CHAPTER 13

# Theory Of Operation

13.1 Activating Other Debuggers
13.2 Virtual Machine Basics

## 13.1 Activating Other Debuggers

Soft-ICE works with most other debuggers by taking advantage of the 8086 family break point interrupt (INT 3). Most debuggers use the single byte INT 3 (CCH) instruction to produce break points. The target instruction is replaced by an INT 3. When the target address is executed, control is given to the debugger's INT 3 handler. The debugger then replaces the (CCH) with the first byte of the original instruction.

When Soft-ICE break points occur, one of several events can happen, depending on the ACTION command. Typically, when using Soft-ICE with another debugger, ACTION is set to INT3. When break point conditions match, Soft-ICE passes control to the host debugger by simulating an INT 3.

Some debuggers may not work properly by simulating INT 3's. For these debuggers, two other ACTION options are provided. They are INT1 and NMI. IX 1 is the 8086 family single-step interrupt. Most debuggers will handle an unsolicited INT 1 as a break point. NMI is supported by many debuggers as a means of breaking out of a hung condition. These debuggers were designed for hardware break-out switches that produced non-maskable interrupts. When ACTION is set for NMI, Soft-ICE simulates the non-maskable interrupt (Interrupt 2). CODEVIEW works best with ACTION set to NMI.

## 13.2 Virtual Machine Basics

The magic of Soft-ICE is made possible by the virtual machine capability of the 80386 processor. Soft-ICE runs in the 80386 protected mode and manages the DOS environment. The 80386 protection circuitry gives Soft-ICE complete control of the DOS environment while protecting it from a wayward program.

How are Soft-ICE break points generated?

Soft-ICE uses three different 80386 features to produce break points:

- * Break points on memory location use the 80386 break registers
- * Break points on memory ranges use the 80386 paging mechanism

- * Break points on I/O instructions use the I/O privilege level and I/O bit mask

How is the BREAK command implemented?

The BREAK command allows use of the keyboard to bring up Soft-ICE, even when interrupts are disabled and the system is hung. Soft-ICE virtualizes the interrupt mechanism so that interrupts are never disabled to Soft-ICE, even when they are disabled to the DOS program running in the virtual machine.

When in break mode, the following instructions are virtualized to make sure the interrupt flag is never cleared:

PUSHF
POPF
STI
CLI
INT n
IRET

Special considerations with virtual 8086 mode

Soft-ICE runs DOS in an 8086 virtual machine. This capability is a major feature of the 80386 microprocessor. When running real address mode software (DOS, etc.) in a virtual machine some 8086 features must be emulated by a program that controls the virtual machine. In our case,

Soft-ICE controls the virtual machine. The following peculiarities are handled by Soft-ICE:

* ROM BIOS interrupt 15H functions 87H, 88H, and 89H
* The undocumented loadall instruction
* Address line 20H control
* 80286 and 80386 protected instructions
* 80386 bugs

ROM BIOS interrupt 15H functions 87H, 88H, and 89H

BIOS function 87H allows a program to access memory above one megabyte in the IBM AT or Personal Series 11 architectures through a block move mechanism.

Function 88H returns the extended memory size. These functions are used by the VDISK device driver. Soft-ICE emulates these BIOS calls for VDISK compatibility. Function 89H is normally used to put you into protected mode, but Soft-ICE can not allow this to happen. Instead it returns with the carry flag set.

## The undocumented loadall instruction

The 80286 contains an undocumented instruction called loadall. This instruction was originally placed on the chip for diagnostic purposes and is not generally used by software. However, it is used by some versions of Microsoft's RAMDRIVE which is sold with Microsoft Windows and MSDOS 3.2. Soft-ICE emulates loadall to the extent of getting RAMDRIVE to work, however it is impossible to do a complete emulation of this instruction.

## Address line 20H control

The IBM AT introduced a special feature that allowed some old programs that were originally written for CP/M to function on the 80286 processor. This feature allowed memory accesses that wrapped from the one megabyte region to the zero region on the 8086 to work on the 80286. Some programs disable this 'wrap compatibility' to access memory just above one megabyte in real address mode. Soft-ICE emulates this ability. This is supported on all 80386 AT machines through the keyboard controller, and through I/O port 92H on the PS/2.

## 80286 and 80386 protected instructions

Some AT specific programs have used 80286 protected instructions. With the emergence of the 80386, some 80386 programs use 80386 protected instructions. These programs will not work with Soft-ICE.

Soft-ICE supports the standard real-address mode extensions that Intel had included with the 80186 & 80286 processors (PUSHALL, POPALL, etc.), but not protected mode instructions such as LGDT, LMSW, etc.

## 80386 Bugs

There are several 80386 bugs up through the C stepping of the chip. Most of these bugs only apply to protected mode software (such as Soft-ICE).

Page 232 is BLANK