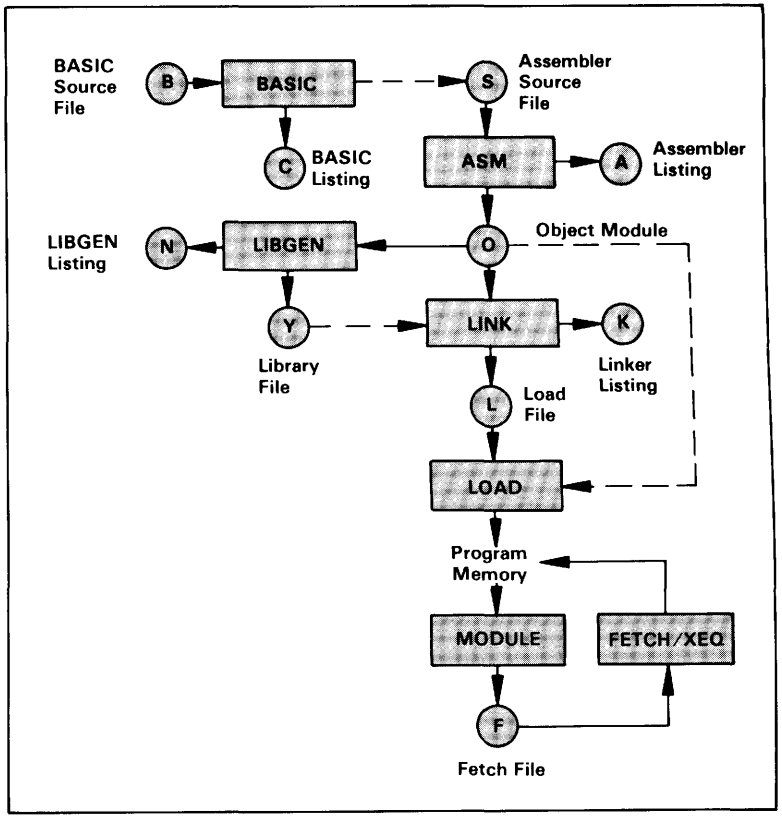


Fig. 1-2. File types diagram.

This figure shows how each file type is created and where it is used. The square boxes represent commands, while circles represent file types. Each file type is identified by the suffix letter as described in Table 1-2.

Table 1-2

Suggested File Type Suffixes

File Type	Suggested Suffix Character
Assembler source	S
Object Module	O
Load	L
Fetch	F
Assembler listing	A
Linker listing	K
Libgen listing	N
Library listing	Y
BASIC source	B
BASIC listing	C
Tekhex	X
Vendor hex	V

To indicate temporary files, you may precede the suffix letter with a semicolon. To indicate permanent files, you may precede the suffix letter with a percent sign.

TEKDOS COMMAND MEMORY AREAS

Memory Area 1

COPY
CSMS
DEBUG
DUP
FORMAT
PRINT
PRINTL
RHEX
RVHEX
RSMS
SEND
VERIFY
WHEX
WVHEX
WSMS

Memory Areas 1 and 2

CMPF
COMM
CPROM
LDIR
LOAD
MODULE
RPROM
WPROM

Memory Area 2

ABORT
ASSIGN
BIAS
BIF
BKPT
CAL
CLBP
CLOCK
CLOSE
CNT
CONT
DELETE
DEVICE
DISM
DRT
DSTAT
DUMP
EVT
EMULATE
EXAM
FILL
HELP
KILL
MAP
MOVE
PATCH
RENAME
RESET
RTT
SET
STATUS
SUSPEND
TRACE
TYPE

Program Memory

ASM
BASIC
CONVERT
EDIT
FDUMP
FIX
GET
LIBGEN
LINK
REPLACE

Resident Commands

FETCH
GO
SYSTEM
XEQ

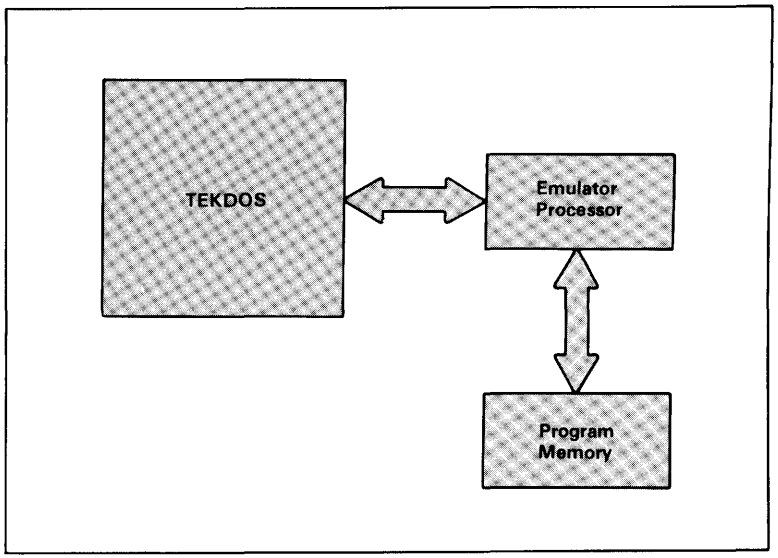
Command Files

User-Defined
COPYSYS
NEWDISC

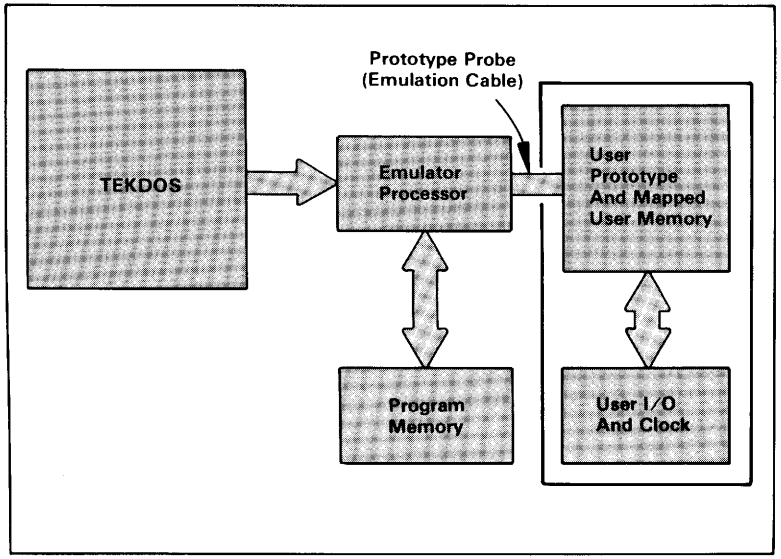
EMULATION MODE CONFIGURATIONS

The following diagrams illustrate the three types of emulation modes:

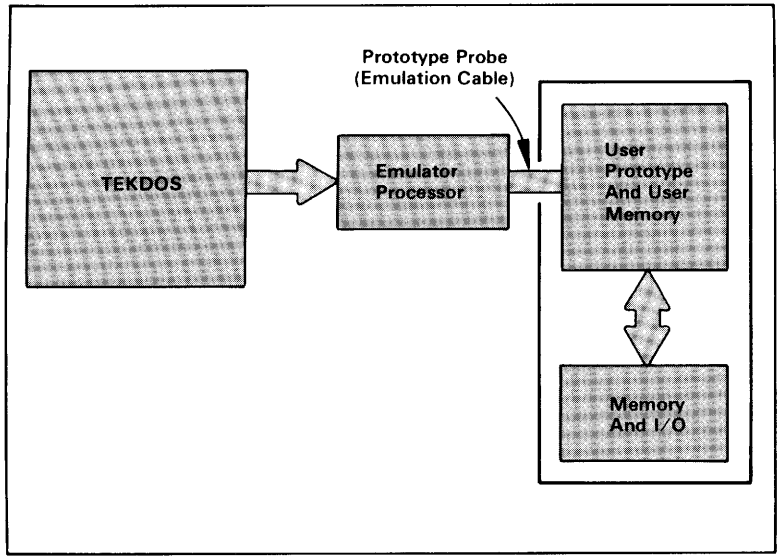
Emulation Mode 0 — System Mode



Emulation Mode 1 — Partial Emulation



Emulation Mode 2 — Prototype Mode



Section 2
COMMAND DICTIONARY

COMMAND INDEX

System Commands

ABORT	CMPF	FETCH	LIBGEN	REPLACE
ASM	CONT	FIX	LINK	STATUS
ASSIGN	CONVERT	FORMAT	LOAD	SUSPEND
BASIC	DELETE	GET	MAP	SYSTEM
BIAS	DEVICE	HELP	MODULE	TYPE
CLOCK	EDIT	KILL	NEWDISC	VERIFY
CLOSE	EMULATE	LDIR	RENAME	XEQ

Data Transfer Commands

COMM	DUP	MOVE	RHEX	WHEX
COPY	EXAM	PATCH	RVHEX	WVHEX
COPYSYS	FDUMP	PRINT	SEND	
DUMP	FILL	PRINTL		

Debug Commands

BKPT	CLBP	DISM	GO	SET
CAL	DEBUG	DSTAT	RESET	TRACE

Special Function Keys

BACK SPACE	ESC	Space Bar
CTRL Z	RUB OUT	

Real-Time Prototype Analyzer Commands

BIF	CNT	DRT	EVT	RTT

PROM Programmer Commands

CPROM	CSMS	RPROM	RSMS	WPROM	WSMS

COMMAND SYNTAX CONVENTIONS

The following list summarizes the conventions used in the syntax blocks contained in this booklet:

1. UPPERCASE words must be entered exactly as shown. These include command names, such as DUP, and parameters, such as OFF.

2. Lowercase words represent items that identify the entry type. These include items like offset amounts, program memory addresses, and file names.

3. Special characters must be entered exactly as shown. These include: * , = : / and the hyphen.

4. Required command line parameters are enclosed in {braces}. Optional command line parameters are enclosed in [brackets].

5. Minimum entries are underlined.

6. To omit an optional parameter in a command line (and thus specify the default value), use two consecutive commas in place of the parameter. To omit two consecutive optional parameters, use three consecutive commas.

7. If parameters are stacked one above another, you may choose only one.

8. Parameters followed by trailing dots may be repeated.

TEKDOS COMMANDS

ABORT { command file name / system command name / * / / }

Terminates user or system program execution. The asterisk parameter aborts all active programs. The slash parameter terminates all user programs and closes all assigned channels.

ASM [object device / object file name[/disc drive]] [list device / list file name[/disc drive]] { source device / source file name[/disc drive] } ...

Creates object and list files from source files. You must specify the source device or file. Default parameter: system disc.

ASSIGN {channel number} { device / file name [/disc drive] } [channel number { device / file name[/disc drive] }] ...

Connects channels to files and devices. CONO, CONI, and file names may each be assigned to more than one channel. All other devices are limited to one channel assignment at a time. Once a device or file has been assigned to a channel, the system cannot access that device, except through service calls. Default parameter: system disc.

BACK SPACE Key (CTRL H)

Deletes the last character entered. Use the RUBOUT key to delete and echo the last character entered.

BASIC [device / assembly file name[/disc drive]] [list device / list file name[/disc drive]] { source device / source file name[/disc drive] }

Invokes the BASIC compiler. You must specify the source device or file. Default parameter: system disc.

BIAS [W=offset] [X=offset] [Y=offset] [Z=offset]

Sets or displays the DUMP, EXAM and PATCH address offsets. All offsets default to 0 at system startup or reboot.

BIF

or

BIF {mode} [C / S]

or

BIF [1 / 2] {CLR}

Sets, clears, or displays BIF command line break options. Used in conjunction with the EVT command. Allowable mode parameters: 1, 2, ARM, LIM, IND, FRZ. Only available in the debug system. Default value: S (suspend).

BKPT {address} [READ / WRITE] [C / S]

Sets breakpoints. Defaults to either READ or WRITE. Only available in the debug system. Default value: S (suspend).

CAL

Allows hexadecimal calculations. Only available within the debug system. Press the ESC key to return to the debug system.

CLBP [address]

Clears breakpoints that were set with the BKPT command. Only available in the debug system. To clear both breakpoints, enter this command without a parameter.

CLOCK {ON / OFF}

Enables or disables the program clock. Access the program clock value with the READ PROGRAM CLOCK SVC. System startup or reboot default: OFF.

CLOSE {channel number} [channel number] . . .

Disconnects channels from files or devices. Channel number must be a number from 0 to 7. To disconnect all assigned channels, enter the ABORT command with the slash parameter.

CMPF {file name[/disc drive]} { file name[/disc drive] } [output device / output file name[/disc drive]]

Compares two files and lists differences to a file or device. Default parameters: CONO, system disc.

CNT [option]

Sets or displays the general purpose counter units. Allowable option parameters: C (bus cycles), E (emulator clock cycles), F (bus fetch cycles), M (milliseconds), T (RTT buffer transactions), U (microseconds), 1 (EVT1 comparisons), 2 (EVT2 comparisons). Default parameter: M.

COMM [E={L / R}] [L={I / O}] [P=prompt sequence] [T=delay time] [M=parity] [C={T / I}]

Allows communication with an external computer. Default parameters: E=R, L=O, no prompt sequence, T=00, M=04, C=T.

CONT { command file name / system command name / * / / }

Continues execution of a suspended system or user program. The asterisk parameter continues all suspended programs.

CONVERT {source device / source file name [/disc drive]}

{destination device / destination file name [/disc drive]}

Converts vendor assembler source code to Tektronix Assembler source code. Default parameter: system disc.

COPY {input device name / input file name[/disc drive]} [input device name / input file name[/disc drive]] ...

{output device name / output file name[/disc drive]}

or

COPY {input file name [/disc drive]}

Transfers data from one device or file to another device or file. Output defaults to CONO if second form is used. Default parameter: system disc.

COPYSYS {input disc drive} {output disc drive}

Copies the TEKDOS operating system from one disc to another. Use the NEWDISC command to format, verify, and create a new TEKDOS system disc.

CPROM [memory address] [1702 / 2704 / 2708] [prom lower address]

[prom upper address] [0 / 1]

Compares PROM data with program memory data. Default parameters: 0, 2708, 0, highest addressable PROM location, 0.

CSMS [program memory address] [device / file name[/disc drive]]

Compares SMS formatted data with program memory data. Memory address defaults to 0. Device must not be CONO. Default parameter: system disc.

CTRL Z

Provides an end-of-file character during an ASCII read operation.

DEBUG [device / file name[/disc drive]]

Loads the debug system into system memory. Default parameter: CONO, system disc.

DELETE {file name/disc drive} [file name/disc drive] ...

Deletes the specified files from the flexible disc. Disc drive must be specified.

DEVICE {device name} {U / D}

Informs TEKDOS of device availability. All devices default to U (up).

DISM [lower address] [upper address]

Translates object code to mnemonics and operands. Only available in the debug system. If you only enter the upper address parameter, disassembly begins at address 0000.

DRT [* / number of bus transactions]

Displays the real-time trace buffer contents. Only available in the debug system. The asterisk displays all transactions.

DSTAT [SHORT / LONG]

Displays the Emulator Processor status. Only available in the debug system. Default parameter: SHORT.

DUMP [W / X / Y / Z] {lower address} [upper address] [device / file name[/disc drive]]

Transfers memory contents to a file or device. Default parameters: zero offset, CONO, system disc.

DUP {input disc drive} {output disc drive} [output disc name]

Copies the contents of one disc to another. Before you enter this command, format and verify the output disc.

EDIT [infile name[/disc drive]] [outfile name[/disc drive]]

Invokes the 8002A Editor. Default parameter: system disc.

EMULATE {emulation mode}

Selects the emulation mode. Possible modes: 0, 1, or 2. Your prototype must be connected in modes 1 and 2.

ESC Key

Suspends or terminates program execution. Deletes lines from the command input buffer.

EVT

or

EVT {mode}

or

EVT [1 / 2] {CLR}

or

EVT {1 / 2} [CLR] {option} {operator} {value}

[{option} {operator} {value}]...

Sets, clears, or displays EVT command line options. Used in conjuction with the BIF command. Allowable mode parameters: ARM, LIM, IND. Allowable option parameters: A (bus address), D (bus data), T (test clips), B (bus options), P (pass count), C (counter delay).

EXAM [W / X / Y / Z] {address}

Displays or alters data. Default parameter: zero offset.

FDUMP {input file name[/disc drive]} [output device / output file name[/disc drive]] [N / W]

Displays a file in its hexadecimal and ASCII representation. Default parameters: CONO, system disc, N (for CONO display), W (for LPT1 display).

FETCH {fetch file name[/disc drive]} [program parameters]...

Loads a fetch file from a disc to program memory. Default parameter: system disc.

FILL {lower address} {upper address} {hexadecimal string}

Fills memory with a hexadecimal string.

FIX

Compares the master bit map with the file bit maps and lists differences on the system console. Insert the known-good system disc in drive 0 and the disc to check in drive 1.

FORMAT {disc drive} [disc name]

Prepares a disc for use with TEKDOS. Disc drive may not specify the system disc.

GET {fetch file[/disc drive]} {system overlay file[/disc drive]}

{member number}

or

GET {@command file[/disc drive]} {system overlay file[/disc drive]}

[log file[/disc drive] / log device]

Copies one or more System Overlay File members to the specified fetch files for modification. Default parameters: CONO, system disc.

GO [address]

Passes execution control to the Emulator Processor.

HELP {ERROR / command name / error code} [output device / output file name[/disc drive]]

or

HELP

or

HELP,, [output device / output file name[/disc drive]] [SHORT / LONG]

Displays command syntax and brief error message explanations. Default parameters: CONO, SHORT, system disc.

KILL {ON / OFF}

Terminates or continues command file execution upon error detection. System startup or reboot default parameter: ON.

LDIR [disc drive] [{:file name} [:file name]...] [.] [/]

[output device / output file name/disc drive]

Outputs the disc directory contents to a file or device. Disc drive must be specified for output file name. The plus sign functions as the wild card character. Default parameters: CONO, system disc.

LIBGEN [newlib] [,listfile [/disc drive] / ,list device] [,oldlib]

or

LIBGEN

or

LIBGEN {@command file name[/disc drive]}

Creates or modifies a library file. Default parameter: system disc.

LINK [load file name[/disc drive]][list device / list file name[/disc drive]]

{object1 [/disc drive] / LIB(libfile) [/disc drive]} [,object2[/disc drive] / ,LIB(libfile)[/disc drive]] ...

or

LINK

or

LINK {@command file name[/disc drive]}

Merges object code files into one executable program. Default parameter: system disc.

LOAD [/offset amount] {file name[/disc drive]} [file name[/disc drive]] ...

Reads Assembler or Linker object files into program memory. Default parameters: zero offset, system disc.

MAP [M / R]

or

MAP {U / P} {{lower address}[–upper address]}

[{lower address} [–upper address]] ...

Displays or sets memory maps assignments. The hyphen must be entered with the upper address parameter. If the system is not in emulation mode 1, the system console displays the message WARNING—EMULATOR NOT IN MODE 1.

MODULE {file name[/disc drive]} {lower address} {upper address}

{starting address} [identifying string]

Writes binary code from program memory into a fetch file on disc. Default parameter: system disc.

MOVE {PP / PU / UP / UU} {lower address} {upper address} {starting address}

Copies data from one memory location to another.

NEWDISC

Formats, verifies, and copies the TEKDOS operating system from the disc in drive 0 to the disc in drive 1. Use COPYSYS to copy the TEKDOS operating system from one disc to another.

PATCH [W / X / Y / Z] {address} {hexadecimal string}

Alters memory with a hexadecimal string. Defaults to zero offset.

{PRINT / PRINTL} {input device / input file name[/disc drive]} [output device / output file name[/disc drive]]

[{lower line number} [upper line number]]

Copies data from a file or device to another file or device. PRINTL numbers the lines. Default parameters: one line, LPT1, system disc.

RENAME {old file name/disc drive} {new file name[/disc drive]}

or

RENAME {disc drive} {disc name}

Renames a file or disc. First disc drive parameter must be specified; second disc drive parameter defaults to the disc drive number specified with old file name.

REPLACE [fetch file[/disc drive]] {system overlay file[/disc drive]}

{member number} [new system overlay file[/disc drive]]

or

REPLACE {@command file[/disc drive]} {system overlay file[/disc drive]}

[log file[/disc drive] / log device] [new system overlay file[/disc drive]]

Creates a new SOF (System Overlay File) that is a copy of an updated SOF. Default parameters: CONO, system disc.

RESET

Sets known values in Emulator Processor registers. Only available in the debug system.

RHEX [/offset amount] {device / file name[/disc drive]}

Reads and converts Tektronix Hexadecimal Format code to binary and stores the code in memory. Default parameters: zero offset, PPTR, system disc.

RPROM [memory address] [1702 / 2704 / 2708] [prom lower address]

[prom upper address] [0 / 1]

Transfers PROM data to program memory. Default parameters: 0, 2708, 0, highest addressable PROM location, 0.

RSMS [program memory address] [device / file name[/disc drive]]

Reads SMS formatted data into program memory from a file or device. Default parameters: 0000, CONI, system disc.

RTT [option]

Selects or displays the type of RTT buffer transactions stored. Allowable option parameters: ALL (all bus transactions), F (instruction fetches), I (I/O addresses), IR (I/O reads), IW (I/O writes), M (memory accesses), MR (memory reads), MW (memory writes), R (read operations), W (write operations). Default parameter: ALL.

RUB OUT Key

Deletes and displays the last character entered on the system console. Use the BACK SPACE key to delete but not display characters.

RVHEX [/offset amount] {device / file name[/disc drive]}

Reads and converts vendor-formatted object code to binary and stores the code in memory. Default parameters: zero offset, PPTR, system disc.

SEND

Transfers Tekhex object code files from the 8002A to 8001 program memory.

SET {initial register} {hexadecimal value} [hexadecimal value] . . .

Alters Emulator Processor register contents. Only available in the debug system.

Space Bar

Suspends or continues the console display. Press once to suspend the display; press again to continue.

STATUS

Displays Emulator Processor, program execution, channel, and command file status.

SUSPEND { command file name / system command name / * }

Suspends user or system program execution. The asterisk suspends all active programs.

SYSTEM [disc drive]

Designates the system disc drive. Default value: system disc.

TRACE

or

TRACE { ALL / JMP / OFF } [STEP] [[lower address] [upper address]]

or

TRACE { LONG / SHORT }

Allows you to monitor program execution. Only available in the debug system. LONG and SHORT affect only certain microprocessors. Up to three TRACE commands may be active at one time. Default parameters: SHORT, 0000, FFFF.

TYPE { ON / OFF }

Displays or suppresses display of each command in a command file as the command executes. System startup or reboot default: ON.

VERIFY {disc drive}

Checks for and records any bad disc sectors. Use after the FORMAT command.

WHEX {lower address} {upper address}

[{,,lower address} {upper address}] ...

[{starting address} [device / file name[/disc drive]]]

Converts program memory contents to Tektronix Hexadecimal Format and writes the converted contents to a device or file. Default parameters: CONO, system disc.

WPROM [memory address] [1702 / 2704 / 2708] [prom lower address]

[prom upper address] [0 / 1]

Writes data from program memory to a PROM. Default parameters: 0, 2708, 0, highest addressable PROM location, 0.

WSMS [program memory address] [device / file name[/disc drive]]

Translates program memory data to SMS format. Writes the data to a file or device. Default parameters: 0000, PPTP, system disc.

WVHEX {lower address} {upper address}

[{,,lower address} {upper address}] ...

[{starting address} [device / file name[/disc drive]]]

Converts program memory contents to vendor format and writes the converted contents to a file or device. Default parameters: CONO, system disc.

XEQ {fetch file name[/disc drive]} [program parameters] . . .

Loads a fetch file into program memory and executes the file. Equivalent to the FETCH and GO command sequence. Default: system disc.

Section 3
EDITOR DICTIONARY

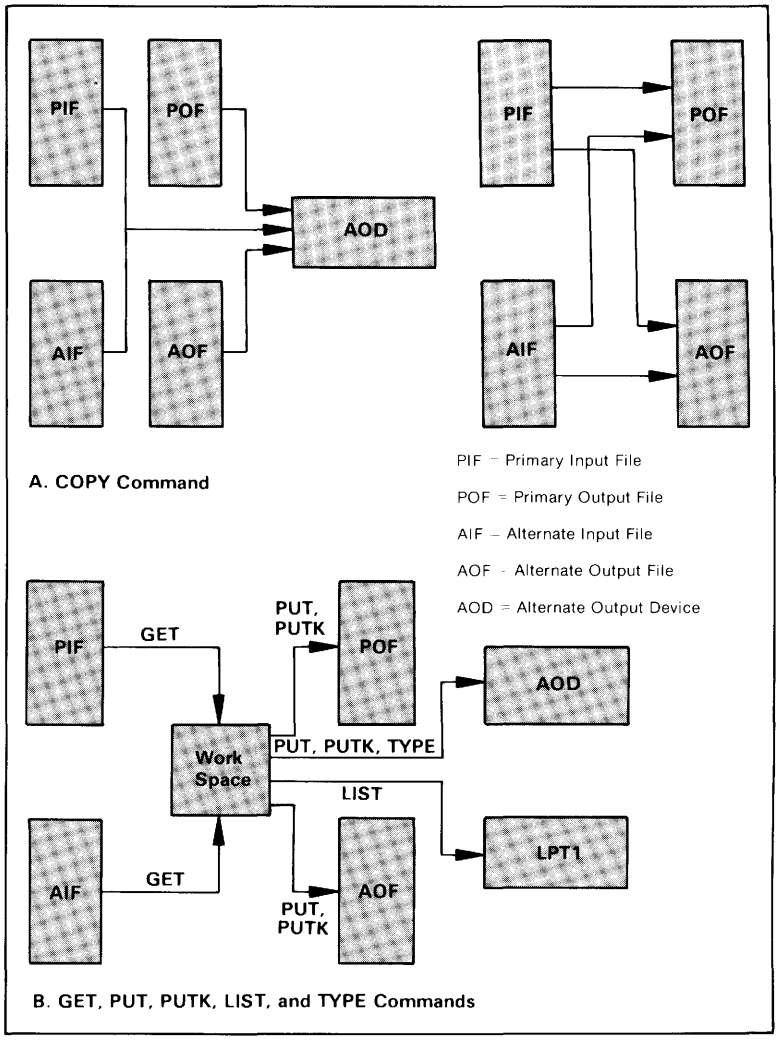


Fig. 3-1. Editor text flow diagram.

These block diagrams illustrate how the Editor can move text between the primary and alternate files. Fig. 3-1(A) demonstrates how text may be transferred with the COPY command. Fig. 3-1(B) depicts the flow of text to and from the workspace.

EDITOR INDEX

Move Pointer Commands

BEGIN	DOWN	END	FIND	UP

Change Text Commands

INPUT	INSERT	KILL	REPLACE	SUBSTITUTE

Transfer and Display Commands

COPY	FILE	GET	
LIST	PUT	PUTK	TYPE

Utility Commands

AGAIN	BRIEF	ESC Key	MACRO	N
QUIT	SUSPEND	TAB	TABS	?

EDITOR SYNTAX CONVENTIONS

The following conventions are used with the Editor commands. The Command Syntax Conventions specified in Section 2 also apply to the Editor commands.

1. The colon character (:) allows you to enter several commands on the same line. Insert a colon between each command on the input line.

2. Use the right and left arrows to repetitively execute a command line. Enclose the command line with the arrows as follows:

   12<PUT 1-5 LPT1:D.10>

   The number that precedes the left arrow designates the number of times the command line will be repeated. The arrows may be nested up to 16 levels.

3. The letter B (beginning line), C (current line), or E (ending line) may be entered as a parameter anywhere a number or number of lines is required. A space must follow the command and the B, C, or E parameter.

EDITOR COMMANDS

AGAIN

Repeats the previous command. Does not repeat the BRIEF, TAB, TABS, or AGAIN commands. Does not redefine a macro, but repeats the execution of the last repeatable command of a macro.

BEGIN

Moves the workspace pointer to the first line of text and displays the line. To suppress the display, follow this command with a period.

BRIEF

Disables the display of certain Editor commands. Enter BRIEF again to re-enable the display.

COPY

COPY { number of lines / range of lines } { primary input file / alternate input file [/disc drive] }

[output device / alternate output file [/disc drive]]

Copies text from an input file to an output file or device. If you do not specify an output file or device, the Editor appends text to the primary output file.

DOWN [number of lines]

Moves the workspace pointer down the specified number of lines and displays the line. To suppress the display, follow this command with a period. Default parameter: one line.

END

Moves the workspace pointer past the last line of text and displays the text *EOF*. To suppress the display, follow this command with a period.

ESC Key

Deletes an input line or suspends the Editor. Press once to delete an input line; press twice to suspend the Editor and return control to TEKDOS.

FILE

Transfers text from the workspace to the primary output file and returns control to TEKDOS. To return control to TEKDOS without saving text, enter the QUIT command.

FIND {delimiter string delimiter}

Searches for the specified string. The workspace pointer does not move if the Editor does not find the string.

GET [{number of lines / range of lines} [primary input file / alternate input file [/disc drive]]]

Transfers text from a primary or alternate input file to the workspace. Lines are inserted above the workspace pointer. The pointer does not change. Default parameters: one line, primary input file, system disc.

INPUT

Allows you to enter text into the workspace from the system console. Follow this command with a carriage return; then enter text. Text is inserted above the workspace pointer. To exit input mode, enter two carriage returns or the CTRL Z function.

INSERT {text line}

Allows you to enter a line of text into the workspace from the system console. Follow this command with a space, the desired line of text, and a carriage return. Text is inserted above the workspace pointer.

KILL [number of lines / range of lines]

Deletes lines of text from the workspace. The workspace pointer moves to the line following the last line deleted. Default parameter: one line.

LIST [number of lines / range of lines]

Displays lines of text on the line printer (LPT1). The workspace pointer does not move. Default parameter: one line.

MACRO [number[=[command line]]]

Displays, executes, deletes, or defines a macro. To display the current macros, enter this command without any parameters. To delete a macro, enter Mx= followed by a carriage return, where x represents the macro number. You may define up to 127 macros.

N

Displays the line position of the workspace pointer.

PUT [{number of lines / range of lines} [output device / alternate output file [/disc drive]]]

Transfers text from the workspace to a file or device. If you do not specify an output device or file, the Editor appends text to the primary output file. Default parameters: one line, system disc.

PUTK [{number of lines / range of lines} [output device / alternate output file [/disc drive]]]

Transfers text from the workspace to a file or device. Deletes the text transferred from the workspace. If you do not specify an output device or file, the Editor appends text to the primary output file. Default parameters: one line, system disc.

QUIT

Terminates the Editor and returns control to TEKDOS. To terminate the Editor and save text, enter the FILE command.

REPLACE {text line}

Replaces the current line with the specified text. The workspace pointer does not move.

SUBSTITUTE {$old string$new string$}

Replaces a character string with a new character string. The delimiters may not be spaces. The workspace pointer does not move.

SUSPEND

Suspends the Editor and returns control to TEKDOS. Same effect as pressing the ESC key twice. To return control to the Editor, use the CONT command.

TAB {character}

Defines the tab character. The character defined may not be a colon, comma, space, or right or left arrow. The tab character is not entered into the workspace.

TABS {column} [column] ...

Assigns tab positions. Column numbers must be in ascending order. Tab positions default to 8, 16, 24, 32, 40, 48, 56, 64.

TYPE [number of lines / range of lines]

Displays lines of text on the system console (CONO). The workspace pointer does not move. Default parameter: one line.

UP [number of lines]

Moves the workspace pointer up the specified number of lines and displays the line moved to. To suppress the display, follow this command with a period. Default parameter: one line.

?

Display the Editor I/O status.

Section 4
SERVICE CALLS

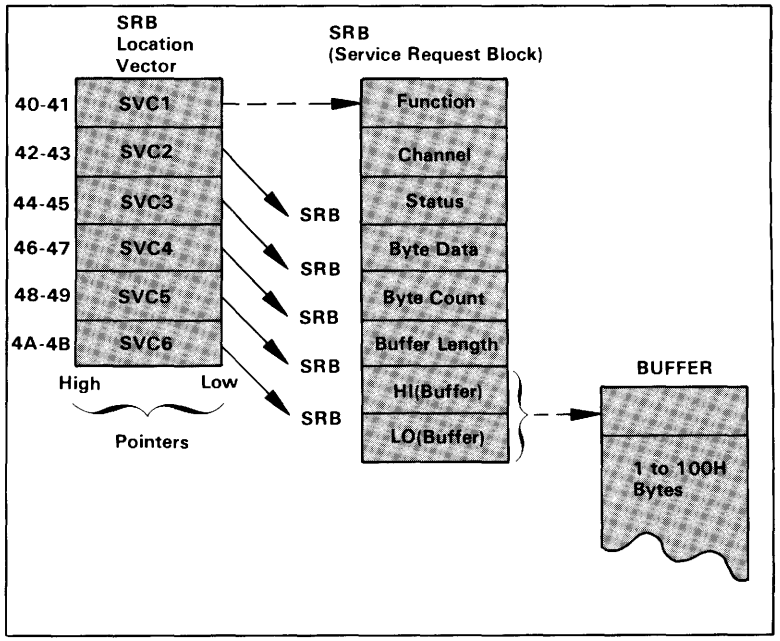


Fig. 4-1. Service call memory layout.

The service call within your program determines which of the six SRB pointers will be used. The SRB addressed by the selected pointer contains the necessary parameters passed for the SVC. Depending upon the function specified in the first byte of the SRB, an I/O buffer may be needed. In this case, the length and the location of the buffer are indicated in the last three bytes of the SRB.

SVC FUNCTION CODE DESCRIPTIONS

01—Read ASCII and wait. Suspends your program until TEKDOS reads a line of characters and a carriage return from the device assigned to the selected channel. Requires the channel number, buffer length, and buffer pointer. Returns the number of characters read in Byte Count and the characters read in the Buffer.

02—Write ASCII and wait. Suspends your program until TEKDOS writes a line of characters and a carriage return to the device assigned to the selected channel. Requires the channel number, buffer length, buffer pointer and the Buffer (containing the line to be written and a carriage return). Returns the number of characters written in Byte Count.

03—Close Channel. Closes an assigned channel. Channels used for output must be closed before your program terminates, to ensure that data is not lost. Requires the channel number.

04—Rewind file. Positions the pointer to the beginning of a disc file. Ignored if the selected channel is not assigned to a disc file. Requires the channel number.

05—Delete file. Deletes a disc file. Ignored if the selected channel is not assigned to a disc file. Requires the channel number.

06—Rename file. Renames a disc file. Ignored if the selected channel is not assigned to a disc file. Requires the channel number, buffer length, buffer pointer, and the Buffer (containing the new file name and a carriage return).

10—Assign device or file. Assigns a channel to a device or file. Must be performed before a read or write SVC may be executed. Requires the channel number, buffer length, buffer pointer, and Buffer (containing the device or disc file name, and a carriage return).

11—Read program clock. Reads the program clock. Enter the CLOCK ON command before executing. Returns the high-order clock byte in Byte Data and the low-order clock byte in Byte Count.

13—Get command file invocation line. Retrieves one of the parameters entered in a command file invocation line. Requires the parameter number, buffer length, and buffer pointer. Returns the Buffer (containing a command file parameter and a carriage return) and the parameter length in Byte Count.

14—Get device identification and type. Requires the channel number assigned to the device. Returns the device identification in Byte Data and the device type in Byte Count. See Table 4-1.

Table 4-1
Device Identification and Type Code

I.D.	TYPE CODE	DESCRIPTION
01	01	ASCII read from CONI (Console input)
02	02	ASCII write to CONO (Console output)
03	02	ASCII write to LPT1 (Line printer)
06	01	ASCII read from TTYR (TTY reader)
08	01	ASCII read from PPTR (Paper tape reader)
09	02	ASCII write to PPTP (Paper tape punch)
0A	01	ASCII read from REMI (Remote input)
0B	02	ASCII write to REMO (Remote output)
FF	43	Binary read/write from disc (disc file name)

15—Get device status. Requires the channel number assigned to the device. Returns the hardware status byte of a device assigned to the channel. The channel number is indicated in Byte Data.

16—Get last CONI character. Returns the last character received from the system console in Byte Data.

17—Load overlay. Loads an overlay module into program memory. Requires the buffer length, buffer pointer, and the Buffer (containing the overlay file name and a carriage return).

18—Load overlay and execute. Loads and executes an overlay module. Requires and returns the same parameters as function code 17.

19—Suspend program. Suspends your program. Continue the program with the CONT command.

1A—Exit program. Terminates your program. Continue the program with the GO command. Does not close the channels. Terminates the debug system.

1C—Get execution line parameter. Requires the parameter number, buffer length, and buffer pointer. Returns a parameter issued with the XEQ command line in the Buffer and the parameter length in Byte Count.

1F—Abort program. Terminates your program. Closes all channels. Terminates the debug system.

21—Read without echo from CONI. Requires the channel number assigned to CONI, the buffer length, and the buffer pointer. Returns the line read from the system console in the Buffer and the number of bytes read in Byte Count. Does not echo the characters. Only used for channels assigned to CONI.

41—Read binary and wait. Suspends your program until TEKDOS reads data from the selected channel. Does not strip parity. Requires the channel number, buffer length, and buffer pointer. Returns the number of bytes transferred in Byte Count and the bytes transferred in the Buffer.

42—Write binary and wait. Suspends your program until the Buffer contents are written to the device assigned to the selected channel. Does not strip parity. Requires the channel number, buffer length, buffer pointer, and Buffer (containing the data—a carriage return is not necessary). Returns the number of bytes written in Byte Count.

57—Load overlay with bias. Load an overlay into program memory with the offset specified in Byte Data and Byte Count. Also requires the buffer length, buffer pointer, and Buffer (containing the overlay file name).

81—Read ASCII and proceed. Reads ASCII characters from a device assigned to the selected channel. TEKDOS returns control to your program before the transfer is completed. Requires the channel number, buffer length, and buffer pointer. Returns the number of characters read in Byte Count and the characters read in the Buffer.

82—Write ASCII and proceed. Writes ASCII characters to a device assigned to a selected channel. TEKDOS returns control to your program before the transfer is complete. Requires the channel number, buffer length, buffer pointer, and Buffer (containing the line to be written and a carriage return). Returns the number of characters transferred in Byte Count.

C1—Read binary and proceed. Similar to function code 41, but uses the proceed function as described in function codes 81 and 82. Requires the channel number, buffer length, and buffer pointer. Returns the number of bytes read in Byte Count and the bytes read in the Buffer.

C2—Write binary and proceed. Similar to function code 42, but uses the proceed function as described in function codes 81 and 82. Requires the channel number, buffer length, buffer pointer, and Buffer (containing the data—a carriage return is not necessary). Returns the number of bytes transferred in Byte Count.

Section 5
COMMAND FILES

Description

A command file is a file containing TEKDOS commands. A command file may contain comment lines for documentation purposes. All comment lines must begin with an asterisk followed by a space. COPYSYS and NEWDISC are examples of command files.

Invocation

To invoke a command file, enter the command file name, the parameters, and a carriage return. If the command file does not reside on the system disc, immediately follow the command file name with a slash (/) and the disc drive number.

One command file may invoke another. When this occurs, TEKDOS terminates the first command file and passes control to the next one.

Parameter Substitution

You may pass parameters when you invoke the command file. To do so, enter these parameters after the command file name and a space. Each parameter in the command file invocation line is assigned a one-digit decimal identifier.

Parameters may be referenced within the command file with the one-digit identifier preceded by the dollar sign ($). TEKDOS substitutes this two-character sequence with the actual parameters passed during command file invocation. TEKDOS identifies the first parameter with $1, the second parameter with $2, and so on. To access the command file you invoked, use $0.

Command File Control Commands

ABORT command name	Terminates system program execution.
ABORT program name	Terminates one of your command files or programs.
ABORT *	Terminates all executing programs.
ABORT /	Terminates all of your command files or programs. Closes all assigned channels.

CONT command name	Continues a suspended command file
CONT program name	Continues a suspended system program
CONT *	Continues all suspended programs
CONT /	Continues all suspended user programs
KILL ON	Terminates command file execution whenever one of the commands within the command file generates an error.
KILL OFF	Ignores an invalid command in a command file, generates the appropriate error message, and continues command file execution.
SUSP command name	Suspends system program execution.
SUSP program name	Suspends one of your command files or programs.
SUSPEND *	Suspends all executing programs.
SUSPEND /	Suspends all of your executing command files or programs.
TYPE ON	Displays commands in a command file as they execute.
TYPE OFF	Suppresses display of commands in a command file.

Section 6
INTERSYSTEM COMMUNICATION

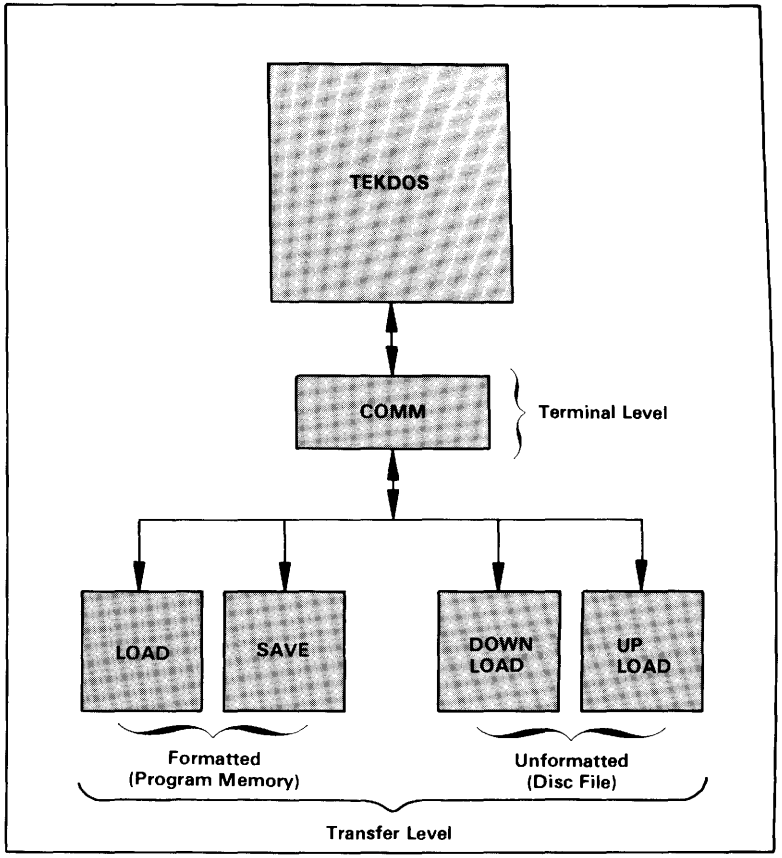


Fig. 6-1. COMM block diagram.

This figure illustrates the various modes of data transfer available with the COMM command and shows the two types of transfer levels.

ENTERING COMM

The 8002A responds like a terminal to the external computer after you enter the COMM command line. This mode of operation is called terminal level. Each external computer requires a certain set of COMM parameters to communicate with the 8002A. Space is provided below to write in the parameters that apply to your external computer. To return control to TEKDOS from terminal level, enter the null character and press the ESC key.

Table 6-1
COMM Parameter Checklist

Parameter	Value
E=	
L=	
P=	
T=	
M=	
C=	

UNFORMATTED TRANSFER

Download Command Line

To transfer text from the external computer to the 8002A, perform the following steps:

1. Enter the external computer command that transfers a file to the system console.
2. Enter the file name you desire to transfer.
3. Enter the null character.
4. Press the right arrow key (>).
5. Enter the name of the 8002A file name where the data will be transferred.
6. Enter a carriage return.

After the data transfer is completed, the 8002A returns to terminal level.

Upload Command Line

To transfer text from the 8002A to an external computer, perform the following steps:

1. Enter the external computer command that transfers a file to the system console.
2. Enter the null character.
3. Press the left arrow key (<).
4. Enter the 8002A file name that contains the data to be transferred.
5. Enter a carriage return.

After the data transfer is complete, the 8002A returns to terminal level.

FORMATTED TRANSFER

Load Mode Command Line

To transfer Tekhex object code files to the 8002A, perform the following steps:

1. Enter the name of the external computer program that downloads a Tekhex file.
2. Enter the null character.
3. Enter a carriage return.

Any of the following conditions returns the 8002A to terminal level from Load mode:

- You press the BREAK key on the system console.
- The 8002A receives a Terminating Block from the external computer.
- The 8002A receives an Abort Block from the external computer.
- An attempt is made to load beyond 8002A program memory location FFFF.
- A memory read or write error is generated in the 8002A.
- Downloading is completed.

Save Mode Command Line

To transfer Tekhex object code from the 8002A to the external computer, perform the following steps:

1. Enter the name of the external computer program that uploads Tekhex object code.
2. Enter the null character without a space following it.
3. Enter the program memory address where the code starts. Enter a space.
4. Enter the program memory address where the code ends. Enter another space.
5. Optionally, enter the program memory address where code begins execution.
6. Enter a carriage return.

Any of the following conditions returns the 8002A to terminal level from Save mode:

- You enter an invalid address or address range.
- You press the BREAK key on the system console.
- A memory read or write error is generated in the 8002A.
- The 8002A receives an Abort Block from the external computer.
- Error code 80, 82, or 83 occurs and the C=T parameter has been specified.
- Uploading is completed.

STATUS MODE

This mode displays the contents of the Remote Port Status Register. To obtain this information, perform the following steps:

1. Enter the null character.
2. Press the S key.
3. Enter a carriage return.

The system console displays the register contents as two hexadecimal digits. For a description of these digits, refer to Table 8-3 in the 8002A System User's Manual.

Section 7
TABLES

ASCII Code Conversion Table

		HEXADECIMAL MOST SIGNIFICANT CHARACTER							
	—	Ø	1	2	3	4	5	6	7
	Ø	NUL	DLE	SP	Ø	@	P	`	p
	1	SOH	DC1	!	1	A	Q	a	q
	2	STX	DC2	"	2	B	R	b	r
	3	ETX	DC3	#	3	C	S	c	s
	4	EOT	DC4	$	4	D	T	d	t
LEAST	5	ENQ	NAK	%	5	E	U	e	u
SIGNIFICANT	6	ACK	SYN	&	6	F	V	f	v
CHARACTER	7	BEL	ETB	'	7	G	W	g	w
	8	BS	CAN	(	8	H	X	h	x
	9	HT	EM	)	9	I	Y	i	y
	A	LF	SUB	*	:	J	Z	j	z
	B	VT	ESC	+	;	K	[	k	{
	C	FF	FS	,	<	L	\	l	|
	D	CR	GS	-	=	M	]	m	}
	E	SO	RS	.	>	N	^	n	~
	F	SI	US	/	?	O	—	o	DEL

EXAMPLES

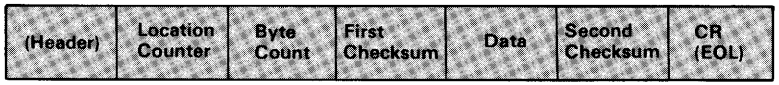
W = 57
H = 48

TEKTRONIX HEXADECIMAL FORMAT

General Description

- The message block contains a slash (/) as the Header Character.
- A carriage return (EOL) terminates the message block.
- A message block is limited to 72 characters.
- All characters must belong to the printable ASCII character set, except EOL.
- A checksum represents the sum of the hexadecimal digits. The checksum is a two-digit number.

Message Block Format

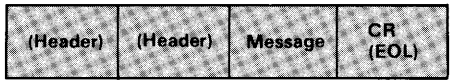


- Location Counter: four digits that represent the starting address of the block in program memory.
- Byte Count: two digits that represent the number of data bytes in the block.
- First Checksum: two digits that represent the sum, modulo 256, of the six digits that make up the Location Counter and Byte Count.
- Data: N bytes, each represented by two digits (maximum of thirty).
- Second Checksum: two digits that represent the sum, modulo 256, of the digits that make up the N data bytes.

Terminating Message Block

- Transfer Address: four digits that represent the program memory address of the first data byte in the record.
- Byte Count: set to zero, indicating a Terminating Block.
- Checksum: two digits that represent the sum, modulo 256, of the six digits that make up the Transfer Address and Byte Count.

Abort Block



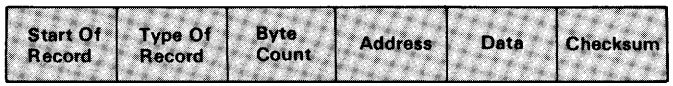
- Error Information: a programmer-defined message string of ASCII characters.

SMS Tape Format

- A TAPE ON character (CTRL R) precedes each SMS data block and resets the address to zero.
- The data block consists of up to 512 bytes.
- A TAPE OFF character (CTRL T) terminates the data block.
- One or two characters represent each data word.
- An apostrophe follows each data word.
- All characters are punched in the standard 8-channel ASCII teletype code.
- Parity is not checked.

Motorola Load Module Format

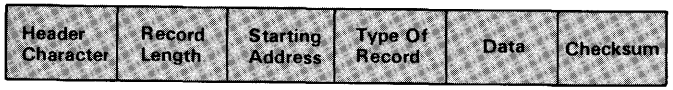
Format Description



- Start of Record: always an S.
- Type of Record: 0 (header record), 1 (data record), or 9 (end-of-file record).
- Byte Count: two digits that specify the number of data bytes in the record, including checksum and address data.
- Address: four digits that represent the program memory address where the record is stored.
- Data: N data bytes, each represented by two digits.
- Checksum: Two digits that represent the one's complement, of the sum modulo 256, of the record type, data bytes, address, and byte count.

Intel Load Module Format

Format Description



- Header Character: always a colon.
- Record Length: two digits that represent the number of data bytes. Record length of 0 indicates the last file record.
- Starting Address: four digits that represent the program memory address of the first data byte in the record.
- Type of Record: 00 = normal record. 01 = last file record.
- Data: N data bytes, each represented by two digits.
- Checksum: two digits that represent the two's complement of the sum, modulo 256, of the preceding bytes.

Section 8
ERROR EXPLANATIONS

TEKDOS Error Codes

01—Directory read error. The Flexible Disc controller cannot read the disc. The disc may not be formatted (see the FORMAT command). The disc may have a damaged sector. A file may have an invalid header. The disc drive door may be open. The disc may be inserted incorrectly. The specified drive number may not be connected.

02—Directory write error. The Flexible Disc controller cannot transfer data to the disc. Make sure the write-enable tab covers the write-protect notch on the disc. The disc drive door may be open. The disc may be inserted incorrectly. The disc may have a damaged directory sector. The disc may not be formatted.

03—Command file not found. TEKDOS does not recognize the command. The command entered does not exist and does not represent the name of a command file. If it is a debug command, make sure you have invoked the debug system (via the DEBUG command). If it is a command file, make sure you have spelled the name correctly. If the command file does not reside on the system disc, make sure you have specified the disc drive number.

04—Command file input error. The file name used for command file invocation does not represent an ASCII file. Make sure your command file is not a load or fetch file.

05—Procedure busy. A command file is executing. Two command files cannot run at the same time.

06—Device read error. The SVC has read fewer bytes than requested by the SRB on a binary read. No return character is in the buffer on the ASCII read. For ASCII reads, make sure the file you are reading ends with a carriage return.

07—Device write error or end-of-device. The file written does not end with a carriage return character. The disc cannot hold any more data. Check your directory listing. The disc cannot contain more than 78 files or more than 304 data blocks. (Some disc sectors are reserved for TEKDOS.)

08—Drive not specified. You did not specify a disc drive number with the first parameter of the RENAME command.

09—Invalid drive. The drive number or file name specified with a command is invalid. File names may not exceed eight characters. File names must begin with a letter. The rest of the characters may be alphanumeric, or consist of the following special characters:

? " # & % ( ) ; = ! * @

10—Overlay load failure. The file you tried to load is not a fetch file. One of your modules, or a system module, may have been deleted, altered, or destroyed. Use the MODULE command to recreate the fetch file if this error occurs when trying to fetch one of your modules. If this error occurs after you enter a TEKDOS command, either create a new system disc, or find the damaged overlay and use the GET and REPLACE commands to reconstruct it.

11—Overlay area in use. An attempt has been made to concurrently execute two commands that reside in the same system memory or program memory area. Use the ABORT command to free the memory areas.

12—Invalid file name. The specified file name does not begin with a letter or has too many characters. File names may not exceed eight characters. File names must begin with a letter. The rest of the characters may be alphanumeric, or consist of the following special characters:

? " # & % ( ) ; = ! * @

13—Input file not found. The specified file name does not exist. If the file does not reside on the system disc, you must enter the disc drive parameter.

14—Invalid input device. An attempt has been made to assign an output device for input.

15—Invalid output device. An attempt has been made to assign an input device for output.

16—Input device assign failure. The specified input device is not operational. Check connections, plugs, and ON-OFF switches.

17—Output device assign failure. The specified output device is not operational. Error code 63 may also occur under this condition. Check connections, plugs, and ON-OFF switches.

18—Device in use. The device specified in an ASSIGN command has already been assigned to a channel. CONO, CONI, and file names may each be assigned to more than one channel. All other devices are limited to one channel assignment at a time. This error may also occur if you have made eight channel assignments and two or three commands are executing, and if a command file has been activated. Close some of the channels (with the CLOSE command) and use the ABORT command to free the system overlay areas.

19—Invalid channel number. The channel number specified in the ASSIGN or CLOSE command line, or specified in the SRB, is not in the range 0-7 inclusive.

20—Channel in use. The specified channel has already been assigned. A channel may have only one device at a time assigned to it. Close the channel (with the CLOSE command) and reassign it (with the ASSIGN command). Display channel assignments with the STATUS command.

21—Channel assign failure. A device assigned to a channel cannot be assigned to another channel (except CONO, CONI, and file names). Close the channel (with the CLOSE command) and reassign it (with the ASSIGN command). Display channel assignments with the STATUS command.

22—Command line buffer overflow. The command line entered is longer than one system console display line. The resulting line, after all command file parameter substitution, is too long.

23—Invalid command. This error occurs when you try to change emulation modes while programs are running or suspended.

24—Job not active. An attempt has been made to abort a non-active program. Check the status of programs with the STATUS command.

25—Job not suspended. An attempt has been made to continue a program that is executing. Check the status of programs with the STATUS command.

26—Job already suspended. An attempt has been made to suspend a program that has already been suspended. Display the status of all programs with the STATUS command.

27—Job executing. The program is already executing.

28—Job under debug control. The program is already under debug control.

29—PROM power failure. PROM commands will not execute because the PROM POWER switch is turned off.

30—Invalid parameter. The system does not recognize one of the parameters entered in a command line. Check the required format and parameters in the Command Dictionary. The 9900 Emulator Processor requires two-byte data entry. Check for odd-numbered field length.

31—Parameter required. A command line that requires another parameter has been entered. Check the required parameters in the Command Dictionary.

32—Too many parameters. The system cannot use one or more of the specified parameters. Check the required format in the Command Dictionary.

33—Bias parameter error. The offset specified with the LOAD, RHEX, or RVHEX command line is illegal. The offset must be a hexadecimal number in the range 0-FFFF inclusive, preceded by a slash.

34—Invalid address. An illegal address parameter has been entered. Only hexadecimal numbers are legal. Make sure the address is less than or equal to FFFF.

35—Invalid start address. An illegal address parameter has been entered. Only hexadecimal numbers are legal. Make sure the address is less than or equal to FFFF.

36—Invalid end address. The end address parameter is illegal. Make sure the address is a hexadecimal number in the range of 0-FFFF and greater than or equal to the start address parameter.

37—Invalid GO address. An illegal address parameter has been entered with the GO command. Only hexadecimal numbers are legal. Make sure the address is less than or equal to FFFF.

39—Invalid hex character. The ESC key has been pressed while using the EXAM command to modify memory, and the last memory byte was not altered. Re-enter the command. The system cannot convert characters received from a TTYR device. Check your TTYR. Check the parity requirements for TEKDOS. Refer to the 8001/8002A Installation Guide.

40—Invalid RHEX input format. The RHEX checksums do not match, indicating that the code block is incorrect. Check the code.

41—Invalid breakpoint access mode. An attempt has been made to set a third breakpoint. Only two breakpoints may be active at one time. This condition may produce the message, TOO MANY BKPTS.

42—Invalid register parameter. An attempt has been made to alter the contents of a register that does not exist. This condition may also generate error code 30.

43—Invalid data parameter. An attempt has been made to set a register to an illegal value. Only hexadecimal numbers are legal.

44—Invalid TRACE mode parameter. An attempt has been made to set a fourth TRACE command. Only three TRACE commands may be active at one time. Check the Command Dictionary for correct parameters.

45—Invalid Emulator Processor SRB address. The SRB address is greater than 16K.

46—Emulator halted. An Emulator Processor HALT instruction occurred while a program was executing outside of the debug system. Error code 68 may also occur under this condition. Make sure the appropriate Emulator Processor module is plugged in. Make sure the Assembler Processor module is plugged in.

47—System area bad. The disc space reserved for TEKDOS has been damaged. Save your files on another disc and re-format and verify the disc in question. If the problem persists, use another disc.

48—Fetch file not found. The system cannot find the fetch file required to execute the command line entered. One of the SOF members may be deleted, incorrectly or inadvertently altered, or missing. You may need to create a new system disc. If this error occurs when trying to fetch one of your modules, re-create the file with the MODULE command.

49—Fetch file assign failure. An attempt has been made to fetch a file with an illegal name.

50—File not a fetch module. A file name parameter that has been entered with the FETCH or XEQ command line does not represent a file created with the MODULE command.

51—Invalid fetch request. An attempt has been made to fetch a file while another program is executing. Use the ABORT command to end the execution of all programs and free all system memory areas.

54—Invalid mode. The system does not recognize the emulation mode parameter. Valid parameters are 0, 1, and 2. An attempt may have been made to use the EDIT, ASM, or LINK command while in emulation mode 1 or 2.

55—Invalid memory. The specified memory is non-existent or unavailable. This condition may also generate error code 30 or 59. Check the Program Memory module jumpers.

57—File name in use. The file name entered with the RENAME command line already exists. Check your directory listing.

58—Device assign failure. The system cannot read from or write to a device. Check connections, cables, and ON-OFF switches, fuses, and correct implementation of the RS-232-C control lines between the device and the 8002A. Refer to the 8001/8002A Installation Guide.

59—Memory write error. The specified memory is non-existent or unavailable. Error code 55 may also occur under this condition. Check all cables and prototype connections. Check the address range of prototype RAM. Check the Program Memory module jumpers.

60—End of media. No more data can be read from or written to a device.

61—File in use. The file name entered in the DELETE command line is in use. Check the status of all programs with the STATUS command.

61—Device not operational. The system has tried to read from or write to a device that is not turned ON. The PROM Programmer module may not be in the 8002A mainframe.

63—Directory full. The disc contains 78 files, and an attempt was made to create another file. A disc may contain a maximum of 78 files. Error code 17 may also occur under this condition.

64—Invalid disc. The disc read from or written to may not be TEKDOS-compatible. If the disc is TEKDOS-compatible, make sure it has been formatted and verified.

65—System memory parity error. Hardware failure. Toggle the SYSTEM RESTART switch and repeat the last command line. If the error occurs again, a damaged or malfunctioning System Memory module may be causing the problem.

66—Program memory parity error. Hardware failure. Toggle the SYSTEM RESTART switch and repeat the last command line. If the error occurs again, a damaged or malfunctioning Program Memory module may be causing the problem.

67—Emulator clock missing. The Emulator Processor is malfunctioning because one or more clock phases are missing. Check the prototype hardware. Make sure the prototype power switch is ON, and that the prototype clock is functioning correctly.

68—Emulator faulted. The Emulator Processor is malfunctioning. Error code 46 may also occur under this condition. Check the hardware, fuses, and clean the board contacts. Make sure that the prototype power switch is ON, and that the prototype clock is functioning correctly.

69—Address not on word boundary. An attempt has been made to modify program memory data that does not begin on a word boundary, or to modify an odd number of bytes.

70—Word or byte boundary error. An uneven number of characters has been entered in an attempt to modify memory. You cannot modify a half-byte.

80—REMI/REMO port does not have a carrier signal. See the Technical Notes section regarding Intersystem Communication.

82—REMI port parity error. See the Technical Notes section regarding Intersystem Communication.

83—REMI port framing error. See the Technical Notes section regarding Intersystem Communication.

SRB STATUS CODES

00—Function complete. No error.

01—Channel assigned to a new file.

02—Illegal channel number.

03—Channel not assigned.

04—Channel busy.

05—Illegal function code.

06—No return character on ASCII read.

07—No return character on ASCII write.

08—Illegal disc drive number.

09—File in use.

0A—Device not operational.

0B—Device not available.

0C—Device not ready.

0D—Device in use.

0E—Directory read error.

0F—Directory write error.

10—Directory full.

11—Device read error.

12—Device write error.

15—File name in use.

16—Illegal file name.

17—File in read/write progress.

18—Channel already assigned.

19—Incorrect flexible disc.

EDITOR MESSAGES

****ABORTED****
The command line exceeded 127 characters. The whole line is lost.

****AI****
The Editor response referred to an alternate input file.

****AO****
The Editor response referred to an alternate output file.

****ASSIGN PROBLEM****
The Editor was unable to assign a channel to a given device.

BREAK
The ESC key was pressed to terminate execution of a file I/O function.

****COMMAND****
An unknown command was encountered in the command line.

****DISC FULL****
The output disc is full and cannot accept any more text.

****EOF****
The end of a file or of the text in the workspace has been reached.

****(INPUT)****
The Editor response referred to an input attempt.

****(LPT1)****
The Editor response referred to the line printer.

MODE
An attempt was made to execute a macro from another macro. This is not allowed.

NEST
The nested repetition brackets < and > do not match.

****NEW FILE****
A new file was created on the disc.

****NO FILES SPECIFIED****
The Editor was invoked without primary input and primary output files specified. You must specify alternate input and output files when using the GET, PUT, and PUTK commands.

****NO PI****
The Editor was invoked without specifying a primary input file. You must specify an alternate input file when using the GET command.

****NO PO****
The Editor was invoked without specifying a primary output file. You must specify an alternate output file when using the PUT and PUTK commands.

****NOT FOUND****
The specified string could not be found.

****NUMBER****
An invalid number of lines was specified.

****(OUTPUT)****
The Editor response referred to an output attempt.

****PI****
The Editor response referred to the primary input file.

****PI NEW FILE?****
The primary input file specified upon invoking the Editor does not exist on the disc. Control is returned to TEKDOS.

****PO****
The Editor response referred to the primary output file.

****PROCEDURE ERROR****
Editor usage is in error.

****RANGE****
The range of lines specified is not in the workspace or an invalid range parameter was given.

READ FILE
An attempt was made to read from a non-existent or invalid input device.

****TEKDOS STAT=XX****
XX is the TEKDOS SRB status byte returned to the Editor when an unusual request or event has occurred. The meaning of the status byte can be found under SRB Status Codes.

****TRUNCATED****
An INPUT or SUBSTITUTE command line has exceeded 127 characters. An INPUT line is truncated and the characters over 127 are placed on a new line. The SUBSTITUTE line is truncated and the characters over 127 are lost.

****WSP FULL****
The workspace is full. You may delete edit macros to get a little more space.