

COM3 - PuTTY

** MONK 68000 monitor Ver 5.1 D.A.Rumball 2023 **

==>??

MONK_68K v5.1 Help screen.

HEX numbers are input in free format, end with a space.

SI....Set input port	SO....Set output port
AD....Ascii memory dump	BR....Show breakpoints
LK....load ascii text to ram	DP....display peripheral data
CP....Continue program	CS....Generate memory checksum
DB....Define breakpoints	DC....Display time and date
DR....Display 68000 registers	TS....test drive stepping
TD....random read drive test	BO....boot operating system
FI....Find byte string	FM....Fill memory with constant
DU....Hex+ASCII dump of memory	HE....Display this screen
DF....format floppy disc	WF....format wini disc
JU....Jump to program	MC....Examine and alter clock
ME....Examine and alter memory	PM....Poke memory
RP....Run program (use regs)	SA....Set 68000 address regs
SB....Set serial baud rates	SD....Set 68000 data regs
SM....Shift block of memory	SR....S-record loader
SS....Set 68000 status reg	RS....read disk sector
TM....Test memory	TR....Trace single instruction
WS....write disk sector	

==>

monk.txt - Notepad

File Edit Format View Help

TTL MON_K 68k system services and debug routines

LIST

```

*****
*****
** MONK Ver 5.1 **
** D.A.Rumball Jun 23 **
*****
*
*****
* This program is the core section of a
* general 68000 system monitor. It can
* be configured to a particular system
* by linking to other modules :-
*
*
* OSSUBS.....Contains the o/s
* interface.
*
* MINIT.....Contains the power on
* and reset code.
*
* EXCEPT.....Contains the 68k
* exception code.
*
* CONSOLE.....Contains the console
* driver code.
*
* CHARS.....Font definitions.
*
* GRAPHICS....Contains any system
* code.
*
* DISK.....Contains the disk driver
* code.
*
* SUBS.....Contains the code for any

```

equates.txt - Notepad

File Edit Format View Help

```

*****
*
* Equate file for MONK ver 2.1
*
* -----
* This file contains the hardware equates for
* MONK ver 2.1. It should be 'included' in
* each module of the program.
*****
*
*****
* Base addresses of the major blocks *
*****
prom equ $000C0000 Prom base address.
ram equ $0007fc00 Monk scratch ram base
io equ $000ff000 Input/output base address
*
*****
* Base addresses of buffers and jump tables etc *
*****
s_stack equ $0001fbfe Initial system stack
u_stack equ $0001fbfe Initial user stack.
*
j_table equ $0001fe00 Jump table base address
*
cpmbase equ $00007000 CP/M 68k scratch ram base
*
text_base equ $00007000 text screen base address

```