

# KIM-1 Quick Reference

## For the MOS Technology KIM-1 Microcomputer Module

Jeff Tranter <[tranter@pobox.com](mailto:tranter@pobox.com)>

### NMI Initialization for Single Step and Stop:

17FA 00  
17FB 1C

### IRQ Initialization for BRK:

17FE 00  
17FF 1C

### Machine context (saved/restored by ST/GO):

00EF PC low  
00F0 PC high  
00F1 Status Register (flags)  
00F2 Stack Pointer  
00F3 A  
00F4 Y  
00F5 X

### Cassette Load and Save

Note: 12V power is required when reading tapes.

#### To save:

1. Store \$00 in \$00F1 (to ensure CPU is in decimal mode).
2. Save start address (low/high) in \$17F5, \$17F6.
3. Save end address+1 (low/high) in \$17F7, \$17F8.
4. Write tape ID (\$01-\$FE) in \$17F9.
5. Start tape in record mode.
6. Run address \$1800 (DUMPT) to save.

#### To load:

1. Store \$00 in \$00F1 (to ensure CPU is in decimal mode).
2. Write tape ID (\$01-\$FE, \$00 loads any ID, \$FF loads using start address values) to \$17F9.
3. Run address \$1873 (LOADT) to load.

### Teleprinter Commands

Press <Rubout> or <Delete> after Reset to initialize serial bit rate.

<hex address> <space>	Show data at address
<hex data> .	Write to current address
<Return>	Advance to next address
<Line Feed>	Move to previous address
<Rubout>	Terminate memory edit
L	Load program from paper tape
Q	Save memory to paper tape (saves from current address to \$17F7, \$17F8)
G	Go from current address

## Memory Map

Range	Comments
\$0000-\$00FF	RAM - page zero (\$00EF, \$00FF are reserved)
\$0100-\$01FF	RAM - stack
\$0200-\$03FF	RAM - user programs
\$0400-\$16FF	Reserved for RAM expansion
\$1700-\$17FF	RAM/IO/TIMER chips
\$1700-\$173F	Application I/O and timer
\$1740-\$177F	KIM I/O and timer
\$1780-\$17BF	Application RAM
\$17C0-\$17E6	Application RAM
\$17E7-\$17FF	KIM RAM
\$1800-\$1FFF	ROM (2K)
\$2000-\$FFFF	Reserved for expansion

## Useful ROM Routines

Name	Address	Description
AK	\$1EFE	Check for key depressed. A non-zero: no key down. A equal 0, key down.
CRLF	\$1E2F	Send CRLF to TTY.
GETBYT	\$1FD9	Get two hex characters from TTY and return them packed in A.
GETCH	\$1E5A	Get one ASCII character from TTY and return in A.
GETKEY	\$1F6A	Return key from keyboard. Value 0-F, 10(AD), 11(DA), 12(+), 13(GO), 14(PC), 15 (no keypress).
OUTCH	\$1EA0	Print ASCII character in A on TTY.
OUTSP	\$1E9E	Print space on TTY.
PRTBYT	\$1E3B	Prints A as two hex characters.
PRTPNT	\$1E1E	Prints contents of \$00FB, \$00FA on TTY.
SCANDS	\$1F1F	Output six hex characters on display. Stored in \$00F9, \$00FA, \$00FB.

**Connector Pinouts**

<b>Connector A (lower) Application Connector. Pins A-Z on bottom, 1-22 on top</b>			
<b>Pin</b>	<b>Signal</b>	<b>Pin</b>	<b>Signal</b>
22	KB Col D	Z	KB Row 1
21	KB Col A	Y	KB Col C
20	KB Col E	X	KB Row 2
19	KB Col B	W	KB Col G
18	KB Col F	V	KB Row 3
17	KB Row 0	U	TTY PTR
16	PB5	T	TTY KYBD
15	PB7	S	TTY PTR RTRN(+)
14	PA0	R	TTY KYBD RTRN(+)
13	PB4	P	AUDIO OUT HI
12	PB3	N	+12V
11	PB2	M	AUDIO OUT LO
10	PB1	L	AUDIO IN
9	PB0	K	DECODE ENAB
8	PA7	J	K7
7	PA6	H	K5
6	PA5	F	K4
5	PA4	E	K3
4	PA1	D	K2
3	PA2	C	K1
2	PA3	B	K0
1	VSS GND	A	VCC +5V
<b>Connector B (upper) Expansion Connector. Pins A-Z on bottom, 1-22 on top.</b>			
<b>Pin</b>	<b>Signal</b>	<b>Pin</b>	<b>Signal</b>
22	VSS GND	Z	RAM/R/W
21	VCC +5V	Y	$\overline{\text{Ø2}}$
20		X	PLL TEST
19		W	$\overline{\text{R/W}}$
18		V	R/W
17	SST OUT	U	$\text{Ø2}$
16	K6	T	AB15
15	DB0	S	AB14
14	DB1	R	AB13

## KIM-1 Quick Reference

13	DB2	P	AB12
12	DB3	N	AB11
11	DB4	M	AB10
10	DB5	L	AB9
9	DB6	K	AB8
8	DB7	J	AB7
7	RST	H	AB6
6	NMI	F	AB5
5	RO	E	AB4
4	IRQ	D	AB3
3	Ø1	C	AB2
2	RDY	B	AB1
1	SYNC	A	AB0