Using SourcePoint® to Solve Common BIOS Development Problems

An Application Note
www.arium.com

Overview

When writing BIOS firmware for PC platforms or porting existing firmware to new PC platforms, there are many common problems that developers face. Before in-target probes (ITPs), overcoming these problems required a POST error debug card to display I/O output to port 0x80 and a large amount of development time spent compiling, flashing, and testing code. With SourcePoint® and an American Arium ITP, finding and fixing code bugs is a much more rapid process. This document lists some common BIOS development problems and describes how SourcePoint can be used to overcome them. This document assumes the reader has a basic understanding of BIOS development, SourcePoint, and American Arium's ITP products.

Locating POST Hangs

Without an ITP, locating a POST (Power-On Self Test) hang required adding I/O port 80 codes to the BIOS to narrow down where the processor was executing before the hang occurred. With SourcePoint, the developer has a couple of options. The user can simply stop the processor when hung in post. If the processor is caught in a loop, the problem usually can be found by setting breakpoints earlier in the code and examining the processor registers.

If the processor is executing in an invalid area of memory, usually caused by an incorrect jump address, finding the location of the error requires more work. The user can set a breakpoint on I/O accesses to port 80, run from reset, and make note of the value written (the value in the AL register). After finding the last POST code written, it is a simple matter to get back to that point in POST and begin stepping through the code using either Step Into (F8) or Step Over (F10) commands. The Step Over command is useful in eliminating large blocks of code from the problem. Setting breakpoints along the way gives the user an easy way to quickly get back to the same area of code without having to repeat the previous procedure.

Testing Code Fixes

When testing a fix, the developer can make changes to code in memory immediately in SourcePoint. This allows the developer to test fixes multiple times before having to compile code and flash the ROM.

Using the 'asm' command in the Command Line window, the developer can assemble instructions in-line. This function requires that the code is located in write-enabled memory space. The following example shows how to patch code starting at linear address 4000:

```
asm 4000L =
P0@00004000L>mov bx,ax
P0@00004002L>mov cx,ax
P0@00004004L>
```

System Management Mode Code Debug

Debugging code that runs in System Management Mode (SMM) has special problems of it own. The memory space in which the code resides can be made inaccessible during normal processor operating mode. SourcePoint allows the developer to specify debug register breakpoints that will stop the processor only when in SMM. SourcePoint also allows the developer to break on entry into SMM and exit out of SMM.

CMOS Corruption

CMOS configuration and corruption problems commonly occur when code incorrectly writes to CMOS or does not update the CMOS checksum. Often this code is hard to find when it is not normally associated with CMOS functions. SourcePoint allows the user to break on access to CMOS by setting breakpoints on the CMOS access I/O ports. This makes locating inappropriate CMOS changes a fairly simple task.

Reset Problems

In BIOS development, it is common to encounter problems with reset occurring and continually rebooting the system. SourcePoint allows the user to break on reset, making it much easier to track down the source of the reset.

Processor Configuration

Without access to processor registers, writing processor initialization and configuration code is a tedious trial-and-error process. SourcePoint provides simple GUI and command line access to all published processor registers, making this an easy task.

Chipset and Memory Configuration

When the BIOS does not correctly configure the core chipset and memory, the code can be particularly difficult to debug without an ITP, because the lack of such configuration usually prevents the system from booting into a usable state. Tracking down configuration issues is made easier with access through SourcePoint to all chipset and memory configuration registers.

PCI Configuration

When debugging problems in PCI configuration code, SourcePoint provides access to the PCI configuration space through I/O ports 0xCF8 and 0xCFC, allowing examination and manipulation of PCI device configuration during code execution.

Summary

BIOS development debugging takes place in an environment that software code debuggers cannot always manage with any ease. SourcePoint is an essential tool in this environment, providing simple solutions to common problems.

14811 Myford Road, Tustin, CA 92780 Voice: 877-508-3970 Fax: 714-731-6344 E-mail: info@arium.com Web www.arium.com