SECTION 3
THE AIM 65 MONITOR

The Monitor controls AIM 65 operation. The Monitor is a computer program that provides powerful software features and linkages to both AIM 65 and user programs. The Monitor is located in two 4K R2332 ROM's that are installed in sockets Z22 and Z23. An AIM 65 Text Editor is physically included with the Monitor but is described separately. (See Section 4.) The structure of the Monitor and Editor software is described in Section 7.4.

3.1 AIM 65 MONITOR FEATURES

The features of the AIM Monitor include:

- Major function entry and re-entry linkage--easy linkage to and from Editor, Assembler, BASIC, and user functions including initial entry and reentry capabilities. Single keystroke or RESET button depression returns control to the Monitor.

- Display and alter any register--any of the six registers may be displayed and altered.

- Display and alter memory--any memory location may be displayed and altered.

- Instruction mnemonic entry--R6500 machine language instructions may be directly entered into memory from typed mnemonic operation codes and hexadecimal operands.

- Disassemble memory--R6500 object code may be decoded (disassembled) from memory into R6500 mnemonics and hexadecimal operands.

- Selectable RUN/STEP program execution--user programs may be executed in the RUN Mode at full R6502 speed or in STEP Mode for debugging.

- Execution control--user programs can be initiated at specified program counter values. From one to 99 instructions or an indefinite number of instructions may be executed in the STEP Mode. Execution may be terminated at any time with the ESC key.

- Trace--instruction, register, and program counter trace capability exists in the STEP Mode. Either instruction or register trace may be performed during execution. Program counter trace may be performed after execution is terminated.

- Breakpoints--up to four breakpoint addresses may be entered, displayed, and selectively enabled to stop user program execution at specified addresses in the STEP Mode.

- RUN Mode BRK instruction control--BRK instructions may be placed in a user program to stop execution.

- Load and dump memory to and from various peripherals--memory may be loaded from, and dumped to, AIM 65 and user provided I/O devices. AIM 65 peripherals include keyboard, printer, and display. AIM 65 provides hardware and software to directly interface with audio cassette recorders and teletypewriter keyboard, printer, and paper tape reader/punch.

- Verify tape checksum--the record checksum on the audio tape can be checked to verify proper recording.

- User defined interface keys--three keys are dedicated to link directly to user-defined functions with simple return capability to the Monitor.

Table 3-1 lists the Monitor commands by functional grouping.

3.2 MAJOR FUNCTION ENTRY AND EXIT

Five commands are provided to enter other major AIM 65 functions from the Monitor. Four of these commands allow both initial entry and re-entry into the Editor and BASIC. There is only one entry command into the Assembler. An ESC command provides re-entry into the Monitor from most AIM 65 functions. The RESET button always returns control to the Monitor and performs "cold" or "warm" initialization (see Section 1.9).

3.2.1 E Command - Enter and Initialize the Editor

The E command enters and initializes the AIM 65 Text Editor. Refer to Section 4.2.1 for a detailed description.

CAUTION

Be careful not to initialize the Editor before desired information in the Editor Text Buffer has been permanently stored (see Section 4.2.1).

3.2.2 T Command - Re-enter the Editor

The T command re-enters the AIM 65 Text Editor at the top of the existing Text Buffer. Refer to Section 4.2.2 for details.

3.2.3 N Command - Enter Assembler

The N command enters the optional AIM 65 Assembler. Refer to Section 5.4 for a description of assembler command processing. The Monitor enters the assembler by executing a jump to subroutine (JSR) to address $D000. If a user provided function other than the Assembler is programmed in ROM or PROM and is installed in socket Z24, it may be called directly from the Monitor by typing N. Return to the Monitor, if desired, with an RTS.

3.2.4 5 Command - Enter and Initialize BASIC

The 5 command enters the optional AIM 65 BASIC Interpreter. Refer to the AIM 65 BASIC User's Manual for a description of the BASIC commands.

CAUTION

Be careful not to initialize BASIC before any desired BASIC program or data in RAM has been permanently stored.

The Monitor enters BASIC by executing a JSR to address $B000. If a user provided function other than BASIC is installed in socket Z25/Z26, see Section 10.2.

3.2.5 6 Command - Re-enter BASIC

The 6 command re-enters the AIM 65 BASIC Interpreter. Refer to the AIM 65 BASIC User's Manual for a description of the BASIC operation and commands.

The Monitor re-enters BASIC by executing a JSR to address $B003. If a user provided function other than BASIC is installed in socket Z25/Z26, see Section 10.2.

3.2.6 RESET - Enter and Initialize Monitor

The RESET command performs a hardware reset of the peripheral devices and initializes the AIM 65 Monitor.

Perform a "warm" reset by depressing the RESET button..

Perform a "cold" reset by either turning AIM 65 power off, waiting a couple of seconds, and then reapplying AIM 65 power or by changing address $A402 to $00 and then depressing the RESET button.

Example:

```
Press RESET
ROCKWELL AIM 65
```

3.2.7 ESC Command - Re-enter Monitor

The ESC command escapes from the existing command and returns to the Monitor. ESC is operative only in the commands that sample the keyboard. AIM 65 will respond to ESC by displaying the AIM 65 Monitor prompt:

```
<
```

3.3 DISPLAY/ALTER REGISTERS

Seven commands are provided to display or alter the contents of the six registers (program counter, processor status, accumulator, X register, Y register, and stack pointer). The alter commands are used most often to establish initial register values for checkout purposes. During normal program operation, the register contents would be initialized by previously executed instructions.

TABLE 3-1. AIM 65 MONITOR COMMANDS

CATEGORY	COMMAND	FUNCTION
Major Function Entry	RESET	Enter and Initialize Monitor
	ESC	Re-Enter Monitor
	E	Enter and Initialize Text Editor
	T	Re-Enter Text Editor
	N	Enter Assembler
	5	Enter and Initialize BASIC
	6	Re-Enter BASIC
Display/Alter Registers	*	Alter Program Counter
	P	Alter Processor Status
	A	Alter Accumulator
	X	Alter X Register
	Y	Alter Y Register
	S	Alter Stack Pointer
	R	Display Registers
Display/Alter Memory	M	Display Specified Memory Contents
	SPACE	Display Next Four Memory Contents
	/	Alter Memory Contents
Instruction Entry/ Disassembly	I	Instruction Mnemonic Entry
	K	Disassemble Memory

TABLE 3-1. AIM 65 MONITOR COMMANDS (Cont.)

CATEGORY	COMMAND	FUNCTION
Execution/Trace	G	Start Execution at Program Counter Address
	Z	Toggle Instruction Trace Mode On/Off
	V	Toggle Register Trace Mode On/Off
	H	Trace Program Counter History
Manipulate Breakpoints	?	Display Breakpoints
	#	Clear All Breakpoints
	B	Set/Clear Breakpoints
	4	Toggle Breakpoint Enable On/Off
Load/Dump Memory	L	Load Memory
	D	Dump Memory
Peripheral Control	CTRL PRINT	Toggle Printer On/Off
	PRINT	Print Display Contents
	LF	Line Feed
	1	Toggle Tape 1 Control On/Off
	2	Toggle Tape 2 Control On/Off
	3	Verify Tape Checksum
User Function Interface	F1	User Function 1
	F2	User Function 2
	F3	User Function 3

3.3.1 * Command - Alter Program Counter

The * command changes the value of the program counter.

Use the * command as follows:

1. Type SHIFT and * simultaneously. AIM 65 will respond with:

   <*>= ^

2. Enter the new hexadecimal value of the program counter. End the input with RETURN or a SPACE.

   Example:

   <*>=0300

In the example above, the program counter was changed to $0300. The instruction in memory location $0300 will be executed first when the G command (Start Execution at Program Counter Address) is entered.

3.3.2 P Command - Alter Processor Status

The P command alters the contents of the processor status register.

To alter the processor status register, type P. AIM 65 will respond with:

<P>= ^

Enter the new value of the processor status register as a two digit hexadecimal number. A leading zero must be entered in the left digit position if the left digit value is zero.

Example:

```
<P>=00
```

In the above example, the value of the processor status register was changed to $00.

3.3.3 A Command - Alter Accumulator

The A command alters the contents of the accumulator.

To alter the accumulator register, type A. AIM 65 will respond with:

```
<A>= ^
```

Enter the new value of the accumulator register as a two digit hexadecimal number. A leading zero must be entered in the left digit if the left digit value is zero.

Example:

```
<A>=01
```

In the above example, the value of A was changed to $01.

3.3.4 X Command - Alter X Register

The X command alters the contents of the X register.

To alter the X register, type X. AIM 65 will respond with:

```
<X>= ^
```

Enter the new value of the X register as a two digit hexadecimal number. A leading zero must be entered in the left digit if the left digit value is zero.

Example:

```
<X>=02
```

In the above example, the value of the X register was changed to $02.

3.3.5 Y Command - Alter Y Register

The Y command alters the contents of the Y register.

To alter the Y register, type Y =. AIM 65 will respond with:

```
<Y>=
```

Enter the new value of the Y register as a two digit hexadecimal number. A leading zero must be entered in the left digit if the left digit value is zero.

Example:

```
<Y>=03
```

In the above example, the value of the Y register was changed to $03.

3.3.6 S Command - Alter Stack Pointer

The S command alters the value of the stack pointer.

To alter the value of the stack pointer, type S. AIM 65 will respond with:

```
<S>=
```

Enter the new value of the stack pointer as a two digit hexadecimal number. A leading zero must be entered in the left digit if the left digit value is zero.

Example:

```
<S>=FF
```

In the above example, the value of the stack pointer was changed to $FF. Note that the stack is always in page one of memory, so the address of the stack is therefore $01FF.

3.3.7 R Command - Display Register Contents

The R command is used to display the current contents of the six registers.

To display the contents of the registers, type R. AIM 65 will print two lines. The first line shows the symbols for the registers and the second line shows the actual contents. The registers and their corresponding symbols are:

Program counter	****
Processor status	PS
Accumulator	AA
X register	XX
Y register	YY
Stack pointer	SS

Example:

```
<R>
 **** PS AA XX YY SS
 0300 00 01 02 03 FF
```

In the above example, the registers and their contents are:

Program counter	(****)	= $0300
Processor status	(PS)	= $00
Accumulator	(AA)	= $01
X register	(XX)	= $02
Y register	(YY)	= $03
Stack pointer	(SS)	= $FF (which means that the stack is at address $01FF since it is always in page one.)

The R command also provides column headings for reference or when the register trace or breakpoints are being used.

3.4 DISPLAY/ALTER MEMORY

Three commands are provided to display or alter memory. The memory addressed may be used for program (instructions), data, or I/O.

3.4.1 M Command - Display Specified Memory Contents

The M command displays the hexadecimal contents of four consecutive memory locations, starting at the specified address.

Use the M command as follows:

1. Type M. AIM 65 will respond with:

   <M>=

2. Enter the hexadecimal address of the first of the four memory locations to be displayed. If the hexadecimal address is less than four digits long, end the input with RETURN or SPACE.

3. AIM 65 will display the contents of the four memory locations.

   Example:

   <M>=0300 EA AD 00 A2

In the above example, the memory locations and their contents are:

ADDRESS	CONTENTS
0300	EA
0301	AD
0302	00
0303	A2

Uninstalled memory will respond with a value equal to the two high order digits of the address.

Example:

<M>=1000 10 10 10 10

3.4.2 SPACE Command - Display Next Four Memory Contents

The SPACE command displays the contents of the next four memory locations, after the initial address value has been entered using the M Command. Use the SPACE command as follows:

1. Use the M command to display the first four memory locations.

2. Type SPACE. AIM 65 will display the contents of the next four memory locations.

After the initial use of the M Command, the SPACE Command may be used any number of times.

NOTE

If the M command is not used first to initialize the starting memory location, a random starting memory location will appear.

3.4.3 / Command - Alter Memory Contents

The / command alters any memory location displayed with the M command or the SPACE command.

Use the / command as follows:

1. Display the memory location to be altered using M command or SPACE command.

2. Type /.

3. AIM 65 will respond with the address of the first memory location that was displayed on the previous line.

4. If the first memory location is to be altered, enter the new contents in hexadecimal. If the location is to be left as is, type one SPACE.

5. Proceed to the next location and alter it, if needed.

6. When the altering of the locations displayed is

complete, type RETURN. If the last memory location on the line was altered, no RETURN is necessary.

7. To alter the next four locations, re-enter the command /.

Example:

```
<M>=0300 EA A9 00 A2
</> 0300 0F    27
```

In the above example, the following operations were performed:

Location 0300 was changed to $0F.
Location 0301 was left unchanged (one SPACE was entered).
Location 0302 was changed to $27.
Location 0303 was unchanged (RETURN was entered after location 0302 was changed).

If an attempt is made to alter uninstalled, protected, write-only address, or failed memory, AIM 65 will display a MEM FAIL message along with the address that caused the error.

Example:

```
<M>=1000 10 10 10 10
</> 1000 30
 MEM FAIL 1000
```

3.5 INSTRUCTION ENTRY/DISASSEMBLY

Two commands allow easy entry of R6500 instructions into memory and examination of instructions already in memory.

The I command encodes (or assembles) symbolic instructions entered on the keyboard into directly executable object code stored in memory. The K command decodes (or disassembles) object code from memory into symbolic instructions for user examination.

3.5.1 I Command - Instruction Mnemonic Entry

The I command enters R6500 instructions directly into memory as object code from symbolic instructions entered from the keyboard. Starting from a user entered address, operation codes (op codes) are entered using three-digit alphabetic abbreviations. Operands, if required, are entered in hexadecimal in accordance with the addressing mode formats. Invalid op codes and operands are ignored but cause an ERROR messsage to be displayed.

Use the I command as follows:

1. Type I. AIM 65 will respond with the current program counter address:

```
<I>
XXXX
```

2. The program counter address can be changed by typing * followed by a four-digit hexadecimal address. If address 0300 is entered, AIM 65 will respond with:

```
XXXX   *=0300
0300
```


3. Enter the three-digit alphabetic abbreviation of the operation code. An input error in either of the first two digits may be corrected by typing DEL and the correct digit.

   If the entered op code does not require an operand, the object code is computed, stored in memory, and displayed in object code form along with the program counter address and the symbolic op code. The program counter is incremented by one. If you want to enter additional instructions in successive addresses, return to Step 3. If instruction entry is complete, return to the Monitor by typing ESC.

   If the op code requires an operand, continue to Step 4.

   If the op code is invalid, an ERROR message will appear. The correct op code may then be re-entered without altering the program counter address since it has not been incremented.

   If a valid but undesired op code was entered, it may be corrected in one of two ways:

   A. If the op code requires an operand, enter RETURN before entering an operand or deliberately enter an invalid operand. An ERROR message will be generated and the whole instruction can be re-entered since the program counter address was not changed.

B. If the op code does not require an operand, the object code was entered into memory and the program counter incremented. In this case, re-establish the previous program counter address as in Step 2.

4. Enter the operand in hexadecimal in accordance with the addressing mode formats. Refer to the R6500 Programming Manual for a complete description of the addressing formats. In some cases, a short form is allowed. The display, however, shows the standard form except for conditional branch instructions, which show the absolute address rather than the relative address. The form for operand entry in the appropriate address mode is shown below (where H is the hexadecimal data):

ADDRESSING MODE	OPERAND FORMAT	NOTES
Accumulator	A	
Immediate	#HH	
Zero Page	HH	
Zero Page, X	HH,X or HHX	
Zero Page, Y	HH,Y or HHY	
Absolute	HHHH	
Absolute, X	HHHH,X or HHHHX	
Absolute, Y	HHHH,Y or HHHHY	
Relative	HH or HHHH	(4)
(Indirect, X)	(HH,X) or (HHX) or (HH,X or (HHX	
(Indirect), Y	(HH),Y or (HH) Y	
(Indirect)	(HHHH)	

NOTES

1. Immediate, page zero, or relative addresses require the entry of two digits (HH).

2. Absolute addresses require the entry of four digits (HHHH).

3. The $ symbol preceeding hexadecimal digits is not permitted since all entries are defined as hexadecimal.

4. For conditional branches, the displacement from the program counter may be entered as a two-digit relative address or as a four-digit absolute address, in which case the correct value of the displacement is automatically computed.

End the operand entry with RETURN or SPACE. The op code and operand are computed and stored in memory. The program counter address, the op code object code, and the symbolic form of the op code and operand are displayed. If SPACE was used,a second line is displayed. This line contains the program counter and the object code form of both the op code and the operand.

If the operand is invalid, an ERROR message will be generated and the entire instruction must be re-entered.

An error in operand entry before RETURN or SPACE is entered may be corrected by entering DEL and re-entering the correct

data. An error in operand entry after RETURN or SPACE is entered may be corrected by typing ESC, re-entering the I Command, re-establishing the correct program counter address, and re-entering the complete instruction.

When entering additional instructions, return to Step 2. If instruction entry is complete, return to the Monitor by typing ESC.

Example:

```
<I>
 0200     *=0300
 0300 EA NOP
 0301 A2 LDX #FE
 0303 E8 INX
 0304 D0 BNE 0303
 0306 4C JMP 0310
 0309     *=0310
 0310 A0 LDY #02
 0312 88 DEY
 0313 D0 BNE 0312
 0315 4C JMP 0301
 0318
```

3.5.2 K Command - Disassemble Memory

The K command disassembles object code from memory into symbolic R6500 instructions. Starting from a specified address, each byte of memory is disassembled until a valid op code is decoded. Once a valid op code is found, the appropriate number of following bytes are disassembled to

determine and display the instruction operand. Invalid op codes are indicated by question marks. Refer to Appendix K for a list of valid instructions.

Use the K Command as follows:

1. Type K. AIM 65 will respond with:

   ```
   <K>*=
   ```

2. Enter the starting address in hexadecimal, then type RETURN. If 0300 was entered, AIM 65 will respond with:

   ```
   <K>*=0300
   /
   ```

3. Specify the number of instructions to disassemble by entering a decimal count from 01 to 99, RETURN meaning one instruction, or a . or SPACE meaning continuous disassembly. 00 means 100 instructions.

   AIM 65 will respond by disassembling instructions until the specified number of instructions are disassembled, RESET is pressed, or ESC is typed. The disassembly can be suspended by typing SPACE (type any key to resume the disassembly).

Example:

```
<K>*=0300
/05
```

```
 0300 EA NOP
 0301 A2 LDX #FE
 0303 E8 INX
 0304 D0 BNE 0303
 0306 4C JMP 0310
<K>*=0310
/04
 0310 A0 LDY #02
 0312 88 DEY
 0313 D0 BNE 0312
 0315 4C JMP 0301
```

3.6 EXECUTION/TRACE COMMANDS

Four commands allow execution and detailed examination of a user written program. The G command executes the user program in the mode determined by the RUN/STEP switch. In the RUN mode, the program executes in real time with complete control of the CPU turned over to the user program.

In the STEP mode, program execution is stopped after each instruction for instruction trace, register trace, and breakpoint examination. The Z command controls the instruction trace while the V command controls the register trace. The breakpoint control is described in Section 3.7. After execution is terminated and control returned to the Monitor, a history trace of the program counter may be obtained using the H command.

3.6.1 G Command - Start Execution at Program Counter Address

The G command starts execution of a user program at the current value of the program counter.

Use the G command as follows:

1. Establish the desired position of the RUN/STEP switch.

2. If the STEP position is selected, perform the following:

   A. Initialize the value of the program counter using the * command.

   B. Set the desired state of the instruction trace mode using the Z command.

   C. Set desired state of the register trace mode using the V command.

   D. Establish any desired breakpoint addresses using the B command.

   E. Enable or disable breakpoint addresses using the 4 command.

   F. Display the register headings and contents using the R command.

3. Type G. AIM 65 will respond with:

   G/

4. In STEP Mode, enter the number of instructions to execute by entering a decimal count from 01 to 99, or a RETURN (meaning one instruction), or a . or SPACE (meaning continuous instruction execution).

5. The AIM 65 will execute instructions as follows until a terminating condition occurs:

   A. In the STEP Mode, the next instruction to be executed will be disassembled and printed if the instruction trace mode (Z command) is on. The contents of the six registers will be printed prior to execution of the next instruction if the register trace mode (Y command) is on. Execution will terminate and control will be returned to the Monitor if the entered count of instructions is reached, a BRK instruction is executed, or a breakpoint address is reached (if breakpoints are enabled).

   B. In the RUN Mode, execution will continue until a BRK instruction is executed, at which time control will be returned to the Monitor. Execution may also be terminated by moving the RUN/STEP switch to the STEP position. If the G command was initiated using the RETURN key, only one instruction will be executed in the STEP Mode before return to the Monitor.

NOTE

If the CPU attempts to execute an unimplemented op code or a jump to an improper address, it may hang up. If this occurs, the RESET switch must be pressed to interrupt program execution and allow the AIM 65 Monitor to regain control.

If execution termination is due to one of the G command terminating conditions, execution may be resumed at the current program counter address by repeating portions of Step 2 without reinitializng the program counter (*). Use the R command to check the value of the program counter before resuming execution.

Example 1: STEP Mode, instruction trace on, register trace on.

NOTE

See Sections 3.6.2 and 3.6.3 for descriptions of the trace commands.

```
<Z>ON
<V>ON
<*>=0300
<R>
 **** PS AA XX YY SS
 0300 A0 00 FF 01 FF
<G>/
 0301 A0 00 FF 01 FF
 0301 A2 LDX #FE
 0303 A0 00 FE 01 FF
 0303 E8 INX
 0304 A0 00 FF 01 FF
 0304 D0 BNE 0303
 0303 A0 00 FF 01 FF
 0303 E8 INX
 0304 22 00 00 01 FF
 0304 D0 BNE 0303
 0306 22 00 00 01 FF
 0306 4C JMP 0310
 0310 22 00 00 01 FF
 0310 A0 LDY #02
 0312 20 00 00 02 FF
 0312 88 DEY
 0313 20 00 00 01 FF
 0313 D0 BNE 0312
 0312 20 00 00 01 FF
 0312 88 DEY
 0313 22 00 00 00 FF
 0313 D0 BNE 0312
 0315 22 00 00 00 FF
 0315 4C JMP 0301
 0301 22 00 00 00 FF
 0301 A2 LDX #FE
 0303 A0 00 FE 00 FF
 0303 E8 INX
 0304 A0 00 FF 00 FF
 0304 D0 BNE 0303
 0303 A0 00 FF 00 FF
 0303 E8 INX
 0304 22 00 00 00 FF
 0304 22 00 00 00 FF
```


Example 2: STEP Mode, instruction trace on, register trace off.

```
<Z>OFF
<Z>ON
<V>OFF
<*>=0300
<R>
 **** PS AA XX YY SS
 0300 22 00 00 00 FF
<G>/
 0301 A2 LDX #FE
 0303 E8 INX
 0304 D0 BNE 0303
 0303 E8 INX
 0304 D0 BNE 0303
 0306 4C JMP 0310
 0310 A0 LDY #02
 0312 88 DEY
 0313 D0 BNE 0312
 0312 88 DEY
 0313 D0 BNE 0312
 0315 4C JMP 0301
 0301 A2 LDX #FE
 0303 E8 INX
 0304 D0 BNE 0303
 0303 E8 INX
 0304 D0 BNE 0303
 0306 4C JMP 0310
 0310 A0 LDY #02
 0312 88 DEY
 0313 D0 BNE 0312
 0312 88 DEY
 0313 D0 BNE 0312
 0315 4C JMP 0301
 0301 A2 LDX #FE
 0303 E8 INX
 0304 D0 BNE 0303
 0303 E8 INX
 0303 E8 INX
```

Example 3: STEP Mode, instruction trace off, register on.

```
<Z>OFF
<V>ON
<*>=0300
<R>
 **** PS AA XX YY SS
 0300 22 00 00 00 FF
<G>/
 0301 22 00 00 00 FF
 0303 A0 00 FE 00 FF
 0304 A0 00 FF 00 FF
 0303 A0 00 FF 00 FF
 0304 22 00 00 00 FF
 0306 22 00 00 00 FF
 0310 22 00 00 00 FF
 0312 20 00 00 02 FF
 0313 20 00 00 01 FF
 0312 20 00 00 01 FF
 0313 22 00 00 00 FF
 0315 22 00 00 00 FF
 0301 22 00 00 00 FF
 0303 A0 00 FE 00 FF
 0304 A0 00 FF 00 FF
 0303 A0 00 FF 00 FF
 0304 22 00 00 00 FF
 0306 22 00 00 00 FF
 0306 22 00 00 00 FF
```

3.6.2 Z Command - Toggle Instruction Trace Mode On/Off

The Z command controls the instruction trace mode when the RUN/STEP switch is in the STEP position and instructions are being executed in response to the G, F1, F2 or F3 command. The instruction trace shows a disassembly of each instruction before the instruction is executed.

To use the Z command, type Z. AIM 65 will respond with the state of the instruction trace mode:

Z ON or
Z OFF

Example:

```
<Z>ON
<Z>OFF
```

In the above example, the first Z command toggled the instruction trace mode to ON. The second Z command toggled the instruction trace mode to OFF.

3.6.3 V Command - Toggle Register Trace Mode On/Off

The V command controls the register trace mode when the RUN/STEP switch is in the STEP position and instructions are being executed in response to the G, F1, F2, or F3 command. The register trace shows the contents of each register, in the format of the R command (Section 3.3.7), after each instruction is executed.

To use the V command, type V. AIM 65 will respond with the status of the register trace mode:

<V> ON or
<V> OFF

3.6.4 H Command - Trace Program Counter History

The H command displays the addresses of the last four instructions that were executed and the address of the next instruction to be executed. This trace capability exists only after the AIM 65 has been executing instructions in the STEP Mode in response to the G, F1, F2, or F3 commands.

Use the H command as follows:

1. Execute the desired instructions in the STEP Mode using the G, F1, F2, or F3 commands.

2. After the Monitor prompt, type H. AIM 65 will respond with:

```
<H>
XXXX  Earliest of last four instructions executed.
XXXX
XXXX
XXXX  Address of instruction just executed.
XXXX  Address of next instruction to be executed.
```

Example:

```
<H>
 0303
 0304
 0306
 0310
 0312
```

The example above shows a program which is a string of sequential non-jump instructions starting at $0303 with a JMP $0310, RTS, RTI or a branch instruction at $0306.

3.7 MANIPULATE BREAKPOINTS

Four commands are provided for breakpoints to check, totally clear, selectively set or clear, and/or enable or disable breakpoints. These commands are used in conjunction with the G command in the STEP Mode to stop instruction execution at specified breakpoint addresses. These commands can, therefore, be used during program checkout to ensure program sequencing to expected addresses or to allow intermediate data in memory to be checked at specified addresses.

3.7.1 ? Command - Display Breakpoints

The ? command displays the address of each of the four breakpoints. The leftmost, four-digit hexadecimal value is the address of breakpoint 0 while the rightmost value is the address of breakpoint 3. $0000 indicates that the breakpoint is cleared, i.e., no breakpoint address is set.

To use the ? command, type SHIFT and ? simultaneously. AIM 65 will respond with:

```
<?>
AAAA AAAA AAAA AAAA
```

Example:

```
<?>
 0312 0000 0000 0000
```

In the above example, the breakpoint numbers and their corresponding addresses are:

BREAKPOINT NUMBER	BREAKPOINT ADDRESS
0	$0312
1	Not Set
2	Not Set
3	Not Set

3.7.2 # Command - Clear Breakpoints

All breakpoints may be cleared by using the # command. Breakpoints should be cleared when AIM 65 power is turned on. RESET does not alter the breakpoint addresses.

To use the # Command, type SHIFT and # simultaneously.

AIM 65 will respond with:

<#>OFF

This indicates that all the breakpoints have been set to $0000.

Example:

<#>OFF

3.7.3 B Command - Set/Clear Breakpoints

The B command sets or clears the address for any of the four breakpoints (breakpoint 0 through breakpoint 3).

For breakpoints to be checked, AIM 65 must be in the STEP Mode, and the breakpoints must be enabled with the 4 command.

When the AIM 65 is in the STEP Mode, and the breakpoints are enabled, the processor halts each time an instruction fetch is made in the address range $0001 to $9FFF. Entry is made to the Monitor via the NMI interrupt vector (unless the NMI vector address in location $A402 has been altered). A check is made to see if the breakpoints are enabled (see the 4 command). If the breakpoints are enabled, each set breakpoint address is compared with the address of the instruction about to be executed. If the address of one of the set breakpoints matches the address of the instruction about to be executed, execution of the program is halted and control is returned to the Monitor.

Use the B command as follows:

1. Type B. AIM 65 will respond with:

   <B> BRK/

2. After the / prompt, specify the breakpoint to be set/cleared by entering a digit between 0 and 3. AIM 65 will respond by printing the number of the breakpoint entered, and an = prompt. For example, if 0 is entered:

   <B> BRK/0=

3. To set a breakpoint, enter the four-digit hexadecimal address at which the program is to halt. To clear a breakpoint, enter 0.

4. After the address has been entered, type RETURN. Control will return to the Monitor. Re-enter the B command to set or clear additional breakpoints.

Example:

```
<B>BRK/0=0310
<B>BRK/1=0312
<B>BRK/2=0
```

In the above example, breakpoint 0 was set to location $0310, breakpoint 1 was set to location 0312, breakpoint 2 was left unchanged and breakpoint 3 was set to location $0000 (i.e., cleared).

3.7.4 4 Command - Toggle Breakpoint Enable On/Off

The 4 command toggles the breakpoint enable ON or OFF. When the breakpoint enable is ON, and the Step mode is selected, the breakpoints in a program are checked.

The normal mode of operation is to assign the breakpoint addresses with the B command and then enable these breakpoints with the 4 command when the user wants them to be checked.

The 4 command also allows breakpoint addresses to be disabled temporarily without requiring them to be reentered later.

The breakpoint enable is automatically turned OFF when AIM 65 power is turned on. Subsequent RESETS do not affect breakpoint enable.

To use the 4 command, type 4. The system will toggle the breakpoint enable and display the result:

```
<4> ON or
<4> OFF
```

Example:

```
<4>ON
<4>OFF
```

In the above example, the breakpoints were enabled (toggled ON) when the first 4 command was entered. The breakpoints were disabled (toggled OFF) when the second 4 command was entered.

3.8 LOAD/DUMP MEMORY

Two commands allow R6500 object code to be loaded into memory from an input device or dumped from memory to an output device.

3.8.1 L Command - Load Memory

The L command loads object code from any system device into memory.

Use the L command as follows:

1. Type L. AIM 65 will respond with:

```
<L> IN=
```

2. Type the code of the input device from which the object code is to be loaded:

DESIRED INPUT DEVICE	ENTER DEVICE CODE	ADDITIONAL PROCEDURE
Keyboard	<RETURN> or <SPACE>	
Audio Tape - AIM 65 Format	<T>	See Section 9.1.2
Audio Tape - KIM-1 Format	<K>	See Section 9.1.2
TTY Punched Tape	<L>	See Section 9.2.2
User Defined	<U>	See Section 7

3. AIM 65 will load the object code from the specified device into memory. When all the code has been loaded, the AIM 65 will print the Monitor prompt (<).

   If any of the records being read contains a checksum error, or if any part of the memory fails to write, an error message will be printed (see Appendix L), indicating the first address of the record which caused the error.

3.8.2 D Command - Dump Memory

The D command is used to dump the contents of memory to an output device. Memory contents dumped are in R6500 object code format (see Appendix F.2) from the address specified after FROM=, through the address specified after TO=. Multiple dumps from different portions of memory may be performed by entering new beginning and ending addresses after responding Y to the MORE? prompt. An N response is required to terminate the dump properly.

Use the D command as follows:

1. Type D. AIM 65 will respond by asking for the dump beginning address:

   ```
   <D>
   FROM=
   ```

2. Enter the beginning address to be dumped, in hexadecimal. An input error may be corrected using DEL, or by continuing to enter up to 11 numbers; AIM 65 will accept only the last four numbers entered. End the input with RETURN or SPACE.

   If 0300 was entered, AIM 65 will respond by asking for the dump ending address:

   ```
   FROM=0300   TO=
   ```

3. Enter the ending address to be dumped, in hexadecimal. An input error may be corrected in the same manner as

in the beginning address. End the input with a RETURN or SPACE. If 0340 was entered, AIM 65 will respond with:

```
FROM=0300     TO=0340
OUT=
```

4. Type the code of the output device to which the dump is to be directed:

DESIRED OUTPUT DEVICE	ENTER DEVICE CODE	ADDITIONAL PROCEDURE
AIM 65 Display/ Printer	<RETURN> or <SPACE>	
AIM 65 Printer	<P>	
Audio Tape - AIM 65 Format	<T>	See Section 9.1.2
Audio Tape - KIM-1 Format	<K>	See Section 9.1.2
TTY Punched Tape	<L>	See Section 9.2.2
User Defined	<U>	See Section 7
Dummy Output (None)	<X>	

5. The memory contents will be dumped to the specified output device in R6500 object code format. When memory has been dumped through the specified ending address, AIM 65 will display:

```
MORE?
```

6. If another section of memory is to be dumped, enter a Y (yes) response. AIM 65 will ask for the new beginning and ending addresses. If no more memory is to be dumped, enter an N (no) response.

7. After an N response, AIM 65 will output the terminating record with a zero byte count (see Appendix F.2).

NOTE

If the dump is not terminated with an N response, the last file record containing the file record count will not be recorded causing a subsequent improper load or tape verify.

Example 1:

```
<D>
FROM=0200 TO=0366
OUT=T F=DUMP1 T=1
MORE?N
```

This example dumps memory locations $200 to $366 to a tape file called DUMP1 located on the tape recorder number 1. No other segments were dumped.

Example 2:

```
<D>
FROM=0200 TO=0366
OUT=T F=DUMP2 T=2
MORE?Y
FROM=0300 TO=0380
MORE?N
```

This example dumps memory locations $200 to $366 to tape file called DUMP2 located on tape recorder number 2. More memory locations were to be dumped, so the question MORE? was answered with Y. The second dump was from location $0300 to location $0380. No more segments were dumped to this file.

Example 3:

```
<D>
FROM=0300 TO=0316
OUT=

;170300E9A1FEE300FD4
C100364F1F30CEF3D72A
00230D0FD4C010DE1
MORE?Y
FROM=0380 TO=03B0

;180380FCF0F25064D1F
1D0E9B0F4E0FF0DAC170F
F76F2E2D222F680129F
;180398E7F4E369EB547
250F650767270F0F072F
7D1F460D5A574901007
;0103B0F301A7
MORE?N;0000050005
```


This example shows an actual memory dump from location $0300 to $0316 and from $0380 to $03B0. The question OUT= was answered with the response RETURN, causing the dump to be printed on the AIM 65 printer and displayed on the AIM 65 display.

NOTE

> If memory is dumped to audio cassette using KIM-1 format (OUT=K), the TO address entered must be one byte greater than the last address to be dumped.

3.9 PERIPHERAL CONTROL

The peripheral control commands allow the printer and audio tape recorders to be controlled and miscellaneous functions to be performed.

3.9.1 Control Print Command - Toggle Printer On/Off

The CTRL PRINT command turns the printer control on if it is off, and off if it is on.

If the printer control is ON, entering CTRL PRINT will toggle it to OFF. If the printer control is OFF, entering CTRL PRINT will toggle it to ON.

The command is entered by depressing the PRINT and CTRL keys together. The status of the printer control will be displayed but not printed.

When the printer control is ON, the Monitor will print the same commands and data that are displayed. If the printer control is OFF, the only time information will be printed is when the PRINT key alone is depressed.

NOTE

The printer control is turned ON by the first RESET after AIM 65 power is turned on. Subsequent RESETs will not change the active state of the printer control.

3.9.2 PRINT Command - Print Display Contents

The PRINT command causes the displayed information to be printed. This commanded print will occur regardless of the printer control state as long as the Monitor, Editor, Assembler, or BASIC Interpreter is active.

3.9.3 LF Command - Advance Paper

The LF (Line Feed) Command is used to advance the printer one line. The LF Command will operate whenever the Monitor, Editor, Assembler, or BASIC Interpreter is active.

3.9.4 1 (2) Command - Toggle Tape 1 (2) Control On/Off

The text to follow describes the 1 command. It applies equally to the 2 command.

The 1 command is used to toggle the Tape 1 control. If the Tape 1 control is ON, entering the 1 command will turn it

OFF. If the Tape 1 control is OFF, entering the 1 command will turn it ON.

The Tape 1 control is usually connected to the tape recorder number 1 Remote line (see Section 9). If so connected, the tape recorder will not record, play, advance, or rewind unless the Tape 1 control is ON. The Monitor and Editor commands requiring tape recorder number 1 operation will command the Tape 1 control ON when required and turn the control signal OFF upon completion of the command.

These commands are L (Load) and D (Dump) in the Monitor and R (Read) and L (List) in the Editor. To manually operate tape recorder number 1, the Tape 1 control must be turned ON using the 1 command.

NOTE

The Tape controls are turned on when AIM 65 power is turned on. RESET does not change the active state of the Tape controls.

3.9.5 3 Command - Verify Tape

The 3 command is used to verify that the block checksum for either source or object code was properly recorded on audio tape using the dump command. This verification should be performed before the contents of memory are altered, in case the data was not properly recorded.

To use the 3 command, type 3. AIM 65 will respond with:

```
<3>IN=
```

Type T. AIM 65 will respond with:

```
<3>IN=T  F=
```

Example:

```
<3>IN=T F=OBJ  T=1
```

Enter the file name and tape recorder number as shown in Section 9.1.6. The operation will continue as described in that section.

3.10 USER FUNCTION INTERFACE

Three commands allow execution of three separate user written functions (programs) from the AIM 65 Monitor using keys F1, F2, and F3. In order to use a function key, the linkage to the user function must be provided. The linkage is in the form of a JMP instruction to the function starting address. The JMP instruction is to be located at the function linkage address (see the specific function number for the actual address). At the completion of the user function, control to the AIM 65 Monitor may be regained with an RTS instruction at the end of the user function instruction.

The JMP instruction can be established at the function linkage addresses using the I Instruction Mnemonic Entry, the N Assembler, or by entering the JMP instruction in hexadecimal form using the / Alter Memory commands.

3.10.1 F1, F2, F3 Command - User Functions 1, 2, and 3

The F1, F2, and F3 commands are used to enter user functions.

To use the F1, F2 or F3 command, proceed as follows:

1. Establish a JMP instruction in address $010C to the Function 1 starting address, in $010F for Function 2, or in $0112 for Function 3.

2. Enter the function by entering command F1, F2, or F3. AIM 65 will respond with the following display and will start function 1, 2, or 3.

KEY	PROMPT
F1	<[>
F2	<]>
F3	<^>

3. Return to the Monitor by executing an RTS instruction without a preceding JSR in the user function program or by depressing the RESET button.

F1 Example:

```
<I>
 0200     *=010C
 010C 4C JMP 0220
 010F     *=0220
 0220 60 RTS
 0221

<I>
```

F2 Example:

```
<I>
 0200     *=010F
 010F 4C JMP 0240
 0112     *=0240
 0240 60 RTS
 0241

<I>
```

F3 Example:

```
<I>
 010F     *=0112
 0112 4C JMP 0260
 0115     *=0260
 0260 60 RTS
 0261

<I>
```