Table 7-10. AIM 65 I/O Memory Map (A000-AFFF)

A000	User R6522
A00F	(Z1)
A010	Not Available
A3FF	
A400	Monitor RAM
A47F	
A480	AIM 65 R6532
	(Z33)
A497	Keyboard I/O
A498	Not Available
A7FF	
A800	AIM 65 R6522
	(Z32)
	Printer, TTY &
A80F	Tape I/O
A810	Not Available
ABFF	
AC00	AIM 65 R6520
	(U1)
AC43	Display I/O
AC44	Not Available
AFFF	

NOTE

(1) Unassigned addresses are not available to the user due to address conflict with AIM 65 I/O assignments.

Table 7-11. AIM 65 Detail Memory Map

ADDRESS	AREA	LABEL	BYTES	FUNCTION
0000	User RAM			Note: Addresses 0004 through 00DE are used by the Assembler.
00AC	↓			
00AD	Assembler ↓	TABUF2	50	Assembler Tape Output Buffer
00DF	↑	NOWLN	2	Current Line
00E1		BOTLN	2	Last Active, so far
00E3	Editor	TEXT	2	Text Buffer Starting Address
00E5		END	2	Text Buffer Ending Address Limit
00E7		SAVE	2	Used by Replace
00E9		OLDLEN	1	Orig Length
00EA		LENGTH	1	New Length
00EB	↓	STRING	20	Find String
0100	Breakpoints	BKS	8	BRK Locations
0106	Dump	S2	2	Vertical Count
0108	User I/O	UIN	2	User Input Handler Address
010A	Handler	UOUT	2	User Output Handler Address
010C	User	KEYF1	3	JMP Instruction to F1 Function
010F	Functions	KEYF2	3	JMP Instruction to F2 Function
0112		KEYF3	3	JMP Instruction to F3 Function
0115	Tape I/O	BLK	1	Input Block No.
0116	↓	TABUFF	80	Tape Buffer (I/O)
0168		BLKO	2	Output Block No.

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
0116	Disassembler	FORMA	1	
0117		LMNEM	1	
0118		RMNEM	1	
0119		SORN	1	
011A		NUM	2	
011C		SYM	6	
0122		SIZE	2	
0124		STSAVE	2	
0126	Mnemonic	TMASK1	1	
0127	Entry	TMASK2	1	
0126	Mnemonic	MOVAD	2	
0128	Entry	BEGAD	2	
012A		ENDAD	2	
012C		TABEND	2	
012E		TYPE	1	
012F		HASHDA	1	
0130		CH	3	
0133		ADFLD	20	
0169	Reserved for			
016F	Monitor			
0170	Stack		144	Available to AIM 65 and User (Addresses 0170 through 0183 are used by the Assembler)
01FF	Pointer			

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
A000	User R6522 VIA (Z1)	UDRB	1	Data Register B
		BIT	SIGNAL	CONN/PIN — AIM 65 SIGNAL NAME
		0	PB0	J1-9 — User Defined
		1	PB1	J1-10
		2	PB2	J1-11
		3	PB3	J1-12
		4	PB4	J1-13
		5	PB5	J1-16
		6	PB6	J1-17
		7	PB7	J1-15
A001		UDRAH	1	Data Register A (Controls Handshake)
		BIT	SIGNAL	CONN/PIN — AIM 65 SIGNAL NAME
		0	PA0	J1-14 — User Defined
		1	PA1	J1-4
		2	PA2	J1-3
		3	PA3	J1-2
		4	PA4	J1-5
		5	PA5	J1-6
		6	PA6	J1-7
		7	PA7	J1-8
A002		UDDRB	1	Data Direction Register B
A003		UDDRA	1	Data Direction Register A
A004		UT1L	1	Write T1L-L; Read T1CL, Clear T1IF
A005		UT1CH	1	Write T1L-H & T1C-H, T1L-L-T1C-L, Clear T1IF; Read T1C-H
A006		UT1LL	1	Write T1L-L; Read T1L-L

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
A007	User R6522 VIA (Z1)	UT1LH	11	Write T1L-H, Clear T1IF; Read T1L-H
A008		UT2L	1	Write T2L-L; Read T2C-L, Clear T2IF;
A009		UT2H	1	Write T2C-H, T2L-L-T2C-L Clear T2IF; Read T2C-H
A00A		USR	1	Shift Register (SR)
A00B		UACR	1	Auxiliary Control Register (ACR)
A00C		UPCR	1	Peripheral Control Register (PCR)
		BIT	SIGNAL	CONN/PIN AIM 65 SIGNAL NAME
		0	CA1	J1-19 User Defined
		1-3	CA2	J1-18
		4	CB1	J1-20
		5-6	CB2	J1-21
A00D		UIFR	1	Interrupt Flag Register (IFR)
A00E		UIER	1	Interrupt Enable Register (IER)
A00F		UDRA	1	Data Register A (No Effect on Handshake)
A010 A3FF				Not Available
A400	Interrupt Indirect Jump Vectors (User Alterable)	IRQV4	2	IRQ after Monitor vector
A402		NMIV2	2	NMI Vector
A404		IRQV2	2	IRQ Vector

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
A406	I/O Devices (User Alterable)	DILINK	2	Display Linkage (Echo to Display)
A408		TSPEED	1	Tape Speed
A409		GAP	1	Timing for GAPs between blocks
A40A	Monitor Parameters and Flags	NPUL	1	No. of Half Pulses
A40B		TIMG	3	Tape Timing
A40E		REGF	1	Register Trace Flag
A40F		DISFLG	1	Instruction Trace Flag
A410		BKFLG	1	Enable or Disable Breakpoints
A411		PRIFLG	1	Enable or Disable Printer
A412		INFLG	1	Input Device
A413		OUTFLG	1	Output Device
A414		HISTP	1	PC History Trace Pointer
A415		CURPO2	1	Display Pointer
A416		CURPOS	1	Printer Pointer
A417		CNTH30	1	Baud Rate
A418		CNTL30	1	TTY Delay
A419		COUNT	1	0-99
A41A		S1	2	Start Address
A41C		ADDR	2	End Address
A41E		CKSUM	2	Checksum
A41C	Mnemonic Entry	CURAD	2	Current Address
A420	Monitor Saved Registers	SAVPS	1	Processor Status
A421		SAVA	1	Accumulator

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
A422	Monitor	SAVX	1	X Register
A423	Saved	SAVY	1	Y Register
A424	Registers	SAVS	1	Stack Pointer
A425	↓	SAVPC	2	Program Counter
A427	Page Zero	STIY	3	STA NM, Y
A42A	Work Area	CPIY	4	CMP NM, Y or LDA NM, Y and RTS
A42A	Page Zero Work	LDIY	3	LDA NM, Y
A42A	Keyboard	KMASK	1	To Mask Off CTRL or SHIFT
A42B		STBKEY	1	Strobe Key (1-8)
A42E	Cassette	NAME	5	File Name
A433	Tape	GANG	1	Output PB7
A434		TAPIN	1	Input Flag (Tape 1 or 2)
A435		TAPOUT	1	Output Flag (Tape 1 or 2)
A436		TAPTR	1	Tape Buffer Pointer
A437	↓	TAPTR2	1	Tape Output Buffer Pointer
A42E	PC Trace	HIST	10	Four Last Addresses, Plus Next Address
A42E	Mnemonic	FLAG	1	
A42F	Entry	BYTESM	2	
A431		TEMPX	1	

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
A432	Mnemonic	TEMPY	1	
A433	Entry	TEMPA	1	
A435		OPCODE	3	
A438	↓	CODFLG	1	
A438	Display Buffer	DIBUFF	40	Display Buffer
A460	Printer	IBUFM	20	Print Buffer
A474		IDIR	1	Direction 0=>+ FF=>-
A475		ICOL	1	Column: 0 = Leftmost 4 = Rightmost
A476		IOFFST	1	Offset: 0 = Left Digit, 1 = Right Dgt.
A477		IDOT	1	No. of Last Dot Encountered
A478		IOUTL	1	Lower 8 Outputs (8 Cols. on Right)
A479		IOUTU	1	Upper 2 Digits
A47A		IBITL	1	1 Bit Mask for Current Output
A47B		IBITU	1	
A47C		IMASK	1	Mask for Current Row
A47D	↓	JUMP	2	Indirect Address of Table for Current Row
A47F	Keyboard	ROLLFL	1	Save Last Strobe for Rollover

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
A480	Monitor R6532 RIOT (Z33)	DRA2	1	Data Register A
		BIT	SIGNAL	CONN/PIN I/O AIM 65 SIGNAL NAME
		0	PA0	J4-2 I Keyboard KI1
		1	PA1	J4-6 I Keyboard KI2
		2	PA2	J4-8 I Keyboard KI3
		3	PA3	J4-7 I Keyboard KI4
		4	PA4	J4-11 I Keyboard KI5
		5	PA5	J4-12 I Keyboard KI6
		6	PA6	J4-5 I Keyboard KI7
		7	PA7	J4-3 I Keyboard KI8
A481		DDRA2	1	Data Direction Register A
A482		DRB2	1	Data Register B
		BIT	SIGNAL	CONN/PIN I/O AIM 65 SIGNAL NAME
		0	PB0	J4-1 O Keyboard KO1
		1	PB1	J4-10 O Keyboard KO2
		2	PB2	J4-16 O Keyboard KO3
		3	PB3	J4-15 O Keyboard KO4
		4	PB4	J4-14 O Keyboard KO5
		5	PB5	J4-13 O Keyboard KO6
		6	PB6	J4-4 O Keyboard KO7
		7	PB7	J4-9 O Keyboard KO8
A483		DDRB2	1	Data Direction Register B
A484		DNPA7	1	Write Disable PA7 Interrupt, Negative Edge Detect

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
A485	Monitor R6532 RIOT (Z33)	DPPA7	1	Write Disable PA7 Interrupt Positive Edge Detect
A486		ENPA7	1	Write Enable PA7 Interrupt Negative Edge Detect
A486		RINT	1	Read Bit 7 = Timer Flag, Bit 6 = PA7 Flag, Clear Int.
A487		EPPA7	1	Write Enable PA7 Interrupt, Positive Edge Detect
A494		DIV1	1	Div. by 0001 (Disable), Add 8 to Enable
A495		DIV8	1	Div. by 0008 (Disable), Add 8 to Enable
A496		DIV64	1	Div. by 0064 (Disable), Add 8 to Enable
A497		DI1024	1	Div. by 1024 (Disable), Add 8 to Enable
A498 A7FF				Not Available
A800	Monitor R6522 VIA (A800-ABFF) (Z32)	DRB	1	Data Register B
		BIT	SIGNAL	CONN/PIN I/O AIM 65 SIGNAL NAME
		0	PB0	J2-2 O Printer TE9
		1	PB1	J2-1 O Printer TE10
		2	PB2	J1-W I TTY PTR
		3	PB3	- I S1 - KB/TTY (1=KB)
		4	PB4	J1-W O TAPE 1A
				J1-F O TAPE 1B
		5	PB5	J1-V O TAPE 2A
				J1-J O TAPE 2B
		6	PB6	J1-Y I TTY SERIAL IN
		7	PB7	J1-L I AUDIO IN
		7	PB7	J1-M O AUDIO OUT LO
		7	PB7	J1-P O AUDIO OUT HI

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
A801	Monitor R6522 VIA (A800-ABFF) (Z32)	DRAH	1	Data Register A (Controls handshake)
		BIT	SIGNAL	CONN/PIN I/O AIM 65 SIGNAL NAME
		0	PA0	J2-11 O Printer TE1
		1	PA1	J2-10 O Printer TE2
		2	PA2	J2-9 O Printer TE3
		3	PA3	J2-8 O Printer TE4
		4	PA4	J2-7 O Printer TE5
		5	PA5	J2-5 O Printer TE6
		6	PA6	J2-4 O Printer TE7
		7	PA7	J2-3 O Printer TE8
A802		DDRB	1	Data Direction Register B
A803		DDRA	1	Data Direction Register A
A804		T1L	1	Write T1L-L; Read T1C-L, Clear T1IF
A805		T1CH	1	Write T1L-H & T1C-H, T1L-L T1C-:, Clear T1IF; Read T1C-H
A806		T1L1	1	Write T1L-L; Read T1L-L
A807		T1LH	1	Write T1L-H, Clear T1IF; Read T1L-H
A808		T2L	1	Write T2L-L; Read T2C-L, Clear T2IF;

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
A809	Monitor R6522 VIA (Z32)	T2H	1	Write T2C-H, T2L-L T2C-L, Clear T2IF; Read T2C-H
A80A		SR	1	Shift Register (SR)
A80B		ACR	1	Auxiliary Control Register (ACR)
A80C		PCR	1	Peripheral Control Register (PCR)
		BIT	SIGNAL	CONN/PIN I/O AIM 65 SIGNAL NAME
		0	CA1	J2-13 I Printer P1
		1-3	CA2	- O Data IN/Data OUT
		4	CB1	J2-14 I Printer Start
		5-7	CB2	J2-17 O Printer M+ (motor)
A80D		IFR	1	Interrupt Flag Register (IFR)
A80E		IER	1	Interrupt Enable Register (IER)
A80F		DRA	1	Data Register A (No effect on handshake)
A810 ABFF				Not available
AC00	Monitor R6520 PIA Display (Z1)	RA	1	Data Register A
		BIT	SIGNAL	CONN/PIN AIM 65 SIGNAL NAME
		0	PA0	- Display A0
		1	PA1	- Display A1
		2	PA2	- Display DS1 CE
		3	PA3	- Display DS2 CE
		4	PA4	- Display DS3 CE
		5	PA5	- Display DS4 CE
		6	PA6	- Display DS5 CE
		7	PA7	- Display W

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
AC01	Monitor	CRA	1	Control Register A
AC02	R6520	RB	1	Register B
	PIA			
	Display	BIT	SIGNAL	CONN/PIN AIM 65 SIGNAL NAME
	(Z1)	0	PB0	- Display D0
		1	PB1	- Display D1
		2	PB2	- Display D2
		3	PB3	- Display D3
		4	PB4	- Display D4
		5	PB5	- Display D5
		6	PB6	- Display D6
		7	PB7	- Display CU
AC03		CRB	1	Control Register B
AC04		DISP1	1	Display 1 Digit 0 - Char 4
AC05			1	Display 1 Digit 1 - Char 3
AC06			1	Display 1 Digit 2 - Char 2
AC07			1	Display 1 Digit 3 - Char 1
AC08		DISP2	1	Display 2 Digit 0 - Char 8
AC09			1	Display 2 Digit 1 - Char 7
AC0A			1	Display 2 Digit 2 - Char 6
AC0B			1	Display 2 Digit 3 - Char 5
AC10		DISP3	1	Display 3 Digit 0 - Char 12
AC11			1	Display 3 Digit 1 - Char 11
AC12			1	Display 3 Digit 2 - Char 10
AC13			1	Display 3 Digit 3 - Char 9

Table 7-11. AIM 65 Detail Memory Map (Cont.)

ADDRESS	AREA	LABEL	BYTES	FUNCTION
AC20	Monitor	DISP4	1	Display 4 Digit 0 - Char 16
AC21	R6520		1	Display 4 Digit 1 - Char 15
AC22	PIA		1	Display 4 Digit 2 - Char 14
AC23	Display		1	Display 4 Digit 3 - Char 13
AC40	(Z1)	DISP5	1	Display 5 Digit 0 - Char 20
AC41			1	Display 5 Digit 1 - Char 19
AC42			1	Display 5 Digit 3 - Char 18
AC43			1	Display 5 Digit 4 - Char 17
AC44 AFFF				Not Available
B000 CFFF	AIM 65 BASIC		8192	
D000 DFFF	AIM 65 Assembler		4096	
E000 FFFF	AIM 65 Monitor		8192	
FFFA	Monitor Interrupt	NMIV1	2	NMI Vector to E075
FFFC	Vectors	RSET	2	RESET Vector to E0BF
FFFE		IRQV1	2	IRQ Vector to E078

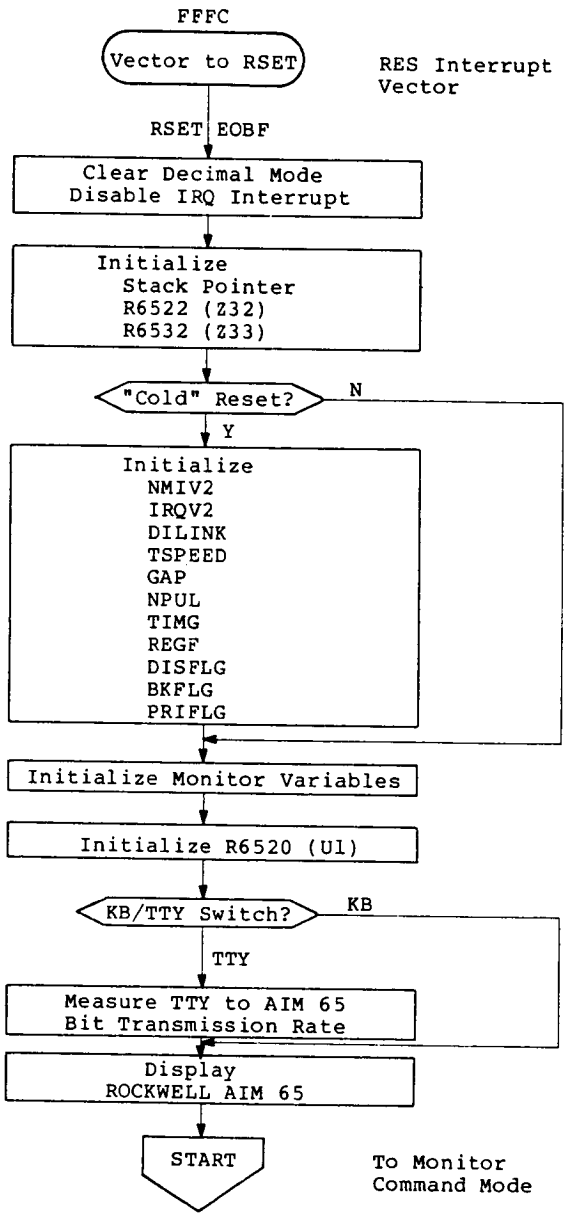

Figure 7-14. AIM 65 Software Flowchart

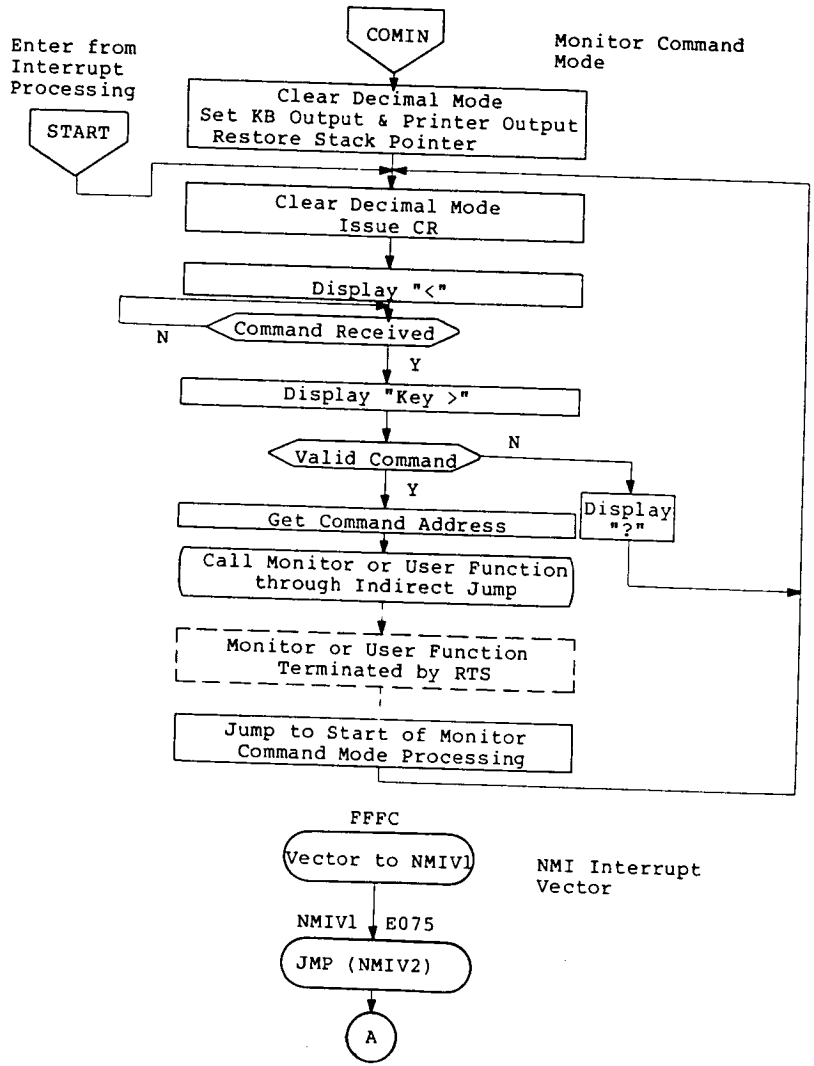

Figure 7-14. AIM 65 Software Flowchart (Cont.)

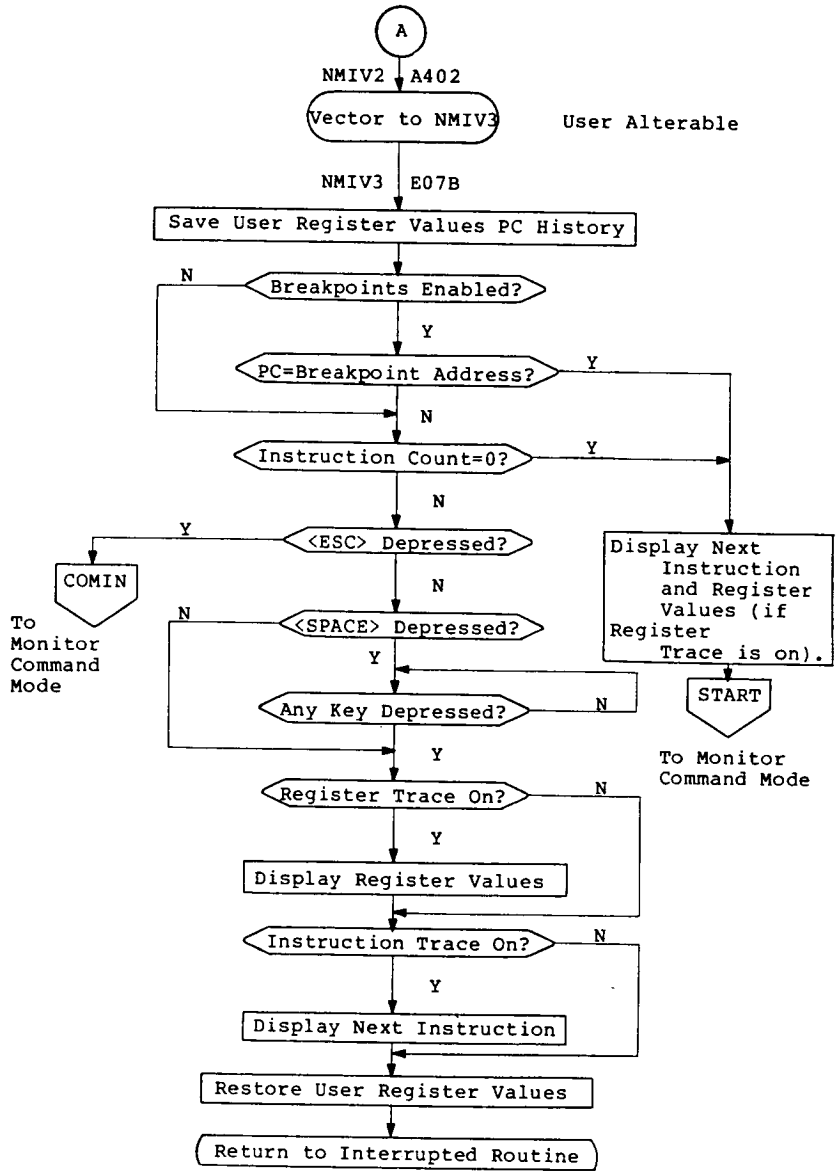

Figure 7-14. AIM 65 Software Flowchart (Cont.)

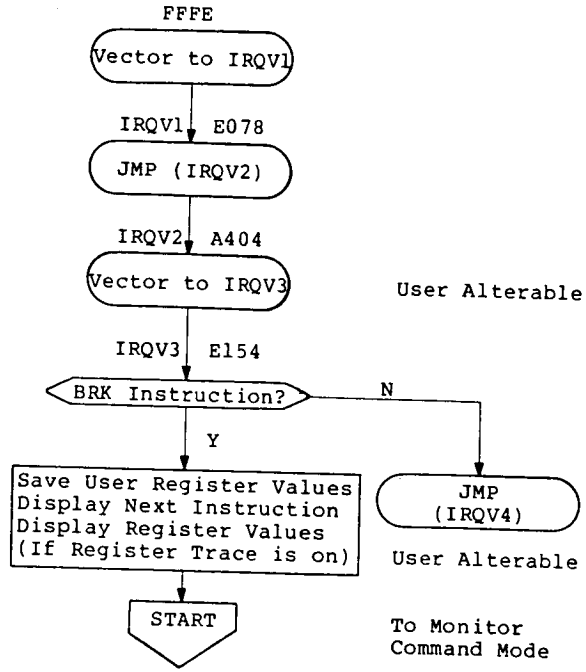

Figure 7-14. AIM 65 Software Flowchart (Cont.)

7.6 USER PARAMETERS

Table 7-12 identifies the user-alterable parameters. Some of the parameters are initialized by a "cold" reset (see Table 7-8).

Table 7-12. User-Alterable Parameters

LOCATION	NAME	BYTES	DESCRIPTION	INITIALIZED
0108	UIN	2	Vector to User Input Handler	No
010A	UOUT	2	Vector to User Output Handler	No
010C	KEYF1	3	JMP Instruction to User Function 1	No
010F	KEYF2	3	JMP Instruction to User Function 2	No
0112	KEYF3	3	JMP Instruction to User Function 3	No
A400	IRQV4	2	Vector to IRQ after Monitor Interrupt Routine	Yes
A402	NMIV2	2	Vector to NMI Interrupt Routine	Yes
A404	IRQV2	2	Vector to IRQ Interrupt Routine	Yes
A406	DILINK	2	Vector to Display Routine	Yes

Table 7-12. User-Alterable Parameters (Cont.)

LOCATION	NAME	BYTES	DESCRIPTION	INITIALIZED
A408	TSPEED	1	Audio Tape Speed Default = $C7 (AIM 65 format) Options: $5A (KIM 1 format x 1) $5B (KIM 1 format x 3)	Yes
A409	GAP	1	Audio Tape Gap Default = $08 = 32 SYN characters Option: $80 = 512 SYN characters for Assembler input and Editor update.	Yes

7.7 USER AVAILABLE SUBROUTINES

Many AIM 65 Monitor subroutines are available for development of application software. An application program designed to operate in AIM 65 while the AIM 65 Monitor ROMs are installed can use the subroutines by calling them with the Jump to Subroutine (JSR) instruction.

Application software designed to operate in hardware separate and independent from AIM 65, or in AIM 65 when the AIM 65 Monitor ROMs are removed, may use the design of these subroutines. In this case, the source instructions for the subroutines may be determined from the AIM 65 program listing.

The user available subroutines are identified and explained in Table 7-13.

AIM 65 subroutines other than those listed in Table 7-13 may be found in the AIM 65 Monitor Listing. These subroutines may be used for application software on an assumed risk basis by the user. Indiscriminate use of some of these subroutines may interfere with the proper operation of the AIM 65 Monitor.

The ENTRY ADDRESS listed in Table 7-13 may be the first address of the subroutine or may be a Jump (JMP) instruction to the first address of the subroutine. The SUBROUTINE NAME is the name used by the AIM 65 Monitor source program. Any data in registers that may be changed by the subroutine are identified in the REGISTERS ALTERED column. Any data in an altered register that is to be retained must be saved before calling the subroutine and restored after return from the subroutine by the application program.

Table 7-13. AIM 65 Monitor Subroutines

SUBRTN NAME	ENTRY ADDR.	REG ALT.	I O	FUNCTION
BLANK	E83E	A	O	Outputs one SP ($20) to Display/Printer.
BLANK2	E83B	A	O	Outputs two SP ($20) to Display/Printer.
CLR	EB44	A	M	Clears display and printer pointers.
CRCK	EA24	A	O	Outputs contents of print buffer if the print pointer is not clear.
CRLF	E9F0	A	O	Outputs one CR ($0D), one LF ($0A) and, if TTY one NUL ($00), to the active output device.
CRLOW	EA13		O	Outputs one CR ($0D) and one LF ($0A) to the display/printer.
CUREAD	FE83	A	I	Reads one character from the keyboard. Returns with the ASCII code in A. Displays cursor upon return.
DUMPTA	E56F		O	Opens a audio tape output file by setting the tape buffer pointer and moving the file name from NAME ($A42E) to the tape buffer. The rest of the tape buffer can be loaded with output data using OUTALL. The recorder 1 or 2 control

Table 7-13. AIM 65 Monitor Subroutines (Cont.)

SUBRTN NAME	ENTRY ADDR.	REG ALT.	I O	FUNCTION
DUMPTA (Cont.)				will turn on automatically depending on the state of TAPOUT ($A435).
DU11	E50A			Closes a tape block if the audio tape output is active. Turns both recorder controls on. Stores CR ($0D) in INFLG ($A412) and OUTFLG (A714) to indicate input from the keyboard and output to the display/printer.
DISASM	F46C		O	Outputs the disassembly of the current instruction pointed to by SAVPC ($A425) to the active output device.
EQUAL	E7D8	A	O	Outputs one "=" ($3D)to the display/printer.
FROM	E7A3	AXY	O	Outputs "FROM=" to the display/printer and puts the address in ADDR ($A41C) and ADDR+1 ($A41D).
GETTAP	EE29	A Y	I	Reads one character from audio tape into A. The tape must be in sync to read the character properly.

Table 7-13. AIM 65 Monitor Subroutines (Cont.)

SUBRTN NAME	ENTRY ADDR.	REG ALT.	I O	FUNCTION
HEX	EA7D	A	M	Converts a Hex number (0-9, A-F) in A from ASCII format to hex format and puts the result in the LSD of A and puts zero in the MSD of A. Sets Carry bit if the input character is not a hex number.
INALL	E993	A	I	Reads one ASCII character from the active input device. Returns with the input character in A. The input device code must be in INFLG ($A412) before calling.
INLOW	E8F8	A	M	Puts CR ($0D) in INFLG ($A412) to indicate keyboard input.
LOADTA	E32F		I	Searches for audio tape file with file name as specified in NAME ($A42E). Turns on recorder control 1 or 2 depending on the state of TAPIN ($A434). Loads only the first block of the located file. Each byte of the loaded block can then be read using INALL. Subsequent blocks will be loaded automatically whenever the tape input buffer is empty.

Table 7-13. AIM 65 Monitor Subroutines (Cont.)

SUBRTN NAME	ENTRY ADDR.	REG ALT.	I O	FUNCTION
LOADTA (Cont.)				Subroutine LOAD1 can be used after LOADTA to load the remaining blocks.
LL	E8FE	A	M	Puts CR ($0D) in INFLG ($A412) to indicate input from the keyboard and puts CR ($0D) in OUTFLG ($A413) to indicate output to the display/printer
NOUT	EA51	A	O	Converts bits 0-3 of A to an ASCII character and outputs it to the active output device.
NUMA	EA46	A	O	Converts two hex numbers in A from hex format to ASCII format (MSD first) and outputs them to the output device.
OUTLOW	E901	A	M	Puts CR ($0D) in OUTFLG ($A413) to indicate output to the display/printer rather than to the active output device.
OUTALL	E9BC		O	Outputs one ASCII character in A to the active output device.

Table 7-13. AIM 65 Monitor Subroutines (Cont.)

SUBRTN NAME	ENTRY ADDR.	REG ALT.	I O	FUNCTION
OUTDIS	EF05			Outputs an ASCII character in A to the display. If more than 20 characters are displayed, the display is scrolled to the left. Stops scrolling when 60 characters have been displayed.
OUTDP	EEFC			Outputs an ASCII character to the display and to the print buffer (calls OUTPRI). Links to the display subroutine (OUTDIS) indirectly through user alterable vector DILINK ($A406).
OUTPRI	F000			Outputs an ASCII character in A to the print buffer. Prints a line when 20 characters are in the buffer or a CR ($0D) in output.
OUTPUT	E97A		O	Outputs one ASCII character from A to the display/printer or TTY. If Bit 7 of PRIFLG ($A411) = 1, the character is also output to the printer (if the printer is enabled). If the TTY is active and bit 0 of PRIFLG = 0, the output is to the TTY.

Table 7-13. AIM 65 Monitor Subroutines (Cont.)

SUBRTN NAME	ENTRY ADDR.	REG ALT.	I O	FUNCTION
OUTTAP	F24A	Y	O	Outputs one character from A to the tape. The SYN ($16) characters must be previously output to tape using TAOSET.
PACK	EA84	A	M	Converts a hex number in A from ASCII format to hexadecimal format and puts the result in the LSD of A and the result of the last call to PACK in the MSD of A.
PHXY	EB9E		M	Push X and Y register contents onto the stack.
PLXY	EBAC	XY	M	Pull X and Y register contents from the stack.
PSL1	E8E7	A	O	Outputs "/" ($2F) to the display/printer.
QM	E7D4	A	O	Outputs "?" ($3F) to the display/printer.
RBYTE	E3FD	A	I	Reads and displays two characters from the active input device. If the input is a hex number in ASCII format, the numbers are converted to hex format and packed into one byte

Table 7-13. AIM 65 Monitor Subroutines (Cont.)

SUBRTN NAME	ENTRY ADDR.	REG ALT.	I O	FUNCTION
RCHEK	E907	AXY	I	The keyboard is scanned. If no key is depressed return is to the calling routine. If <ESC> is depressed, return is to the Monitor. If <SPACE> is depressed, the subroutine waits until another key is depressed to continue.
RDRUB	E95F	A Y	I	Reads one character from the keyboard. Echos the input code to the display/printer. Allows rubout to delete the character if Y ≠ 0.
READ	E93C	A	I	Reads one ASCII character from the keyboard. Returns with the ASCII code in A.
RED1	FE96	A	I	Reads one ASCII character from the keyboard. Echoes the input code to the display/printer, unless the character is a CR ($0D).
REDOUT	E973	A	I	Reads one ASCII character from the keyboard. Echos the input code to the display/printer if the character is not a CR ($0D). Displays cursor upon return.
SEMI	E9BA	A	O	Outputs one ";" ($3B) to the active output device.
TAISET	EDEA		M	Sets up recorder 1 or 2 input depending on the state of TAPIN ($A434).

Table 7-13. AIM 65 Monitor Subroutines (Cont.)

SUBRTN NAME	ENTRY ADDR.	REG ALT.	I O	FUNCTION
TAISET	EDEA		M	(Cont.) Checks for SYN ($16) character. Returns to calling routine upon detection of five consecutive SYN characters.
TAOSET	F21D		O	Sets up recorder 1 or 2 output depending upon the state of TAPOUT ($A435). Outputs the number of SYN ($16) characters equal to GAP ($A409) times 4.
TIBYTE	ED3B	A		Loads one input character byte from the audio tape buffer into A. Inputs a block of data from the recorder if the tape buffer is empty.
TIBY1	ED53		I	Loads a block of 80 bytes from audio tape into the tape buffer when BLK ($0115) = 0. The desired tape block number is stored in TABUFF ($0116) before calling TIBY1.
TOBYTE	F18B	A		Stores one output character (byte) in the audio tape buffer. Outputs a block of data to the recorder if the tape buffer is full.
TO	E7A7	AXY	O	Outputs "TO" to the display/printer and puts the entered address in ADDR ($A41C) and ADDR+1 ($A41D).

Table 7-13. AIM 65 Monitor Subroutines (Cont.)

SUBRTN NAME	ENTRY ADDR.	REG ALT.	I O	FUNCTION
WHEREI	E848	AXY	I	Determines and sets up the active input device from the answer to "IN=" and puts the device code into INFLG ($A412): T ($54) AIM 65 Audio Tape K ($4B) KIM-1 Audio Tape U ($55) User Defined. X ($58) Dummy (no output) OTHER Display/Printer
WHEREO	E871	AXY	O	Determines and sets up the active output device from the answer to "OUT=" and puts the device code into OUTFLG ($A413): T ($54) AIM 65 Audio Tape K ($4B) KIM 1 Audio Tape P ($50) Printer U ($55) User Defined X ($58) Dummy (no output) OTHER Display/Printer

7.7.1 Character Input Subroutine Considerations

The character input subroutines provide different responses to the AIM 65 display and input devices., The following table identifies which subroutines display a cursor (^) before reading the input character and/or echo the input character to the input device:

SUBROUTINE	DISPLAY CURSOR	ECHO CHARACTER	ALLOWS DELETE
CUREAD	Yes	No	No
READ	No	No	No
REDOUT	Yes	Yes	No
RED1	No	Yes	No
RDRUB	Yes	Yes	Yes

7.7.2 CR and LF Output Subroutine Considerations

The CR and LF subroutines output different combinations of CR, LF, and NUL (if TTY) characters depending upon the output device code:

	<RETURN> or <SPACE> (3)						
	S3=KB		S3=TTY				
Subroutine	Display	Printer	TTY	T	K	P	U
CRLF (1)	CR	CR	CR, LF, NUL	CR	CR	CR	CR, LF
CRLOW (2)	CR	CR	CR, LF, NUL				
CRCK (2)		CR	CR, LF, NUL				

Notes: (1) To active device
(2) To input terminal only
(3) Or any character except T, K, P, or U

Example: Display and print a message - entry from F1 and return to monitor.

a. Source listing

```
=<L>
/.
OUT=
*=$010C
JMP F1
;
OUTPUT=$E97A
CLR=$EB44
*=$0300
;
F1 JSR CLR
LDX #$FF
LOOP INX
LDA MSG1,X
CMP #';'
BEQ RET
JSR OUTPUT
JMP LOOP
RET RTS
;
MSG1 .BYT 'APPLICATI
ON'
.BYT ' MESSAGE);'
.END
```

b. Disassembly listing

```
<K>*=010C
/01
 010C 4C JMP 0300
<K>*=0300
/
 0300 20 JSR EB44
 0303 A2 LDX #FF
 0305 E8 INX
 0306 BD LDA 0314,X
 0309 C9 CMP #3B
 030B F0 BEQ 0313
 030D 20 JSR E97A
 0310 4C JMP 0305
 0313 60 RTS
```

7.8 AIM 65 INTERRUPT LINKAGE AND HANDLING

The AIM 65 Monitor uses the three available interrupts: Reset (RES), Non-Maskable Interrupt (NMI) and Interrupt Request (IRQ). Each will occur under in response to an activation of one of the three input lines of the R6502 CPU; RES, NMI, and IRQ, respectively. Reset is used only by AIM 65. The user may bypass the AIM 65 NMI and IRQ interrupt routines and link directly to user provided interrupt routiness. Alternatively, with the IRQ interrupt, the user may link to a user provided IRQ interrupt routine after interrupt processing by the AIM 65 IRQ interrupt routine.

Before proceeding you should read Section 9 in the R6500 Programming Manual ("Reset and Interrupt Considerations") to gain insight into the R6502 interrupt processing features.

In response to any one of the interrupts, the R6502 CPU will fetch two bytes of data stored at a specific pair of addresses and load the fetched data into the program counter. The two bytes of data constitute the entry address of the corresponding interrupt routine. The R6502 CPU will then continue processing, starting at the new address in the program counter. The three address pairs, called interrupt vectors, contain vectors, or pointers, to the interrupt routine entry addresses. The address of the interrupt vectors are hardware determined and are not under the control of the programmer. The fixed addresses are:

FFFA,	FFFB	-	NMI Interrupt Vector
FFFC,	FFFD	-	RES Interrupt Vector
FFFE,	FFFF	-	IRQ Interrupt Vector

RES Interrupt Handling

The RES vector points to the Reset Interrupt Routine entry point in the Monitor program ($E0BF). Upon AIM 65 power turn-on or depression of the RESET button, AIM 65 Monitor processing will begin at this address.

NMI Interrupt Handling

The NMI vector points to address $E075 (NMIVI), where a jump indirect through address $A402 (NMIV2) is coded. NMIV2 is a user-alterable variable, initialized upon a cold reset, that normally points to the AIM 65 NMI Interrupt Routine at $E07B (NMIV3).

The AIM 65 NMI Interrupt Routine performs single step processing when a user program is being executed and the RUN/STEP switch is in the STEP position. It is possible for the user to change the NMIV2 variable to point to a user provided NMI Interrupt Routine. If this is done, however, the single step processing will not be performed. Also, in this case, RUN/ STEP switch should be left in the RUN position, otherwise an NMI interrupt will be generated each time an instruction of the user program is executed.

IRQ Interrupt Handling

The IRQ interrupt vector points to address $E078 (IRQVI), which contains a jump indirect instruction through address $A404 (IRQV2). IRQV2 is a user-alterable variable, initialized upon a cold reset, that normally points to the AIM 65 IRQ Interrupt Routine.

The AIM 65 Interrupt Routine determines the cause of the interrupt. If the $\overline{IRQ}$ interrupt was caused by a BRK instruction and the RUN/STEP switch is set to RUN, the register values are saved, user program execution is terminated and control is returned to the Monitor. If the $\overline{IRQ}$ interrupt was caused by a BRK instruction and the RUN/STEP switch is set to STEP, user program execution continues.

If the $\overline{IRQ}$ interrupt was not due to a BRK instruction, it must have been caused by a user provided input. The IRQ Interrupt Routine then jumps indirectly to a user IRQ Interrupt Routine through address $A400 (IRQV4).

7.8.1 Monitor Subroutine Examples

The examples in this section show how the Monitor subroutines can be used in either program-controlled or operator-controlled procedures. Some of the Monitor input and output subroutines are similar. Experiment by replacing the given subroutines with other subroutines that appear nearly the same, to see the effect.

The examples are shown in AIM 65 assembly and disassembly form.

The equates to follow were programmed at the beginning of the assembly examples. Not all equates are required with each example, however.

```
==0000 CRCK
       =$EA24
==0000 CRLF
       =$E3F0
==0000 DUMPTA
       =$E56F
==0000 DU11
       =$E50A
==0000 OUTDIG
       =$EF05
```



```
==0000 OUTPRI
       =$F000
==0000 OUTPUT
       =$E97A
==0000 INALL
       =$E993
==0000 OUTALL
       =$E9BC
==0000 LOAD
       =$E2E6
==0000 LOAD1
       =$E2E9
==0000 LOADTA
       =$E32F
==0000 LL
       =$E8FE
==0000 RDRUB
       =$E95F
==0000 READ
       =$E93C
==0000 REDOUT
       =$E973
==0000 TIBYTE
       =$ED2B
==0000 TOBYTE
       =$F18B
==0000 WHEREI
       =$E848
==0000 WHERE0
       =$E871
==0000 PRIFLG
       =$A411
==0000 INFLG
       =$A412
==0000 OUTFLG
       =$A413
==0000 NAME
       =$A42E
==0000 TAPIN
       =$A434
==0000 TAPOUT
       =$A435
==0000 DRB
       =$A300
```

1. Display and print an assembled message - under program control. The printer control must be ON in order to print. The message is terminated by a semicolon. Type F1 to enter; the program returns to the Monitor upon completion.

(Aim 65 assembly format)

```
---------------------
FUNCTION 1 LINKAGE

==0200
        *=$010C
4C0002 JMP EX1
---------------------
MAIN PROGRAM

==010F
        *=$0200
==0200 EX1
20F0E9 JSR CRLF
A2FF   LDX #$FF
==0205 LOOP1
E8     INX
BD1702 LDA MSG1,X
C93B   CMP #';'
F006   BEQ PRT
20BCE9 JSR OUTALL
4C0502 JMP LOOP1
==0213 PRT
2024EA JSR CRCK
==0216 EX1R
60     RTS
---------------------
MESSAGE

==0217 MSG1
2055   .BYT ' USER M
ESSAGE;'
```


(AIM 65 disassembly format)

```
<K>*=010C
/
 010C 4C JMP 0200
<K>*=0200
/
 0200 20 JSR E9F0
 0203 A2 LDX #FF
 0205 E8 INX
 0206 BD LDA 0217,X
 0209 C9 CMP #3B
 020B F0 BED 0213
 020D 20 JSR E9EC
 0210 4C JMP 0205
 0213 20 JSR EA24
 0216 60 RTS
```

2. Input from keyboard and echo input character and cursor to the display, then output to the printer - under program control. Type F2 to enter. Type ESC return to the Monitor.

(AIM 65 assembly format)

```
---------------------
FUNCTION 2 LINKAGE

==0225
        *=$010F
4C3002 JMP EX2
---------------------
MAIN PROGRAM

==0112
        *=$0230
==0230 EX2
20F0E9 JSR CRLF
A90D   LDA #$0D
8D12A4 STA INFLG
A950   LDA #'P'
8D13A4 STA OUTFLG
==023D LOOP2
2071E9 JSR REDOUT
20BCE9 JSR OUTALL
4C3D02 JMP LOOP2
```

(AIM 65 disassembly format)

```
<K>*=010F
/
 010F 4C JMP 0230

<K>*=0230
/
 0230 20 JSR E9F0
 0233 A9 LDA #0D
 0235 8D STA A412
 0238 A9 LDA #50
 023A 8D STA A413
 023D 20 JSR E373
 0240 20 JSR E98C
 0243 4C JMP 023D
```

3. Input from keyboard without echoing input character and cursor to the display, followed by output to the display and printer -- under program control.

(AIM 65 assembly format)

```
---------------------
FUNCTION 3 LINKAGE

==0246
       *=$0112
4C5002 JMP EX3
---------------------
MAIN PROGRAM

==0115
       *=$0250
==0250 EX3
20F0E9 JSR CRLF
20FEE8 JSR LL
==0256 LOOP3
2073E9 JSR RDOUT
4C5602 JMP LOOP3
```


(AIM 65 disassembly format)

```
<K>*=0112
/
 0112 4C JMP 0250

<K>*=0250
/
 0250 20 JSR E9F0
 0253 20 JSR E8FE
 0256 20 JSR E973
 0259 4C JMP 0256
 025C 20 JSR E9F0
 025F A9 LDA #50
 0261 8D STA A413
 0264 20 JSR E92C
 0267 20 JSR E96C
 026A 4C JMP 0264
```

4. Input from a user-specified device and output to a user-specified device -- under operator control, excluding user-defined I/O. Type F1 to enter and ESC to exit. Note that typing RETURN in response to both IN= and OUT= prompts will display each entered digit twice, since the input is echoed to the display on entry as well as displayed to the printer on output.

(AIM 65 assembly format)

```
---------------------
FUNCTION 1 LINKAGE

==0000
       *=$010C
==010C
4C7302 JMP EX4
---------------------
MAIN PROGRAM

==010F
       *=$0270
==0270 EX4
20F0E9 JSR CRLF
2041E8 JSR WHERE
```

(AIM 65 assembly format) cont.

```
20F0E9 JSR CRLF
207AE8 JSR WHEREO
==027C LOOP4
209BE9 JSR INALL
207AE9 JSR OUTPUT
4C7C02 JMP LOOP4
```

(AIM 65 disassembly format)

```
<K>*=010C
/
 010C 4C JMP 0270
<K>*=0270
/
 0270 20 JSR E9F0
 0273 20 JSR E848
 0276 20 JSR E9F0
 0279 20 JSR E871
 027C 20 JSR E993
 027F 20 JSR E97A
 0282 4C JMP 027C
```

5. Input from the keyboard and output to the display and printer -- under program control, with user function linkage. Enter F2 to enter the program and ESC to return to the Monitor.

(AIM 65 assembly format)

```
--------------------
LINKAGE

;USER DEFINED INPUT
==0285
        *=$0103
A981    WOP IN5
;USER DEFINED OUTPUT
==010A
        *=$010A
A081    WOP OUT5
;USER DEFINED F2
```


(AIM 65 assembly format) cont.

```
==010C
        *=$010F
4C9002 JMP EX5
--------------------
MAIN PROGRAM

==0112
        *=$0290
==0290 EX5
20F0E9 JSR CRLF
A980   LDA #$80
8D11A4 STA PRIFLG
A955   LDA #'U'
8D12A4 STA INPLG
8D11A4 STA OUTFLG
==02A0 LOOP5
2093E9 JSR INALL
20B0E9 JSR OUTALL
4CA002 JMP LOOP5
--------------------
INPUT PROCESSING

==02A9 IN5
2003E9 JSR READ
60     RTS
--------------------
OUTPUT PROCESSING

==02AD OUT5
68     PLA
207AE9 JSR OUTPUT
60     RTS
```

(AIM 65 disassembly format)

```
<K>*=010F
/
 010F 4C JMP 0290
<K>*=0290
/
 0290 20 JSR E9F0
 0293 A9 LDA #80
 0295 8D STA A411
 0298 A9 LDA #55
 029A 8D STA A412
 029D 8D STA A413
 02A0 20 JSR E993
 02A3 20 JSR E98C
```

(AIM 65 disassembly format) cont.

```
02A6 4C JMP 02F0
02A9 20 JSR E93C
02AC 60 RTS
02AD 68 PLA
02AE 20 JSR E97A
02B1 60 RTS
```

6. Input from user-specified device and output to user-specified device -- under operator control, with user function linkage. User-defined I/O linkage is also included. Type F3 to enter. Type U in response to IN= and OUT= prompts, to link to user-defined I/O. Type ESC to return to the Monitor.

(AIM 65 assembly format)

```
--------------------
LINKAGE

;USER-DEFINED INPUT
==02B2
        *=$0109
C502    WDP IN6
;USER DEFINED OUTPUT
==010A
        *=$010A
C001    WDP OUT6
;USER DEFINED FI
==010C
        *=$0112
4CB002 JMP ENT
--------------------
MAIN PROGRAM

==0115
        *=$02B0
==02B0 ENT
20F0E9 JSR CRLF
2043E8 JSR WHERE1
20F0E9 JSR CRLF
207AE8 JSR WHERE2
==02BC LOOP6
20D7E9 JSR INALL
20BCE9 JSR OUTALL
```


(AIM 65 assembly format) cont.

```
4CBC02 JMP LOOP6
--------------------
INPUT PROCESSING

==02C5 IN6
B001   BCS IN6N
60     RTS
==02C8 IN6N
20D0E9 JSR READ
60     RTS
--------------------
OUTPUT PROCESSING

==02CC OUT6
B004   BCS OUT6N
20F0E9 JSR CRLF
60     RTS
==02D2 OUT6N
68     PLA
207AE9 JSR OUTPUT
60     RTS
```

(AIM 65 disassembly format)

```
<K>*=0112
/
 0112 4C JMP 02C0

<K>*=02C0
/
 02C0 20 JSR E9F0
 02C3 20 JSR E848
 02C6 20 JSR E9F0
 02C9 20 JSR E871
 02CC 20 JSR E992
 02CF 20 JSR E9BC
 02D2 4C JMP 02CC
 02D5 B0 BCS 02D8
 02D7 60 RTS
 02D8 20 JSR E93C
 02DB 60 RTS
 02DC B0 BCS 02E2
 02DE 20 JSR E9F0
 02E1 60 RTS
 02E2 68 PLA
 02E3 20 JSR E97A
 02E6 60 RTS
```

7. Inputs a file from audio cassette tape -- under program control. Enter with the 5-character file name stored in TAB7, and the tape recorder number stored in IN7 ($0=recorder #1, $1=recorder #2). Insert user-defined processing of input characters. Type F1 to enter. The program returns to the Monitor upon completion of input file processing.

(AIM 65 assembly format)

```
--------------------
FUNCTION 1 LINKAGE

==0000
        *=$010C
==010C
4C0E03 JMP EX7
--------------------
MAIN PROGRAM

==010F
        *=$0300
==0300 IN7
        *=*+1
==0301 TAB7
        *=*+5
==0306 EX7
A954    LDA #'T'
8D12A4 STA INFLG
AD0003 LDA IN7
8D34A4 STA TAPIN
A004    LDY #4
==0313 LOOP7
B90103 LDA TAB7,Y
992EA4 STA NAME,Y
88      DEY
10F7    BPL LOOP7
202FE3 JSR LOADTA
==031F LOOP7A
2093E9 JSR INALL
;INSERT USER
;PROCESSING OF INPUT
;CHARACTER---
```


(AIM 65 assembly format) cont.

```
;JMP TO EX7R UPON
;DETECTION OF END
;OF FILE
4C1F03 JMP LOOP7A
==0325 EX7R
A9CF    LDA #$CF
;TURN OFF RECORDERS
2D00A8 AND DRB
8D00A8 STA DRB
60      RTS
```

(AIM 65 disassembly format)

```
<K>*=010C
/
 010C 4C JMP 0306

<K>*=0306
/
 0306 A9 LDA #54
 0308 8D STA A412
 030B AD LDA 0300
 030E 8D STA A434
 0311 A0 LDY #04
 0313 B9 LDA 0301,Y
 0316 99 STA A42E,Y
 0319 88 DEY
 031A 10 BPL 0313
 031C 20 JSR E32F
 031F 20 JSR E993
 0322 4C JMP 031F
 0325 A9 LDA #CF
 0327 2D AND A800
 032A 8D STA A800
 032D 60 RTS
```

8. Outputs a file to audio tape -- under program control. Enter with the 5-character file name stored in TAB8 and the tape recorder number stored in OUT8 ($0=recorder #1, $1=recorder #2). Insert user-defined processing to generate output characters. Type F2 to enter. The program returns to the Monitor upon completion of output file processing.

(AIM 65 assembly format)

```
;----------------------
;FUNCTION 2 LINKAGE

==0128
        *=$010F
404E80 JMP EX8
;----------------------
;MAIN PROGRAM

==0112
        *=$0140
==0140 OUT8
        *=*+1
==0141 TAB8
        *=*+5
==0146 EX8
A954    LDA #'T'
8D12A4 STA OUTFLG
AD4080 LDA OUT8
8D22A4 STA TAPOUT
A004    LDY #4
==0152 LOOP8
B94180 LDA TAB8,Y
991EA4 STA NAME,Y
88      DEY
10F7    BPL LOOP8
206FE5 JSR DUMPTA
==015F LOOP99
;INSERT USER
;PROCESSING TO PUT
;OUTPUT CHARACTER
;IN A ---
;JMP TO EX8R UPON
;END OF OUTPUT DATA
20BCE8 JSR OUTALL
```


(AIM 65 assembly format) cont.

```
405F03 JMP LOOP8A
;CLOSE SOURCE FILE
;WITH TWO CR'S
==0165 EX8A
20F0E9 JSR CRLF
20F0E9 JSR CRLF
200AE5 JSR DU11
A9CF    LDA #$CF
;TURN OFF RECORDER#
2D00A8 AND ORB
8D00A8 STA ORB
==0175
60      RTS
```

(AIM 65 disassembly format)

```
<K>*=010F
/
 010F 4C JMP 0346

<K>*=0346
/
 0346 A9 LDA #54
 0348 8D STA A412
 034B AD LDA 0140
 034E 8D STA A425
 0351 A0 LDY #04
 0353 B9 LDA 0141,Y
 0356 99 STA A42E,Y
 0359 88 DEY
 035A 10 BPL 0353
 035C 20 JSR E56F
 035F 20 JSR E9B0
 0362 4C JMP 035F
 0365 20 JSR E9F0
 0368 20 JSR E9F0
 036B 20 JSR E50A
 036E A9 LDA #CF
 0370 2D AND A800
 0373 8D STA A800
 0376 60 RTS
```