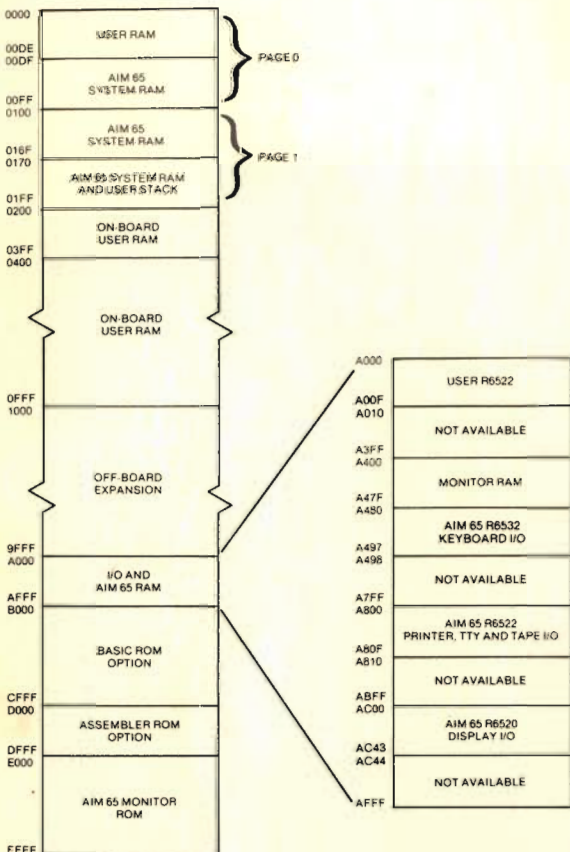


## AIM 65 MEMORY MAP



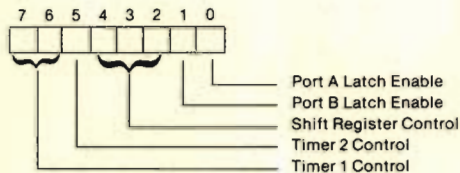
## AIM65 USER-ALTERABLE ADDRESSES

Location	Name	Bytes	Description
0108	UIN	2	Vector to User Input Handler
010A	UOUT	2	Vector to User Output Handler
010C	KEYF1	3	JMP to User Function 1
010F	KEYF2	3	JMP to User Function 2
0112	KEYF3	3	JMP to User Function 3
A400	IRQV4	2	Vector to IRQ after Monitor Interrupt Routine
A402	NMIV2	2	Vector to NMI Interrupt Routine
A404	IRQV2	2	Vector to IRQ Interrupt Routine
A406	DILINK	2	Vector to Display Routine
A408	TSPEED	1	Audio Tape Speed Default = \$C7 (AIM 65) Options = \$5A (KIM-1 x 1) \$5B (KIM-1 x 3)
A409	GAP	1	Audio Tape Gap Default = \$08 = 32 SYN characters Option = \$80 for Assembler input & Editor update

## USER R6522 VERSATILE INTERFACE ADAPTER (VIA) R6522 MEMORY ASSIGNMENTS

Location	Function
A000	Port B Output Data Register (ORB)
A001	Port A Output Data Register (ORA)
A002	Port B Data Direction Register (DDRB)
A003	Port A Data Direction Register (DDRA)
	Timer R/W = L R/W = H
A004	T1 Write T1L-L Read T1C-L Clear T1 Interrupt Flag
A005	T1 Write T1L-H & T1C-H T1L-L → T1C-L Clear T1 Interrupt Flag Read T1L-L
A006	T1 Write T1L-L Read T1L-H
A007	T1 Write T1L-H Clear T1 Interrupt Flag
A008	T2 Write T2L-L Read T2C-L Clear T2 Interrupt Flag
A009	T2 Write T2C-H T2L-L → T2C-L Clear T2 Interrupt Flag Read T2C-H
A00A	Shift Register (SR)
A00B	Auxiliary Control Register (ACR)
A00C	Peripheral Control Register (PCR)
A00D	Interrupt Flag Register (IFR)
A00E	Interrupt Enable Register (IER)
A00F	Port A Output Data Register (ORA) <i>No effect on handshake</i>

## R6522 AUXILIARY CONTROL REGISTER (ACR)



### PORT A LATCH ENABLE

ACR0 = 1 Port A latch is enabled to latch input data when CA1 Interrupt Flag (IFR1) is set.  
= 0 Port A latch is disabled, reflects current data on PA pins.

### PORT B LATCH ENABLE

ACR1 = 1 Port B latch is enabled to latch the voltage on the pins for the input lines or the ORB contents for the output lines when CB1 Interrupt Flag (IFR4) is set.  
= 0 Port B latch is disabled, reflects current data on PB pins.

### SHIFT REGISTER CONTROL

ACR4	ACR3	ACR2	Mode
0	0	0	Shift Register Disabled.
0	0	1	Shift in under control of Timer 2.
0	1	0	Shift in under control of $\emptyset$ 2.
0	1	1	Shift in under control of external clock.
1	0	0	Free-running output at rate determined by Timer 2.
1	0	1	Shift out under control of Timer 2.
1	1	0	Shift out under control of $\emptyset$ 2.
1	1	1	Shift out under control of external clock.

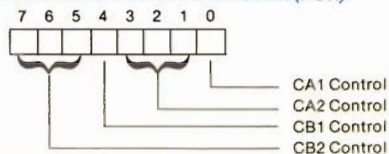
### TIMER 2 CONTROL

ACR5 = 0 T2 acts as an interval timer in the one-shot mode.  
= 1 T2 counts a predetermined number of pulses on PB6.

### TIMER 1 CONTROL

ACR7	ACR6	Mode
0	0	T1 one-shot mode — Generate a single time-out interrupt each time T1 is loaded. Output to PB7 disabled.
0	1	T1 free-running mode — Generate continuous interrupts. Output to PB7 disabled.
1	0	T1 one-shot mode — Generate a single time-out interrupt and an output pulse on PB7 each time T1 is loaded.
1	1	T1 free-running mode — Generate continuous interrupts and a square wave output on PB7.

## R6522 PERIPHERAL CONTROL REGISTER (PCR)



### CA1 CONTROL

PCR0 = 0 The CA1 Interrupt Flag (IFR1) will be set by a negative transition (high to low) on the CA1 pin.  
 = 1 The CA1 Interrupt Flag (IFR1) will be set by a positive transition (low to high) on the CA1 pin.

### CA2 CONTROL

PCR3	PCR2	PCR1	Mode
0	0	0	CA2 negative edge interrupt (IFR0/ORC clear) mode — Set CA2 interrupt flag (IFR0) on a negative transition of the CA2 input signal. Clear IFR0 on a read or write of the ORC or by writing logic 1 into IFR0.
0	0	1	CA2 negative edge interrupt (IFR0 clear) mode — Set IFR0 on a negative transition of the CA2 input signal. Clear IFR0 by writing logic 1 into IFR0.
0	1	0	CA2 positive edge interrupt (IFR0/ORC clear) mode — Set CA2 interrupt flag (IFR0) on a positive transition of the CA2 input signal. Clear IFR0 on a read or write of the ORC or by writing logic 1 into IFR0.
0	1	1	CA2 positive edge interrupt (IFR0 clear) mode — Set IFR0 on a positive transition of the CA2 input signal. Clear IFR0 by writing logic 1 into IFR0.
1	0	0	CA2 handshake output mode — Set CA2 output low on a read or write of the Peripheral A Output Register. Reset CA2 high with an active transition on CA1.
1	0	1	CA2 pulse output mode — CA2 goes low for one cycle following a read or write of the Peripheral A Output Register.
1	1	0	CA2 low output mode — The CA2 output is held low in this mode.
1	1	1	CA2 high output mode — The CA2 output is held high in this mode.

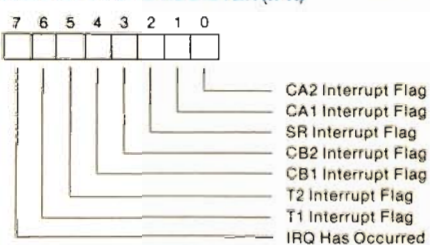
### CB1 CONTROL

PCR4 = 0 The CB1 Interrupt Flag (IFR4) will be set by a negative transition (high to low) on the CB1 pin.  
 = 1 The CB1 Interrupt Flag (IFR4) will be set by a positive transition (low to high) on the CB1 pin.

### CB2 CONTROL

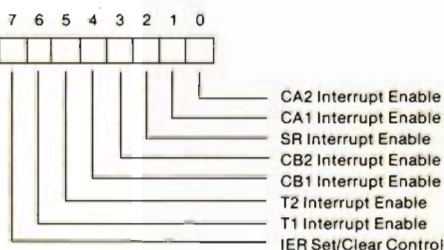
PCR7	PCR6	PCR5	Mode
0	0	0	CB2 negative edge interrupt (IFR3/ORB clear) mode — Set CB2 interrupt flag (IFR3) on a negative transition of the CB2 input signal. Clear IFR3 on a read or write of the ORB or by writing logic 1 into IFR3.
0	0	1	CB2 negative edge interrupt (IFR3 clear) mode — Set IFR3 on a negative transition of the CB2 input signal. Clear IFR3 by writing logic 1 into IFR3.
0	1	0	CB2 positive edge interrupt (IFR3/ORB clear) mode — Set CB2 interrupt flag (IFR3) on a positive transition of the CB2 input signal. Clear IFR3 on a read or write of the ORB or by writing logic 1 into IFR3.
0	1	1	CB2 positive edge interrupt (IFR3 clear) mode — Set IFR3 on a positive transition of the CB2 input signal. Clear IFR3 by writing logic 1 into IFR3.
1	0	0	CB2 handshake output mode — Set CB2 output low on a write of the Peripheral B Output Register. Reset CB2 high with an active transition on CB1.
1	0	1	CB2 pulse output mode — CB2 goes low for one cycle following a read or write of the Peripheral B Output Register.
1	1	0	CB2 low output mode — The CB2 output is held low in this mode.
1	1	1	CB2 high output mode — The CB2 output is held high in this mode.

## R6522 INTERRUPT FLAG REGISTER (IFR)



IFR Bit	Set By	Cleared By
0	Active transition on CA2	Reading or writing the ORC (\$A001 or \$A00F)
1	Active transition on CA1	Reading or writing the ORC (\$A001 or \$A00F)
2	Completion of eight shifts	Reading or writing the SR (\$A00A)
3	Active transition on CB2	Reading or writing the ORB (\$A000)
4	Active transition on CB1	Reading or writing the ORB (\$A000)
5	Time-out of Timer 2	Reading T2C-L (\$A008) or writing T2C-H (\$A009)
6	Time-out of Timer 1	Reading T1C-L (\$A004) or writing T1L-H (\$A005 or \$A007)
7	Any IFR bit set with its corresponding IER bit also set	Clearing IFR0-IFR6 (\$A00D) or IER0-IER6 (\$A00E)

## R6522 INTERRUPT ENABLE REGISTER (IER)




### INTERRUPT ENABLE BITS (IER0-6)

IERn = 0 Disable interrupt  
 = 1 Enable interrupt

### IER SET/CLEAR CONTROL (IER7)

IER7 = 0 For each data bus bit set to logic 1, clear corresponding IER bit  
 = 1 For each data bus bit set to logic 1, set corresponding IER bit.

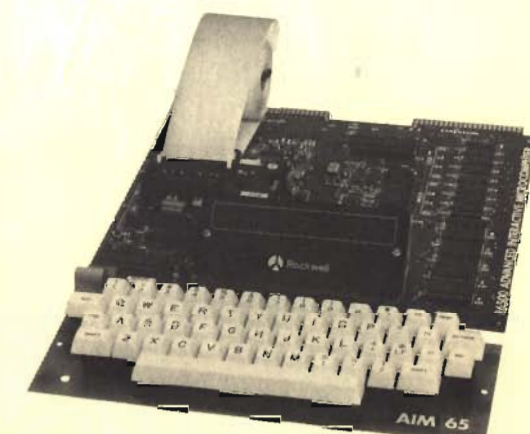
**Note:** IER7 is active only when R/W = L; when R/W = H, IER7 will read logic 1.



# Rockwell

## AIM 65

### Summary Card



## Rockwell Microelectronic Devices

### Sales Offices

**WESTERN REGION, U.S.A.**  
 3316 Miraloma Avenue  
 P. O. Box 3669  
 Anaheim, CA 92803  
 Phone: (714) 832-3698

**CENTRAL REGION, U.S.A.**  
 Contact: Robert O. Whitesell & Associates  
 6891 East Washington Street  
 Indianapolis, Indiana 46219  
 Attn: Milt Gamble, Mgr.  
 Phone: (317) 359-9283

**EASTERN REGION, U.S.A.**  
 Carolier Office Building  
 850-870 U.S. Route 1  
 North Brunswick, New Jersey 08902  
 Phone: (201) 246-3630

**FAR EAST**  
 Rockwell International Overseas Corp.  
 Ichiban cho Central Building  
 22-1 Ichiban-cho, Chiyoda-ku  
 Tokyo 102, Japan  
 Phone: 265-8808

**MIDWEST REGION, U.S.A.**  
 1011 E. Touhy - Suite 245  
 Des Plaines, Illinois 60018  
 Phone: (312) 297-8862

**EUROPE**  
 Rockwell International GmbH  
 Microelectronic Devices  
 Fraunhoferstrasse 11  
 D-8023 München-Martinsried  
 Germany  
 Phone: (089) 859-9575

DOC. NO. 29650N51  
 REV. 1, MARCH 1979

## AIM 65 MONITOR COMMANDS

### MAJOR FUNCTION ENTRY COMMANDS

- [RESET] — Enter and Initialize Monitor  
ROCKWELL AIM 65
- E — Enter and Initialize Editor  
<E>
- T — Re-enter Text Editor at Top of Text  
<T>  
TOP LINE OF TEXT
- N — Enter Assembler  
<N>
- 5 — Enter and Initialize BASIC Interpreter  
<5>
- 6 — Re-enter BASIC Interpreter  
<6>

### INSTRUCTION ENTRY AND DISASSEMBLY COMMANDS

- I — Enter Mnemonic Instruction Entry Mode  
<I>  
AAAA [\*] = [ADDRESS]  
AAAA XX [OPCODE][HEX OPERAND]  
AAAA XX XX XX
- K — Disassemble Memory  
<K>\* = [ADDRESS]  
/[DECIMAL NUMBER]  
AAAA XX OPCODE HEX OPERAND

### DISPLAY/ALTER REGISTER COMMANDS

- \* — Alter Program Counter  
<\*> = [ADDRESS]
- A — Alter Accumulator  
<A> = [BYTE]
- X — Alter X Register  
<X> = [BYTE]
- Y — Alter Y Register  
<Y> = [BYTE]
- P — Alter Processor Status  
<P> = [BYTE]
- S — Alter Stack Pointer  
<S> = [BYTE]
- R — Display Register Values  
<R>  
\*\*\*\* PS AA XX YY SS  
0200 00 00 01 02 FF

### DISPLAY/ALTER MEMORY CONTENTS

- M — Display Specified Memory Locations  
<M> = [ADDRESS] XX XX XX XX
- SPACE — Display Next 4 Memory Locations  
< >AAAA XX XX XX XX
- / — Alter Current Memory Locations  
</>AAAA XX XX XX XX

### LOAD/DUMP MEMORY COMMANDS

- L — Load Object Code into Memory  
<L>IN = [INPUT DEVICE]
- D — Dump Memory  
<D>  
FROM = [ADDRESS] TO = [ADDRESS]  
OUT = [OUTPUT DEVICE]  
MORE? [Y, N]

### BREAKPOINT MANIPULATION COMMANDS

- # — Clear All Breakpoints  
<#> OFF
- 4 — Toggle Breakpoint Enable  
<4> OFF/ON
- B — Set/Clear Breakpoint Address  
<B> BRK/[0, 1, 2, 3] = [ADDRESS]
- ? — Display Breakpoint Addresses  
<?>  
AAAA AAAA AAAA AAAA

## AIM 65 MONITOR COMMANDS (Continued)

### EXECUTION/TRACE CONTROL COMMANDS

- G — Start Execution of User's Program  
<G>/[DECIMAL NUMBER]
- Z — Toggle Instruction Trace Mode  
<Z> ON/OFF
- V — Toggle Register Trace Mode  
<V> ON/OFF
- H — Trace Program Counter History  
<H>  
AAAA  
:  
AAAA

### CONTROL PERIPHERAL DEVICES

- CTRL PRINT — Toggle Printer On/Off  
<CTRL><PRINT>
- PRINT — Print Display Contents  
<PRINT>
- LF — Advance Printer Paper  
<LF>
- 1 — Toggle Tape 1 Control On/Off  
<1>
- 2 — Toggle Tape 2 Control On/Off  
<2>
- 3 — Tape Verify Block Checksum  
<3>IN = [T] F = [FILE NAME] T = [1, 2]

### USER FUNCTION COMMANDS

- F1 — Call User Function 1  
<F1>
- F2 — Call User Function 2  
<F2>
- F3 — Call User Function 3  
<F3>

### AIM 65 COMMAND DEFINITIONS

- [ADDRESS] Hexadecimal address, one to four characters
- [BYTE] Two-digit hexadecimal value from 00 to FF.
- [DECIMAL NUMBER] A two-digit decimal number in the range 00 to 99.
- [FILE NAME] A string of 1 to 5 characters.
- [HEX OPERAND] The instruction operand.

#### Addressing Mode Operand Format

Addressing Mode	Operand Format
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
(Indirect, X)	(HH,X) or (HHX) or (HH,X) or (HHX)
(Indirect, Y)	(HH,Y) or (HHY)
(Absolute Indirect)(HHHH)	

- [INPUT DEVICE] RETURN or SPACE — AIM 65 Keyboard (S2 = KB) or TTY Keyboard (S2 = TTY)
  - M — Memory
  - T — Audio Tape, AIM 65 format
  - K — Audio Tape, KIM-1 format
  - L — TTY Paper Tape Reader
  - U — User-defined input device
- [MNEMONIC OPCODE] A three-letter mnemonic abbreviation.
- [OUTPUT DEVICE] RETURN or SPACE — AIM 65 Display/Printer (S2 = KB) or TTY Printer (S2 = TTY)
  - P — AIM 65 Printer
  - X — Dummy
  - T — Audio Tape, AIM 65 format
  - K — Audio Tape, KIM-1 format
  - L — TTY Paper Tape Punch
  - U — User-defined output device

## AIM 65 TEXT EDITOR COMMANDS

### ENTER AND EXIT EDITOR COMMANDS

- E — Enter and Initialize Editor  
 <E>  
 EDITOR  
 FROM = [ADDRESS] TO = [ADDRESS]  
 IN = [INPUT DEVICE]  
 Note: Defaults are TO = \$0200,  
 FROM = Last contiguous RAM, IN = Keyboard
- Q — Exit the Text Editor and Return to Monitor  
 = <Q>

### LINE ORIENTED COMMANDS

- R — Read Lines into Text Buffer from Input Device  
 = <R>  
 IN = [INPUT DEVICE]
- I — Insert One Line of Text Ahead of Active Line  
 = <I>  
 INSERTED TEXT LINE  
 ACTIVE LINE OF TEXT
- K — Delete Current Line of Text  
 = <K>  
 DELETED LINE OF TEXT  
 ACTIVE LINE OF TEXT
- U — Move the Text Pointer Up One Line  
 = <U>  
 PRIOR LINE OF TEXT
- D — Move the Text Pointer Down One Line  
 = <D>  
 NEXT LINE OF TEXT
- T — Move the Text Pointer to the Top of the Text  
 = <T>  
 TOP LINE OF TEXT
- B — Move the Text Pointer to the Bottom of the Text  
 = <B>  
 BOTTOM LINE OF TEXT
- L — List Lines of Text to Output Device  
 = <L>  
 / [DECIMAL NUMBER]
- SPACE — Display the Active Line  
 = < >  
 ACTIVE LINE OF TEXT

### STRING ORIENTED COMMANDS

- F — Find a Character String  
 = <F>  
 [CHARACTER STRING]  
 LINE CONTAINING CHARACTER STRING
- C — Change a Character String  
 = <C>  
 [OLD STRING]  
 LINE CONTAINING OLD STRING  
 TO = [NEW STRING]  
 SAME LINE, WITH NEW STRING

### ASSEMBLER ERROR CODES

- 01 Undefined Symbol  
 02 Label Previously Defined or Forward Reference to Page 0 Symbol  
 03 Illegal or Missing Opcode  
 04 Address Not Valid  
 05 Accumulator Mode Not Allowed  
 06 Forward Reference to Page Zero  
 07 Ran off End of Line  
 08 Label Does Not Begin with Alphabetic Character  
 09 Label Greater Than Six Characters  
 10 Label or Opcode Contains Non-Alphanumeric  
 11 Forward Reference in Equate  
 12 Invalid Index — Must Be X or Y  
 13 Invalid Expression  
 14 Undefined Assembler Directive  
 15 Invalid Page 0 Operand  
 17 Relative Branch Out of Range  
 18 Illegal Operand Type for This Instruction  
 19 Out of Bounds on Indirect Addressing  
 20 A, X, Y, S and P are Reserved Labels  
 21 Program Counter Negative — Reset to 0

## AIM 65 ASSEMBLER

### ASSEMBLER COMMAND SUMMARY

<N>  
 ASSEMBLER  
 FROM = [ADDRESS] TO = [ADDRESS]  
 IN = [INPUT DEVICE]  
 LIST? [Y, N]  
 LIST-OUT = [OUTPUT DEVICE]  
 OBJ? [Y, N] Note: N = Object code to Memory  
 OBJ-OUT = [OUTPUT DEVICE] Note: Prompts only on Y response to OBJ?  
 PASS 1  
 SYM TBL OVERFLOW } Displayed only if Symbol Table overflows  
 ASSEMBLER }  
 PASS 2  
 = AAAA LABEL }  
 OBJECT CODE MNEMONIC OP CODE } Displayed only if  
 SYMBOLIC OPERAND COMMENT } LIST?Y, or LIST?N  
 and error detected  
 \* \* ERROR NN Note: Error code displayed only on error  
 ERRORS = MMMM Decimal count of errors detected

### ASSEMBLER EXPRESSIONS

#### ELEMENTS

Numeric constants — may be written in one of four bases.

Prefix Character	Base
(none)	10 (Decimal)
\$	16 (Hexadecimal)
@	8 (Octal)
%	2 (Binary)

#### OPERATORS

Type	Operator	Operation
Arithmetic	+	Addition
Arithmetic	-	Subtraction
Special	>	High-Byte Selection
Special	<	Low-Byte Selection

Operators < and > truncate a two-byte value to its high or low byte, respectively.

### ASSEMBLER DIRECTIVES

- = — Assigns the value of an operand containing no forward references to either a symbol or the location counter.  
 { SYMBOL } = Operand
- .BYTE — Assigns multiple ASCII strings or expressions to consecutive single byte memory locations in high-byte, low-byte order.  
 .BYT Expression, Expression, ... Expression
- .WORD — Assigns multiple expression operands to consecutive memory locations in low-byte, high-byte order.  
 .WOR Expression, Expression, ... Expression
- .DBYTE — Assigns multiple expression operands to consecutive double byte (16 bits) memory locations.  
 .DBY Expression, Expression, ... Expression
- .PAGE — Generates a title under a dashed line.  
 .PAG { NEW TITLE } (New Title)  
 { BLANK } (No Change of Title)  
 { } (Blanks Title)
- .SKIP — Generates one blank line.  
 .SKI
- .OPT — Controls assembly listings. All are optional and can be specified in any order or in separate statements.  
 .OPT { LIS } { GEN } { ERR }  
 { NOL } { NOG } { NOE }
- .FILE — Last record in a multiple file source program (except the last file) which points to the continuation file.  
 .FIL File Name
- .END — Last record in a single or multiple source file.  
 .END

## AIM 65 SUBROUTINE SUMMARY

Sub. Name	Entry Addr.	Registers Altered	Function
BLANK	E83E	A	Outputs one SP to D/P
BLANK2	E83B	A	Outputs two SP's to D/P
CLR	E844	A	Clears D/P pointers
CRCK	EA24	A	Outputs print buffer to Printer
CRLF	E9F0	A	Outputs CR, LF & NUL to AOD
CRLOW	EA13	A	Outputs CR & LF to D/P
CUREAD	FE83	A	Inputs one ASCII character from KB to A, displays cursor
DISASM	F46C	A	Outputs disassembled instruction
DUMPTA	E56F	A	Opens Audio Tape output file
EQUAL	E7D8	A	Outputs "=" to D/P
FROM	E7A3	A,X,Y	Outputs "FROM=" to D/P and enters address
GETTAP	EE29	A,Y	Inputs one character from Audio Tape
HEX	EA7D	A	Converts a hex number in A from ASCII to binary, and puts result in the LSD of A, with zero in MSD of A.
INALL	E993	A	Inputs one ASCII character from AID to A
INLOW	E8F8	A	Indicates KB input in INFLG
LL	E8FE	A	Sets input from KB and output to D/P
LOADTA	E32F	A	Searches for audio input file
NOUT	EA51	A	Converts a hex number in LSD of A from binary to ASCII, and outputs it to AOD.
NUMA	EA46	A	Converts two hex numbers in A from binary to ASCII, and outputs them to AOD, MSD first.
OUTALL	E9BC	A	Outputs ASCII character in A to AOD
OUTPUT	E97A	A	Outputs ASCII character in A to D/P
OUTTAP	F24A	Y	Outputs ASCII character in A to Audio Tape after SYN
PACK	EA84	A	Converts a hex number in A from ASCII to binary, and puts result in the LSD of A, with the result of the last call to PACK or HEX in the MSD of A.
PHXY	EB9E	A	Push X and Y Registers onto Stack
PLXY	EBAC	X,Y	Pull X and Y Registers from Stack
PSL1	E837	A	Outputs "?" to D/P
QM	E7D4	A	Outputs "?" to D/P
RBYTE	E3FD	A	Inputs two ASCII characters from AID, and converts to binary with result in A
RCHEK	E907	A,X,Y	Scans KB, returns to Monitor on ESC, to caller on no entry, wait on SP
RDRUB	E95F	A,Y	Inputs one ASCII character from KB to A, with echo to D/P. Allows DEL, if Y ≠ 0.
READ	E93C	A	Inputs one ASCII character from KB to A
RED1	FE96	A	Inputs one character from KB, with echo to D/P
REDOUT	E973	A	Inputs one ASCII character from KB to A, with echo to D/P, displays cursor
SEM!	E9BA	A	Outputs ";" to AOD
TAISET	EDEA	A	Sets up Audio Tape input, detects five SYN characters
TAOSET	F21D	A	Sets up Audio Tape output, issues SYN characters
TIBY1	ED53	A	Loads a block of 80 bytes from Audio Tape
TO	E7A7	A,X,Y	Outputs "TO" to D/P and enters address
WHEREI	E848	A,X,Y	Sets up the AID and loads INFLG
WHEREO	E871	A,X,Y	Sets up the AOD and loads OUTFLG

#### ABBREVIATIONS

D/P = Display/Printer  
 AOD = Active Output Device  
 AID = Active Input Device