1394 Open HCI BIOS 101

Curtis E. Stevens
Consulting Engineer
Phoenix Technologies

Agenda

- Introduction
- Initial Chip Configuration
- Adapter Cards
- System Requirements
- BIOS Boot (x86)
- Legacy DOS Support

1394 HCI OPEN

Introduction

- **Why BIOS?**
  - **Required for boot**
    - **Required for initial configuration**
  - **Desirable for legacy DOS support**

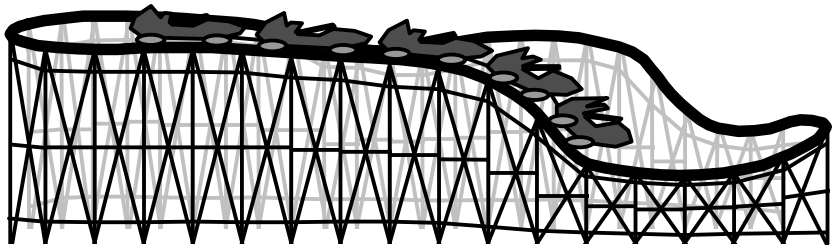
1394 HCI PEN

Initial Chip Configuration
From a Motherboard Perspective

1394 HCI OPEN

GUID/EUI-64

- Global Unique IDentifier (GUID)
  - OpenHCI 1394 Terminology
- Extended Unique Identifier (EUI-64)
  - IEEE EUI-64
- Terms can be used interchangeably
- This presentation uses GUID
  - Its shorter!!!



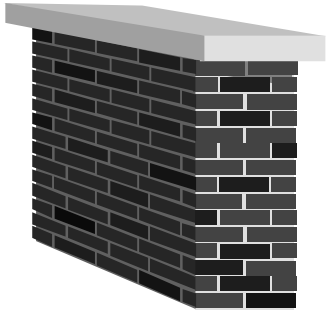
1394 HCI PEN

GUID Requirements

- Each 1394 controller must have a unique GUID
- The GUID must be set before an OS is launched
- The GUID must be stored in an area which the user can't flash
  - BIOS upgrades could erase the GUID
  - BIOS upgrades could propagate non-unique GUID's

1394 HCI PEN

Storage Options



- **Boot Block**
  - Can not be corrupted by the user
  - Not always available
  - Requires a custom Boot Block for each motherboard
- **Alternative ROM storage**
  - 1394 Serial ROM
  - Added cost
  - Better security
  - Can also be used to store other system specific information

1394 HCI PEN

Initialization Sequence

- **BASIC initialization**
  - **Store the GUID**
  - **Initialize the Asynchronous Receive Unit (ARU)**
    - **The ARU allows the host to enumerate the bus**
    - **If the BIOS does not enumerate the bus the ARU does not need to be enabled**
  - **Configure bus specific registers**
  - **Store a pointer to Config ROM**
- **Enable the Link**
  - **All required registers must be initialized before the Link is enabled**

Initialization Sequence After Link Enable

- Requests for config ROM information are processed by the chip without host intervention
  - The host CPU need not have IRQ's enabled
  - The host CPU need not respond to any device
  - The host CPU need not respond to unsolicited requests in the ARU
    - The host need not enable the ARU

1394 HCI PEN

Adapter Cards

1394 HCI OPEN

Requirements

- Very similar to motherboard requirements
- Must use serial ROM initialization
  - The serialROM bit in the version register is 1!
  - Ensures that correct GUID is always stored
  - Prevents tampering
- Serial ROM *should* not be removable
  - User pops the Serial ROM and BIOS off the card
  - Uses device driver which can load anything

1394 HCI OPEN

Bootability

- Requires an option ROM
  - May be socketed
  - Does NOT load the GUID, this is a function of the serial ROM
  - Option ROM is not required if bootability is not important
- For x86 this option ROM provides INT 13 services

1394 HCI PEN

PCI System Requirements

1394 HCI OPEN

PCI System Resource Summary

- 1 IRQ
- 1k PCI register space
- 1K 1394 memory mapped register space not required
- Config ROM from 24 bytes min to 1k
- .5k ARU (minimum suggested RAM)

1394 HCI OPEN

Config ROM

- **Initial Config ROM can be a simple copy of an internal ROM data structure**
- **Motherboard BIOS**
  - **Will normally reside in BIOS shadow area**
    - **Possibly copied from flash**
    - **BIOS Shadow is respected by most memory managers**
  - **May also reside in UMB space**
- **Adapter BIOS**
  - **Config ROM will be in the Option ROM or corresponding shadow region**

1394 HCI PEN

Config ROM (Cont)

- Minimum size is 24 bytes (6 quadlets), includes:
  - Header
  - Bus Info Block (resides in host controller)
    - A note for device manufacturers: If you want to boot your device be sure to set the “bootable” bit in the Bus Info Block
  - Empty Root Directory
- Can grow larger by expanding the root directory

1394 HCI PEN

Config ROM Pitfalls

- eXtended BIOS Data Area (XBDA)
  - If the system is booting from 1394
    - Config ROM must not be located here
    - Memory Managers such as EMM386, QEMM, and 386MAX by default relocate XBDA to UMB, or the bottom of DOS
    - 1394 Register pointers get lost
  - Once the OS is loaded device drivers can provide a new config ROM

1394 HCI OPEN

Asynchronous Receive Unit

- The following is only 1 design possibility
- Use 2 256 byte buffers to form a circular que
  - This imposes a 512 byte requirement which the BIOS must place in system memory
  - The only place a motherboard BIOS can reliably allocate memory is the eXtended BIOS Data Area (XBDA)
  - Adapter cards can use “private” memory for this capability

1394 HCI OPEN Asynchronous Receive Unit (Runtime)

- **During boot the BIOS must …**
  - **Check the pointer to the ARU for validity**
    - **At some point a memory manager may move XBDA**
    - **The Physical XBDA start address must be calculated if the system is in v86**
  - **If the ARU pointer is invalid a valid pointer must be provided**

1394 HCI

PEN

BIOS Boot Specification (x86)

intel®

COMPAQ

Phoenix®

1394 HCI PEN

Overview

- Provides a method for ordering boot devices
- Provides a method for ordering adapter ROMs which hook INT 13
- Provides support for legacy devices

1394 HCI OPEN

Boot Devices

- **Builds on the PnP specification**
  - **Requires a $PnP header**
  - **Device may be Boot Entry Vector (BEV)**
  - **Or, device may be BIOS Aware IPL Device (BAID)**
- **Provides specifications for adapter ROM vendors**
  - **Requires a $PnP header in PCI adapter ROMs**
  - **Provides formatting requirements for the product ID string**

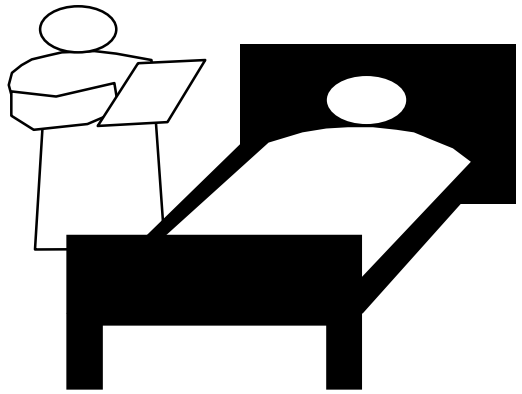
1394 HCI OPEN

INT 13 Hookers

- Provides a method for ordering adapter ROMs which hook INT 13
  - Requires $PnP header
  - Device must be Boot Connection Vector (BCV)
- Defines how $PnP adapter ROM headers apply to booting
- Allows for BIOS level product differentiation

1394 HCI OPEN

Legacy Cards



- Allows legacy devices to be installed in any order
- PCI cards without the $PnP header are treated as legacy devices
- Provides a runtime interface for managing boot devices
  - Extends the $PnP interface by using PnP function numbers
    - Numbers 60-6F are now reserved for BBS
  - 32 Bit protect mode capable

1394 HCI OPEN

In Short
BBS Provides the Following:

- A structured way for adapter ROMs to gain access to system resources such as INT 13
- A structured way for the BIOS to enumerate boot devices before an OS is launched
- A structured way for the system to re-boot with a different boot device under program control

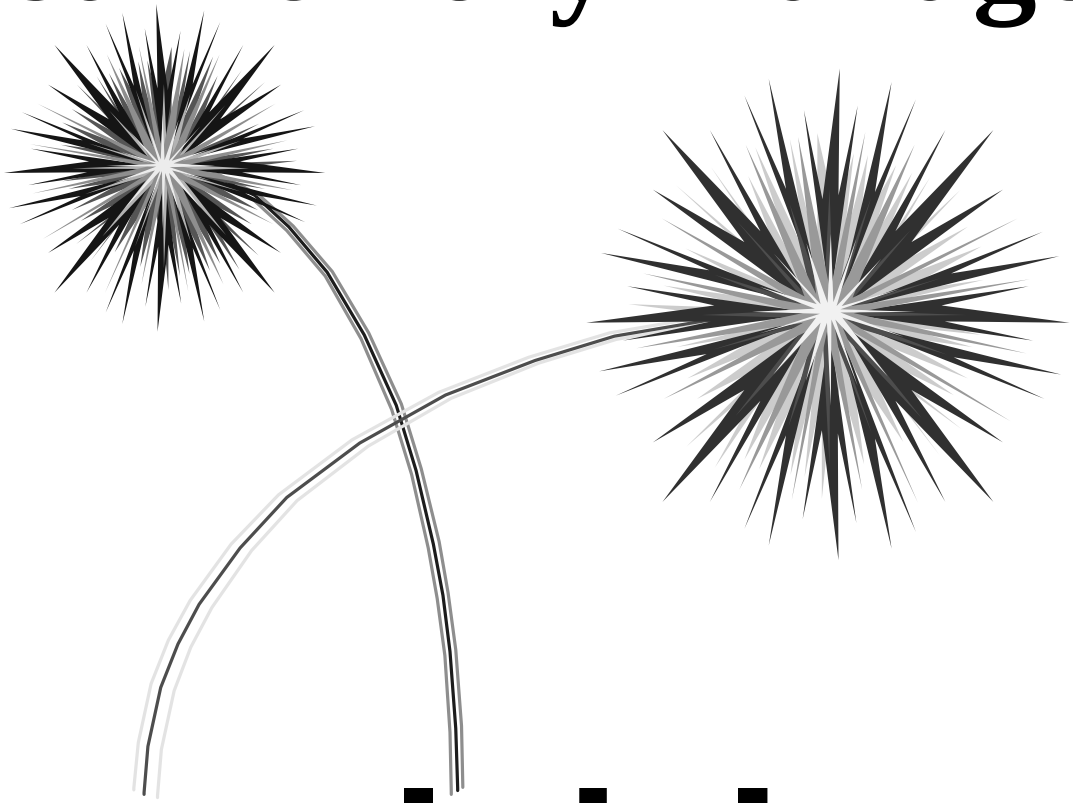
1394 HCI OPEN

For More Information...

- **BIOS Boot Specification v1.01**
  - **Can be downloaded from WWW.PHOENIX.COM/TECHS/SPECS.HTML**
  - **Contact Scott Townsend at Scott_Townsend@PTLTD.COM**

1394 HCI PEN

Post Memory Manager



Phoenix®

intel®

1394 HCI OPEN

Overview

- Simple, verifiable interface
- Provides services for allocating memory during POST
  - Allocate
  - Deallocate
  - Find
- All services are available in 32-bit protect mode
- Does not provide run-time memory

1394 HCI OPEN

Interface

- Verified via a $PMM header and checksum
- Designed to be “C” friendly
  - Push parameters on the stack
  - Return’s a 32 bit result in DX:AX
- Only valid during POST

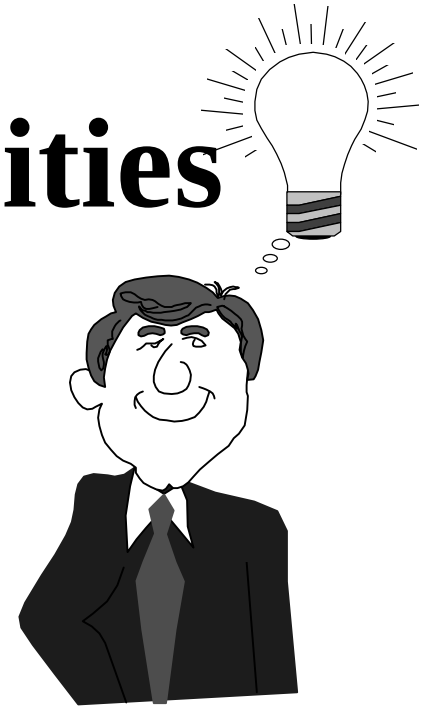
1394 HCI OPEN

Sample PMM Header

Offset	Name	Size	Value	Description

1394 HCI OPEN

Capabilities



- **Allocate/Deallocate**
  - **Conventional memory**
  - **UMB memory**
  - **Extended memory**
  - **Caller can provide a “handle”**
  - **Please oh please deallocate memory when you are done**
- **Find function**
  - **Locates the physical address associated with a “handle”**

1394 HCI OPEN

In the PMM Spec. you will find...

- How to locate and verify $PMM services with sample code
- Sample calls to Allocate and Deallocate
- Usage models for the PMM functions

1394 HCI OPEN

For More Information...

- **Post Memory Manager v1.0**
  - **Can be downloaded from WWW.PHOENIX.COM/TECHS/SPECS.HTML**
  - **Contact Scott Townsend at Scott_Townsend@PTLTD.COM**

1394 HCI PEN

Legacy DOS Support x86

1394 HCI OPEN

INT 13 Support

- Legacy DOS support is used by PC's to boot OS's
- Legacy INT 13 is required for DOS 6.22 and below
  - All addressing is CHS based
- DOS '95, Win NT and Win '95 can use INT 13 Extensions
  - Extensions are LBA based
  - See EDD BIOS Specification, a PC '97 requirement for a description of these extensions
- A CHS geometry must be derived for DOS 6.22 and below

1394 HCI OPEN

INT 13 Support (Cont)

- INT 13 is single threaded
  - The BIOS does not respond to random requests
  - The BIOS responds to a boot device only after that device has been enumerated and a command has been issued
  - In effect, the BIOS acts as the root
- Hot Swapping is not supported
  - Drives are enumerated by the OS at boot
  - Device driver is needed for hot plugging

1394 HCI OPEN

Asynchronous Receive Unit

- 1394 ARU can be supported
  - Use the ARU to enumerate the bus
  - Motherboard BIOS will place this in eXtended BIOS Data Area (XBDA)
  - Option ROM BIOS may place this information elsewhere
  - Minimum .5k is required
  - Unexpected messages will be dumped from the ARU
    - The BIOS only responds to devices it enumerates

1394 HCI OPEN

DMA Notes

- DOS provides virtual addresses
- BIOS is normally OS independent
- BIOS Data Area (BDA) has a flag which indicates when virtual memory services are available
  - OS/2, Win NT, Win ‘95, WFW, Himem.SYS, EMM386, QEMM, 386MAX and others
  - Some OS’s do not support this bit
    - These OS’s will not be 1394 bootable on x86 if they happen to set this bit when they provide a physical memory address

1394 HCI OPEN

DMA (Cont)

- If the INT 13 services detect v86
  - User buffer is converted to a page table
    - This table is stored in XBDA
  - The page table is used in all media access commands to the 1394 device
- If INT 13 services do not detect v86
  - The user buffer address is used directly
  - XBDA is not required

1394 HCI PEN

Any Questions?