

Ed Smith's

# SOFTWARE WORKS

330 Camino De Las Colinas, Redondo Beach, California 90277 213/375-3977

S M I T H B U G

Copyright 1978-Ed Smith

The SMITHBUG resident system monitor EPROM can be used alone or in conjunction with your present monitor. It occupies the upper 2K bytes of memory space, starting at \$F800, and requires a 6850 ACIA at location \$8004 (SWTBUG Version) or \$8008 (SMARTBUG Version) and 128 bytes of RAM at location \$A000. The RAM variables locations have been kept compatible with either SMARTBUG or SWTBUG. As SMITHBUG is an EPROM, it can be reprogrammed or patched to suit individual system requirements.

SMITHBUG uses the greater than symbol ">" as its prompt and responds to the following valid commands: A, B, C, D, E, F, G, H, I, J, K, M, N, O, Q, R, S, T, V, X, 1, 2, @, & and \*. Any other character typed after the prompt will be ignored and a new prompt will be issued.

## COMMAND SUMMARY

<u>Command Character</u>	<u>Operation</u>
R ✓	Display pseudoregisters (user's stack)
A ✓	Change A pseudoregister and display pseudoregisters
B ✓	Change B pseudoregister and display pseudoregisters
C ✓	Change condition codes pseudoregister and display pseudoregisters
X ✓	Change X pseudoregister and display pseudoregisters
S ✓	Change pseudoregister stack address and display resulting pseudoregisters
D ✓	Disassemble code in memory
T ✓	Trace (execute) a program one instruction at a time
K	Continue with trace at pseudoregister program counter location
1	Set one breakpoint in program and execute beginning at pseudoregister program counter location
2	Set two breakpoints and execute as above
E ✓	Examine (or examine and change) a memory location
V ✓	View a 128-byte block of memory
M ✓	Move a block of memory
I	Insert a specified value in all locations of a memory block
→ F <i>one byte only</i>	Find a specified byte in a memory block
@	Fill with ASCII characters from keyboard
G	Go to location in PC pseudoregister and execute
J	Jump to address to be specified
N	Inhibit echo to system terminal
O	Allow echo to system terminal
→ H	Send monitor output to hard copy device
→ Q	Jump to disc boot routine
→ &	Jump to disc DOS routine warm entry point
→ *	Jump to adjunct monitor

Note: There are no tape load or save commands. It is assumed your present monitor will provide those.

### Detailed Monitor Command Descriptions

In the examples to follow, the users entries are underlined.

## Pseudo Register Display/Modify Commands (R, A, B, C, X, S)

Typing an R will display the current contents of the A, B, X and condition code pseudoregisters and the current position of the stack pointer and program counter. Contents of the A, B, S, PC and X pseudoregisters will be displayed in hexadecimal. Bits set in the condition code pseudoregister will be indicated by the alphabetic designator assigned to that bit position. Bits that are not set will be indicated by a hyphen (-). For example:

```
R  -IN-V-  B=FF  A=C1  X=5439  PC=E800  S=E7B0
```

The Interrupt, Negative, and Overflow bits are set. The Half carry, Zero and Carry bits are not set.

To change the contents of a pseudoregister, type the appropriate letter (A, B, C, X, or S) followed by a new value in hexadecimal. The updated pseudoregisters will be displayed after any change. Changing the stack pointer S will cause all the pseudoregister values to change.

The PC pseudoregister may be changed by use of the E command to alter the appropriate memory locations in the User's Stack. These locations are always the sixth and seventh locations above the current stack pointer. It may also be changed by using the T command, an address and a carriage return after the instruction display.

## "D" Disassemble Command

```
D  
FROM ADDR LLLL  
LLLL 00 NNN A RRRR ...  
LLLL 00 NNN A RR TTTT ...
```

Typing D followed by a four digit hexadecimal address puts SMITHBUG in the single-step disassembly mode. The program counter (LLLL) and opcode (00) are shown in hexadecimal. The instruction mnemonic and operand is shown in standard Motorola format for everything except indexed instructions which are shown as HH,X where HH is the hexadecimal operand. The target address (TTTT) is shown for all relative branches. In addition, each byte of the instruction is shown as either a period or an ASCII character at the right side of the display. Hit the SPACE BAR to disassemble the next instruction. Hit carriage return to exit the mode. Invalid instructions are shown as "\*\*\*\*".

## "T" Trace Command

```
T  
FROM ADDR LLLL
```

Typing T followed by a four digit hexadecimal address puts SMITHBUG in the single-step trace mode. The first line displayed will be a

pseudoregister R dump; the second line will be a D disassembly display. Hitting the SPACE BAR will execute the displayed instruction and produce a R dump of the resulting CPU status to the right side of the D display. A disassembly of the next instruction to be executed will be produced on the next display line. Continuing use of the SPACE BAR will software "single-step" through the program if it is in RAM. Prior to executing the next instruction, the user may change the A, B, C or X registers with the A, B, C or X commands. The resulting CPU status is shown after each change. Stepping to a ROM location will cause the display of a "?" and the trace function ended. A carriage return will exit the trace mode. Jumps and JSR's to ROM addresses above \$E000 are automatically executed without loss of the trace mode. This allows monitor subroutine calls within a program to be executed as a single trace step.

### "K" Continue Command

Typing K will enter the trace mode at the program counter address shown in the R dump display.

### "1" or "2" Breakpoints

1

BKADDR LLLL

or

2

BKADDR LLLL

BKADDR NNNN

Typing 1 followed by a four digit address, or 2 followed by two addresses, will insert an SWI (software interrupt) instruction at the addresses given and then do an automatic G go to the PC address shown in the R display. The instruction(s) is/are saved and is/are restored when the (or either) instruction address is encountered in normal program execution. When the SWI is executed, it is vectored through the address contained in location SWIPTR. (When SMITHBUG is cold started at \$F800, it puts address \$F91F in this location). This will perform an R register display and await the next SMITHBUG command. Typing K at this point will pick up the Trace display at the breakpoint address.

If you are in the Trace mode and want to bypass the single-stepping displays until a point further down stream is reached, use a carriage return to exit Trace and hit the 1 or 2 key to insert the desired next breakpoint. After the break, type K to pick up Trace.

Note: Prior to typing 1 or 2, use the R display to insure the PC address is one you wish.

## "E" Examine and/or Change Memory Contents

### Example:

E 24FA		Type new value to change and open next location
24FA 23	<u>5C</u>	
24FB 39	<u>Space</u>	Touch space bar to open next location with no change
24FC A7	<u>↑</u>	Type Uparrow to open previous location with no change
24FB 39	<u>Return</u>	Touch Return key (or any non-hex key) to terminate.

Location 24FA will be opened and the value 23 located there will be displayed. The value can then be changed by typing the desired hexadecimal value. SMITHBUG will install the new value and then open and display the next location. To open and view contents of successive locations without changing them, touch the SPACE BAR. To open and display the previous location, type the up arrow (↑). To exit this mode, touch RETURN or any non-hex key.

## "V" View Command

V  
FROM ADDR LLLL

Will display a 128 byte block of memory giving both hexadecimal and ASCII equivalent values. To display the next block, hit SPACE. To view a different address, hit V. To exit, hit any other key.

## "F" Find Command

F  
FROM ADDR SSSS  
THRU ADDR EEEE  
VALUE ZZ

Will display any addresses containing ZZ within the limits SSSS and EEEE.

## "M" Move Command

M  
FROM ADDR SSSS  
THRU ADDR EEEE  
TO ADDR TTTT

Block move data from SSSS through EEEE to TTTT.

## "I" Insert Command

I  
FROM ADDR SSSS  
THRU ADDR EEEE  
VALUE ZZ

Will set memory locations SSSS through EEEE to the value ZZ.

## "@" ASCII Command (Fill memory with text from keyboard)

@  
FROM ADDR SSSSI CAN TYPE TO ↑D

Will store character input from keyboard as a byte starting at address SSSS. The input will continue until a control D (hex 04) is entered. The 04 will be the last byte saved. The backspace key (or control H) can be used to correct typing errors.

## "G" Go to User Program

Prior to using the G command, the user must prepare the user's stack. Verification of the proper values can be easily accomplished with the R command.

R HINZVC B=24 A=98 X=52CF PC=0100 S=A042

G

On receipt of the G command, SMITHBUG will pull the pseudoregister values from the user's stack and place them in the respective MPU registers. Execution will begin at the PC pseudoregister value 0100.

## "J" Jump to User Program

J TO ADDR 0100 user program entry point

After receipt of the J command, SMITHBUG will prompt the user to type the entry address of the target program. When the address is entered, control will be transferred to that point. The pseudoregister values are not used with this command and no register initialization is possible.

## "O", "N" and "H" Command

The monitor variable ECHO is a flag that determines whether the console input routine will echo the typed characters and whether the output routine will output to the terminal or will transfer to an alternate output routine external to SMITHBUG.

If the value in ECHO is zero, all characters typed on the console will be echoed back to the console and all output will be sent to the console. If ECHO contains a value greater than zero, the echo will be

suppressed but output will still be displayed. If ECHO contains a value less than zero, the echo will be permitted but all output will be via a jump to location \$A04A.

#### "O" Set Echo On

The O command clears memory location ECHO. When this location is clear, all characters typed on the system console keyboard will be echoed back to the console display, and all output will be sent to the console. This is the initial condition after a system reset.

#### "N" Negate Echo

The N command sets the contents of ECHO to 4E. In this condition, typed characters will not be echoed and hence will not be displayed on the system console.

#### "H" Hardcopy Device On

The H command sets the contents of location ECHO to B8. In this condition, SMITHTBUG will send all output to the alternate output routine. This routine must be supplied by the user and can be added to the system in additional EPROM or can be executed from RAM. The JMP vector to this routine must be placed at \$A04A, \$A04B and \$A04C before using H.

If you want OUTEEE to output both to the system console as well as to your separate hardcopy device when in the H mode, your print routine should end with a jump to FA06. Otherwise, it should end with a RTS (\$39).

The SMARTBUG version ECHO variable is at \$A00B. SWTBUG is at \$A00C.

#### "Q" Quickstart

This command is for the convenience of those people using the Smoke Signal Broadcasting BFD-68 Disc System. Typing Q does the same thing as typing J 8020. SMITHTBUG transfers control to \$8020 which is the beginning address of the routine that boots in the disc operating system from a cold start.

#### "&" Disc

Typing & (the DOS68 prompt character) transfers control from SMITHTBUG to \$7283 which is the warmstart address of DOS68, the disc operating system used with the BFD-68 disc system. This provides a convenient means of re-entering the DOS68 monitor from SMITHTBUG when DOS68 has previously been booted in from disc and is resident in memory.

#### "\*" Command

Typing \* (the SMARTBUG prompt character) will transfer control to \$E0E3, the SMARTBUG, SWTBUG warm entry point.

The last 3 command characters and jump locations can be customized by changing the COMMAND JUMP TABLE on page 17 of the SMITHBUG listing.

#### IRQ and NMI

If the system encounters an IRQ interrupt request, it will jump to the location contained in memory locations \$A000 and \$A001. An NMI interrupt will cause SMITHBUG to jump to the location contained in memory locations \$A006 and \$A007.

At system turn on, this is set to jump to the SWI vector routine.

#### COMPATIBILITY WITH MIKBUG, SMARTBUG OR SWTBUG

A functionally equivalent SMITHBUG routine can be found for all the normal subroutine calls within the above listed monitors. As SMITHBUG is a stand alone monitor, you can free up the memory addresses used by the above monitors by replacing all calls to them with their equivalent address within SMITHBUG.



## APPENDIX D

## ASCII Hexadecimal to Decimal Conversion Table

CHARACTER	HEXADECIMAL	DECIMAL	CHARACTER	HEXADECIMAL	DECIMAL	CHARACTER	HEXADECIMAL	DECIMAL
NUL	00	000	+	2B	043	V	56	086
SOH	01	001	,	2C	044	W	57	087
STX	02	002	-	2D	045	X	58	088
ETX	03	003	.	2E	046	Y	59	089
EOT	04	004	/	2F	047	Z	5A	090
END	05	005	0	30	048	[	5B	091
ACK	06	006	1	31	049	\	5C	092
BEL	07	007	2	32	050	]	5D	093
BS	08	008	3	33	051	>	5E	094
HT	09	009	4	34	052		5F	095
LF	0A	010	5	35	053	'	60	096
VT	0B	011	6	36	054	a	61	097
FF	0C	012	7	37	055	b	62	098
CR	0D	013	8	38	056	c	63	099
SO	0E	014	9	39	057	d	64	100
SI	0F	015	:	3A	058	e	65	101
DLE	10	016	;	3B	059	f	66	102
DC1	11	017	<	3C	060	g	67	103
DC2	12	018	=	3D	061	h	68	104
DC3	13	019	>	3E	062	i	69	105
DC4	14	020	?	3F	063	j	6A	106
NAK	15	021	@	40	064	k	6B	107
SYN	16	022	A	41	065	l	6C	108
ETB	17	023	B	42	066	m	6D	109
CAN	18	024	C	43	067	n	6E	110
EM	19	025	D	44	068	o	6F	111
SUB	1A	026	E	45	069	p	70	112
ESC	1B	027	F	46	070	q	71	113
FS	1C	028	G	47	071	r	72	114
GS	1D	029	H	48	072	s	73	115
RS	1E	030	I	49	073	t	74	116
US	1F	031	J	4A	074	u	75	117
SP	20	032	K	4B	075	v	76	118
!	21	033	L	4C	076	w	77	119
"	22	034	M	4D	077	x	78	120
#	23	035	N	4E	078	y	79	121
\$	24	036	O	4F	079	z	7A	122
%	25	037	P	50	080	{	7B	123
&	26	038	Q	51	081		7C	124
'	27	039	R	52	082	~	7D	125
(	28	040	S	53	083	DEL	7E	126
)	29	041	T	54	084		7F	127
*	2A	042	U	55	085			