

## API functions

\$00	System reset	none
\$01	Input character (waits)	none
\$02	Output character (waits)	A = Character to output
\$03	Input status	none
\$04	Input line	A = Size, DE = Address
\$05	Input line default	none
\$06	Output line	DE = Address
\$07	Output new line	none
\$08	Get version details	none
\$09	Claim jump table entry	A = Number, DE = Address
\$0A	Delay in milliseconds	DE = Milliseconds
\$0B	Output message	A = Number
\$0C	Read jump table entry	A = Number
\$0D	Select console in/out device	A = Device number
\$0E	Select console input device	A = Device number
\$0F	Select console output device	A = Device number
\$10	Input char from console device	E = Device number
\$11	Output char to console device	A = Character, E = Device number
\$12	Poll idle events	none
\$13	Configure idle events	A = Mode (0=Off, 1=On)
\$14	Timer 1 control (A x 1ms)	A = Period, DE = Address
\$15	Timer 2 control (A x 10ms)	A = Period, DE = Address
\$16	Timer 3 control (A x 100ms)	A = Period, DE = Address
\$17	Output port initialise	A = Port address
\$18	Write to output port	A = Output byte
\$19	Read from output port	none
\$1A	Test output port bit	A = Bit (0 to 7)
\$1B	Set output port bit	A = Bit (0 to 7)
\$1C	Clear output port bit	A = Bit (0 to 7)
\$1D	Invert output port bit	A = Bit (0 to 7)
\$1E	Input port initialise	A = Port address
\$1F	Read from input port	none
\$20	Test input port bit	A = Bit (0 to 7)
\$21	Set baud rate	A = Device, E = Rate code
\$22	Execute command line	DE = Pointer to line
\$23	Get pointer to command line	none
\$24	Skip delimiter in command line	DE = Pointer to line
\$25	Skip non-delimiter in command	DE = Pointer to line
\$26	Get Hex parameter from command	DE = Pointer
\$27	Get current console I/O devices	none
\$28	Get top of free memory	none
\$29	Set top of free memory	DE = Top of free memory

## On Entry

A = Reset type
none
A = Character to output
none
A = Size, DE = Address
none
DE = Address
none
none
A = Number, DE = Address
DE = Milliseconds
A = Number
A = Number
A = Device number
A = Device number
A = Device number
E = Device number
A = Character, E = Device number
none
A = Mode (0=Off, 1=On)
A = Period, DE = Address
A = Period, DE = Address
A = Period, DE = Address
A = Port address
A = Output byte
none
A = Bit (0 to 7)
A = Bit (0 to 7)
A = Bit (0 to 7)
A = Bit (0 to 7)
A = Port address
none
A = Bit (0 to 7)
A = Device, E = Rate code
DE = Pointer to line
none
DE = Pointer to line
DE = Pointer to line
DE = Pointer to line
DE = Pointer
none
none
DE = Top of free memory

## On Exit

none
A = Character input
A = Character output
NZ if char avail
A = Length, DE = Address
A = Length, DE = Address
none
none
Version info in A B C D E H L
none
none
none
DE = Address
none
none
DE = Address
none
A = Output (zero)
A = Output byte
A = Output byte
A = Output byte
A = Output masked
A = Output byte
A = Output byte
A = Output byte
A = Input byte
A = Input byte
A = Input masked
NZ flagged if OK
Z flagged if OK
DE = Pointer to line
DE = Pointer to line
A = Char, DE = Pointer
A = Char, DE = Pointer
A = Status, DE = Ptr, HL = Value
D = Output, E = Input
DE = Top of free memory
none

## API Monitor Command:

API <function number> [<A>] [<DE>]  
*result displayed:* <A> <DE>

## API Assembler Instructions:

LD C, <function number>  
RST \$30

## Self-test (at reset)

Test output is via LEDs on the system's status display port (eg. RC2014 or LiNC80 Digital I/O module)

Pass	Single sweep of lights followed by all lights off
Failed RAM	Continuous sweeping of lights
Failed serial	LED bit 0 stays on if serial module not found

## Monitor command line interpreter

? / HELP		Display help
A	<address>	Assemble instructions
API	<function> [<A>] [<DE>]	Call API function
B	<address>	Breakpoint set or clear
BAUD	<device> <rate code>	Set baud rate
CONSOLE	<device identifier>	Select console device
D	<address>	Disassemble instructions
DEVICES		List devices detected
DIR		List files in the ROM
E	<address>	Edit memory
F	<flag or condition name>	Flags display or modify
FILL	<start> <end> <byte>	Fill memory
G	<address>	Go to program
I	<port>	Input from port
M	<address>	Memory display
O	<port> <data>	Output to port
R	<register name>	Registers display or edit
RESET		Reset monitor
S	<address>	Step one instruction

Monitor ROM may also include additional commands, such as "BASIC".

## Jump Table

\$00	Non-maskable interrupt handler	
\$01	Restart \$08, console character output	
\$02	Restart \$10, console character input	
\$03	Restart \$18, console input status	
\$04	Restart \$20, handler (not currently used)	
\$05	Restart \$28, breakpoint handler	
\$06	Restart \$30, applications programming interface (API) handler	
\$07	Restart \$38, mode 1 interrupt handler	
\$08	Console input routine	
\$09	Console output routine	
\$0A	Reserved for get console input status	
\$0B	Reserved for get console output status	
\$0C	Idle event handler	
\$0D	Timer 1 event handler	
\$0E	Timer 2 event handler	
\$0F	Timer 3 event handler	
\$10	Device 1 input character	default = serial port channel A
\$11	Device 1 output character	default = serial port channel A
\$12	Device 2 input character	default = serial port channel B
\$13	Device 2 output character	default = serial port channel B
\$14	Device 3 input character	
\$15	Device 3 output character	
\$16	Device 4 input character	
\$17	Device 4 output character	
\$18	Device 5 input character	
\$19	Device 5 output character	
\$1A	Device 6 input character	
\$1B	Device 6 output character	

## Memory map

ROM (minimum) 8 kbytes, 0x0000 to approx 0x1E00 used  
RAM (minimum) 32 kbytes assumed from 0x8000 to 0xFFFF, 0xFC00 to 0xFFFF used

## Flag and condition names

<i>Flags:</i>	<i>set</i>	<i>clear</i>
Zero	Z	NZ
Carry	C	NC
Negative	S	NS
Half carry	H	NH
Parity even	Pa	NP
Subtract	N	NN

## Conditions:

Zero	Z	zero set
Not zero	NZ	zero clear
Carry	C	carry set
Not carry	NC	carry clear
Negative	M	sign set
Positive	P	sign clear
Even	PE	parity set
Odd	PO	parity clear

## Baud rate codes

<i>Baud rate</i>	<i>Rate codes</i>
230,400	\$1 \$23
115,200	\$2 \$11
57,600	\$3 \$57
38,400	\$4 \$38
19,200	\$5 \$19
14,400	\$6 \$14
9,600	\$7 \$96
4,800	\$8 \$48
2,400	\$9 \$24
1,200	\$A \$12
600	\$B \$60
300	\$C \$30

Either code can be used

## Console devices

<i>Console device</i>	<i>Identifiers</i>
#1 (eg. SIO A)	\$1 \$A
#2 (eg. SIO B)	\$2 \$B
#3 (serial port 3)	\$3
#4 (unassigned)	\$4
#5 (unassigned)	\$5
#6 (unassigned)	\$6

Either identifier can be used