Release Notes

HBA 1100 Software/Firmware

Released
August 2019
## Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Revision Date</th>
<th>Details of Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>August 2019</td>
<td>Update for SR 2.4.8 Release</td>
</tr>
<tr>
<td>13</td>
<td>March 2019</td>
<td>Update for SR 2.4.4 Release</td>
</tr>
<tr>
<td>12</td>
<td>January 2019</td>
<td>SR2.4 Production Release</td>
</tr>
<tr>
<td>11</td>
<td>October 2018</td>
<td>SR2.3 firmware update with Cavium/ARM support and Ubuntu driver.</td>
</tr>
<tr>
<td>10</td>
<td>June 2018</td>
<td>SR2.3 Production Release</td>
</tr>
<tr>
<td>9</td>
<td>October 2017</td>
<td>Update Supported OSs</td>
</tr>
<tr>
<td>8</td>
<td>October 2017</td>
<td>First Production Release</td>
</tr>
<tr>
<td>1-7</td>
<td>December 2017 - July 2017</td>
<td>Pre-Production Releases</td>
</tr>
</tbody>
</table>
Contents

1 About This Release ............................................................................................................................ 4
  1.1 Release Identification .................................................................................................................. 4
  1.2 Components and Documents Included in this Release ................................................................. 5
  1.3 Files Included in this Release ....................................................................................................... 6

2 What is New? ................................................................................................................................ 9
  2.1 Features ...................................................................................................................................... 9
  2.2 Fixes ......................................................................................................................................... 9
    2.2.1 Firmware Fixes ..................................................................................................................... 9
    2.2.2 UEFI Fixes .......................................................................................................................... 12
    2.2.3 Driver Fixes ......................................................................................................................... 13
    2.2.4 Management Software Fixes ............................................................................................ 16
  2.3 Limitations .................................................................................................................................. 17
    2.3.1 Firmware Limitations .......................................................................................................... 17
    2.3.2 UEFI Limitations ................................................................................................................ 17
    2.3.3 Driver Limitations ................................................................................................................ 17
    2.3.4 Hardware Limitations ......................................................................................................... 18
    2.3.5 Management Software Limitations .................................................................................... 18

3 Updating the Board Firmware for PQI Operation ....................................................................... 19
  3.1 Updating Controllers to latest (PQI) Firmware ........................................................................... 19

4 Installing the Drivers ..................................................................................................................... 20
1 About This Release

The development release described in this document includes firmware, OS drivers, tools, and host management software for the HBA 1100 Series adapter solutions from Microsemi.

1.1 Release Identification

The firmware, software, and driver versions for this release are shown in Table 1 • Release Summary on page 4.

Table 1 • Release Summary

<table>
<thead>
<tr>
<th>Solutions Release</th>
<th>2.4.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package Release Date</td>
<td>August 13, 2019</td>
</tr>
<tr>
<td>Firmware Version</td>
<td>2.30 B0^1^2 (basecode 06.048.003.000)</td>
</tr>
<tr>
<td>UEFI Version</td>
<td>1.3.8.1</td>
</tr>
<tr>
<td>Legacy BIOS</td>
<td>1.3.8.2</td>
</tr>
<tr>
<td>Driver Versions</td>
<td>Windows SmartPQI:</td>
</tr>
<tr>
<td></td>
<td>• Windows 2012/2016/2019: 106.100.0.1014</td>
</tr>
<tr>
<td></td>
<td>• Windows 7/2008: 6.100.0.1014</td>
</tr>
<tr>
<td></td>
<td>Linux SmartPQI:</td>
</tr>
<tr>
<td></td>
<td>• RHEL 6/RHEL 7/RHEL 8/SLES 12/SLES 15: 1.2.8-026</td>
</tr>
<tr>
<td></td>
<td>• Ubuntu 16/18: 1.2.8-026</td>
</tr>
<tr>
<td></td>
<td>• CentOS 6/7/8: 1.2.8-026</td>
</tr>
<tr>
<td></td>
<td>• Debian 8/9: 1.2.8-026</td>
</tr>
<tr>
<td></td>
<td>VMware SmartPQI:</td>
</tr>
<tr>
<td></td>
<td>• VMWare ESXi 6.0/6.5/6.7: 1.0.3-2323</td>
</tr>
<tr>
<td></td>
<td>FreeBSD/Solaris SmartPQI:</td>
</tr>
<tr>
<td></td>
<td>• FreeBSD 11/12: 1.0.3-2323</td>
</tr>
<tr>
<td></td>
<td>• Solaris 11: 1.0.3-2323</td>
</tr>
<tr>
<td>arconf/Maxview</td>
<td>B23600</td>
</tr>
</tbody>
</table>

Note:

1. Downgrading to 1.04 B0 or older builds from this release or prior 1.29 releases may cause the board to not boot or have supercap errors due to an incompatibility in SEEPROMs between this release and prior releases. Refer to the section "Updating the Controller Firmware" to downgrade an existing board.

2. If the firmware running on the board is older than 0.01 B594, existing data in the logical volumes must be backed up if it needs to be used after the upgrade. After the upgrade from firmware prior to 0.01 B594, the logical volumes will need to be recreated.

3. Only run the driver on firmware 0.01 build 500 or later.
1.2 Components and Documents Included in this Release

Download the firmware, drivers, host management software, and supporting documentation for your HBA1100 controller solution from the Microsemi Web site at https://storage.microsemi.com/en-us/support/start/
1.3 Files Included in this Release

This release consists of the files listed in the following tables:

Firmware Files

Table 2 • Firmware Files

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
<th>Pre-Assembly Use</th>
<th>Post-Assembly Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>SmartFWx100.bin</td>
<td>Programmable NOR Flash File Use to program NOR Flash for boards that are already running firmware.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3 • Firmware Programming Tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
<th>Executable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arconfer</td>
<td>The command allows to upgrade/downgrade the firmware and BIOS image to the controller.</td>
<td>Refer to Table 7 • Host Management Utilities on page 8</td>
</tr>
<tr>
<td>romupdate</td>
<td></td>
<td></td>
</tr>
<tr>
<td>maxView</td>
<td>The firmware upgrade wizard allows to upgrade/downgrade the firmware and BIOS image to one or more controller(s) of same model in the system.</td>
<td>Refer to Table 7 • Host Management Utilities on page 8</td>
</tr>
<tr>
<td>firmware upgrade wizard</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Driver Files

Table 4 • Windows Storport Miniport SmartPQI Drivers

<table>
<thead>
<tr>
<th>Package</th>
<th>Drivers</th>
<th>Binary</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012</td>
<td>Server 2019</td>
<td>SmartPqi.sys</td>
<td>x64</td>
</tr>
<tr>
<td></td>
<td>Server 2016 and Windows 10</td>
<td>SmartPqi.inf</td>
<td>x64</td>
</tr>
<tr>
<td></td>
<td>Server 2012, R2 and Windows 8.1, 8</td>
<td>Smartpqi.cat</td>
<td></td>
</tr>
<tr>
<td>2008</td>
<td>Server 2008 R2 SP1 and Windows 7</td>
<td>SmartPqi.sys</td>
<td>x64</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SmartPqi.inf</td>
<td>x64</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SmartPqi.cat</td>
<td>x64</td>
</tr>
</tbody>
</table>

Table 5 • Linux SmartPQI Drivers

<table>
<thead>
<tr>
<th>Drivers</th>
<th>Intel/AMD x64</th>
<th>Cavium ThunderX2 ARM x64³</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red Hat Enterprise Linux/CentOS 8.0</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux/CentOS 7.6, 7.5, 7.4³</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Red Hat Enterprise Linux/CentOS 7.3, 7.2, 7.1</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux/CentOS 6.10, 6.9³, 6.7, 6.6</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>
### Drivers

<table>
<thead>
<tr>
<th>Drivers</th>
<th>Intel/AMD x64</th>
<th>Cavium ThunderX2 ARM x64</th>
</tr>
</thead>
<tbody>
<tr>
<td>SuSE Linux Enterprise Server 12^1, SP4, SP3^2</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>SuSE Linux Enterprise Server 12^1, SP2, SP1, and Base</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>SuSE Linux Enterprise Server 15 SP1^3</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Oracle Linux 7.6 UEK5u2, 7.5 UEK4</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Oracle Linux 7.2 with UEK 3.10.0-327.el7</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Oracle Linux 7.3 with UEK 4.1.12-61.118</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Oracle Linux 7.4 with UEK4 (4.1.12-94)</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Oracle Linux 7.5 with UEK4 (4.1.12-112)</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Ubuntu 18.04.2, 18.04.1</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Ubuntu 16.04.5, 16.04.4</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Debian 9.8</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Debian 8.11</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Citrix xenServer 8.0,7.6</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

### Note:

1. To mitigate against the Spectre Variant 2 vulnerability, the RHEL 6.9, RHEL 7.4, RHEL 7.5 and SLES12 SP3 and higher drivers have been compiled to avoid the usage of indirect jumps. This method is known as "Retpoline".

2. SLES 12 SP3 cannot be installed on drives attached to the HBA 1100 controller in Cavium Thunder X2 systems due to a CPU and OS installation issue. This driver will support the HBA 1100 controller in Cavium Thunder X2 systems for non-boot drives. For Cavium Thunder X2 servers, if you choose to install SLES12 SP3, you have to install it on the system board’s SATA controller.

3. CentOS 7.4 requires the kernel to be updated to 4.11.0-44 or later.

### Table 6 • FreeBSD, Solaris, and VMware SmartPQI Drivers

<table>
<thead>
<tr>
<th>Drivers</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>FreeBSD 12.0, 11.2</td>
<td>x64</td>
</tr>
<tr>
<td>Solaris 11.3, 11.4</td>
<td>x64</td>
</tr>
<tr>
<td>VMware 6.0, 6.5, 6.7</td>
<td>x64</td>
</tr>
</tbody>
</table>
### Host Management Software

#### Table 7 • Host Management Utilities

<table>
<thead>
<tr>
<th>Description</th>
<th>OS</th>
<th>Executable</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARCCONF Command Line Utility</td>
<td>Windows x64</td>
<td>See the Arcconf download package for the OS-applicable installation executable.</td>
</tr>
<tr>
<td></td>
<td>Linux x64</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VMware EXSi 5.5/6.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XenServer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FreeBSD x64</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solaris x86</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Linux ARM</td>
<td></td>
</tr>
<tr>
<td>ARCCONF for UEFI</td>
<td>Windows x64</td>
<td>Included as part of the firmware downloadable image.</td>
</tr>
<tr>
<td></td>
<td>Linux x64</td>
<td></td>
</tr>
<tr>
<td>maxView Storage Manager</td>
<td>Windows x64</td>
<td>See the maxView Storage Manager download package for the OS-applicable installation executable.</td>
</tr>
<tr>
<td></td>
<td>Linux x64</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VMware EXSi 5.5/6.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XenServer</td>
<td></td>
</tr>
<tr>
<td>maxView vSphere Plugin</td>
<td>vCenter 5.5 and 6.0</td>
<td>See the VMware maxView Storage Manager download package for the OS-applicable installation executable.</td>
</tr>
<tr>
<td>Boot USB (offline or pre-boot) for ARCCONF and maxView Storage Manager</td>
<td>Linux x64</td>
<td>See the maxView BootUSB download package for the .iso file.</td>
</tr>
</tbody>
</table>
What is New?

2.1 Features

Table 8 • Feature Summary on page 9 lists features supported for this release. Features new to this release are designated as "New".

Table 8 • Feature Summary

<table>
<thead>
<tr>
<th>Feature</th>
<th>Supported in this Release</th>
<th>Future Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>UEFI Driver, Boot Support</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Legacy Boot Support</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Dynamic Power Management</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>SMR Drive Support</td>
<td>Enumeration, Unrestricted Command Flow-Through</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>SATL Translation for HA/HM SMR Management</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Identify All Drive Types</td>
<td>X</td>
</tr>
<tr>
<td>Driver Support</td>
<td>Linux</td>
<td>X</td>
</tr>
<tr>
<td>Flash Support</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>MCTP BMC Management</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2.2 Fixes

2.2.1 Firmware Fixes

2.2.1.1 Fixes and Enhancements for Firmware Release 2.30 B0

This release includes the following fixes and enhancements:

- Fixed a 0x1ABD lockup due to a dead lock situation while processing request completions.
  - **Root Cause**: The completion task was not notified about request completions since the corresponding interrupts were not getting enabled to read the completion queue. This resulted in firmware not noticing the completions even after recovery mechanisms, which resulted in a lockup.
  - **Fix**: Re-enable the non-empty callback interrupts after processing completions to read the completion queue.
  - **Risk**: Low

- Fixed an issue where an inquiry command completion is significantly delayed in a controller queue.
  - **Root Cause**: When the controller is run in low power mode, lower layer firmware threads are reduced to achieve power savings. This causes a single thread to handle all IOs and internal operations like rebuilding volumes etc. Queues are serviced based on priority scheduling and this
caused the lower priority queues to starve for longer durations. One such low priority request was to DMA error data of an Inquiry command to the host.

**Fix:** Firmware modified to round robin service all queues in low power mode.

- Fixed an issue where the firmware attempts communication with a non-SES virtual device using SES protocol.
  - **Root Cause:** Lower layer firmware reported all virtual phys to upper layer firmware. Upper layer firmware treats all virtual PHY connections as SES connections. An error occurs when upper layer firmware tries to talk to a virtual device that cannot talk SES.
  - **Fix:** Lower layer firmware modified to hide virtual devices that are not SES targets from the upper layer firmware.
  - **Risk:** Low

- Fixed an issue where the Western Digital drives would not accept any IOs after running the Sanitize Crypto Scramble command.
  - **Root Cause:** This problem occurred only when using Western Digital drives. After the Sanitize Crypto Scramble had completed, the drive would not accept any IOs until about 9 minutes later. The hdparm–sanitize-status commands report the Sanitize Crypto Scramble operation is complete; however, sending TEST UNIT READY commands to the drive shows that the firmware was returning LOGICAL UNIT NOT READY, SANITIZE IN PROGRESS. The firmware was out of sync with the state of the drive.
  - **Fix:** If firmware has set the "Sanitize in progress" flag for the target device, send Sanitize Status Ext in the background for any commands sent by host.
  - **Risk:** Low

- Fixed an issue where the captured controller log is empty.
  - **Root Cause:** Any unimportant event (with class, subclass and detail as zero) logged to event buffer, is also written to persistent memory (MRAM/NVSRAM) during a controller reset. After the controller is reset, the same event is copied back to the event buffer. Host management software will issue commands to read the logs/events until they get the class, subclass and detail all as zero in a response. If the very first log/event returned to the software tools contains all zeros, no further logs are read from the controller.
  - **Fix:** Do not copy back unimportant events from persistent memory to event buffer after a controller reset.
  - **Risk:** Low

- Fixed 0x27006 lockup while handling I/O time outs.
  - **Root Cause:** LUN reset TMF sent during error handling never gets completed after device handle swap logic for SAS devices.
  - **Fix:** Complete all TMFs pending for a device after the device handle swap logic.
  - **Risk:** Low

- Fixed an issue where MCTP EID is set to zero for SMBUS support.
  - **Root Cause:** Firmware checks the EID only for PCIe VDM transport and on servers which supports only SMBUS, this is set always to zero.
  - **Fix:** Get the SMBUS EID if PCIe VDM EID is zero.
  - **Risk:** Low

- Fixed a lockup issue with a portion of firmware, which does error logging using an invalid address.
  - **Root Cause:** When ECC error is encountered, controller firmware attempts to print a few DDR registers as part of the crash dump for further analysis. Firmware was not reading the correct registers for DDR3 and DDR4 configurations.
  - **Fix:** Modified firmware to check the DDR type before printing debug info and dump the appropriate registers.
  - **Risk:**
• Fixed an issue where controller firmware does not robustly handle a drive behavior in which the drive violates SAS specification.
  ◦ **Root Cause**: A SSD drive reported an NCQ error and completion in the same SDB FIS. That behavior is a SAS protocol violation which triggers two messages to lower level firmware. The first message was a completion with an underrun and firmware acted upon this and returned appropriate completion to the host. The second message for the error, reports that the SAS protocol hardware had set a bit called the Drain bit to block all I/Os. In this case, since firmware had already completed the request, it cannot find any context for the associated request for which this error was generated and drops the message. That action causes the Drain bit in the SAS protocol hardware to be left set and all the subsequent I/Os are flushed back (drained) by the SAS protocol hardware.
  ◦ **Fix**: Modify the lower layer controller firmware to clear the drain bit when a SATA error is generated and if it is not entering NCQ error mode. The Drain bit is needed only for NCQ error mode.
  ◦ **Risk**:

• Fixed a controller lockup during the suspend/resume operation.
  ◦ **Root Cause**: During the suspend/resume operation, the scan devices process requires a large amount of memory space and under rare conditions when memory is not available, the suspend/resume operation is interrupted leading to a controller lockup.
  ◦ **Fix**: During suspend/resume operation, if the memory allocation fails than use other internal memory pools.
  ◦ **Risk**: Low

• Fixed an issue where the expander firmware update fails in dual domain configuration.
  ◦ **Root Cause**: During expander firmware upgrade on dual domain configuration, drives behind the expander gets hot removed or hot added back but the internal firmware drive state does not revert to optimal state, leading to controller hang and expander firmware update failure.
  ◦ **Fix**: The firmware will always add the device path even when the mode is set to disable hot plug events. This change will put a device from single path to multi path. This will in turn avoid queuing requests to the device in the internal queue and make it conducive to LUN resets from the host.
  ◦ **Risk**: Low

• Fixed a controller lockup when waking up suspended threads during a memory pool freeing context.
  ◦ **Root Cause**: During a memory pool free operation, interrupts are not enabled when waking up the suspended task, causing a controller lockup.
  ◦ **Fix**: Set the correct interrupt bit before waking up task from a memory pool free operation.
  ◦ **Risk**: Low

• Fixed an issue where the SES status page returned to host gets delay up to 60 seconds.
  ◦ **Root Cause**: When host requests SES status pages via Receive diagnostic command, firmware used a 60 second on demand polling mechanism to get the data and hence the delay.
  ◦ **Fix**: An internal non blocking logical request callback mechanism will be used for SES Receive Diagnostic command to complete the request on time.
  ◦ **Risk**: Low

• Fixed a hardware bug. When DRAM hits this bug, the result is DRAM-wide uncorrectable and correctable ECC errors leading to the controller locking up.
  ◦ **Root Cause**: There is a hardware bug in the self-refresh logic in the Winbond DRAMs that are used in the 16 and 24 port SmartHBA 2100 and HBA 1100 adapters.
  ◦ **Fix**: Disable Low-power mode which effectively disables auto-self refresh.
  ◦ **Risk**: Low

• Modified the vendor identification as “Adaptec” instead of “MSCC” for HBA, SmartHBA, and SmartRAID controllers.
  ◦ **Root Cause**: A request was made to have consistent sysfs information reported for the vendor text string.
2.2.2 UEFI Fixes

Note: Microsoft signed and secure boot is supported.

2.2.2.1 Fixes and Enhancements for UEFI Build 1.3.8.2/Legacy BIOS Build 1.3.8.1

This release includes the following Legacy BIOS fixes and enhancements:

- Added support for blank family ID.
  - Root Cause: When the family ID was blank, there were alignment issues in displaying the controller name during the POST.
  - Fix: Corrected the alignment when the family ID is blank.
  - Risk: Low

This release includes the following UEFI fixes and enhancements:

- Added support for drive firmware update using the ATA microcode download method.
  - Fix: Option added to update drive firmware using the ATA microcode download method.
  - Risk: Low

- Fixed an issue where PCI attribute is not set to support addressing system memory above 4 GiB.
  - Root Cause: UEFI driver does not enable the EFI_PCI_IO_ATTRIBUTE_DUAL_ADDRESS_CYCLE PCI attribute in EFI_DRIVER_BINDING_PROTOCOL.Start().
  - Fix: Enabled the EFI_PCI_IO_ATTRIBUTE_DUAL_ADDRESS_CYCLE PCI attribute in EFI_DRIVER_BINDING_PROTOCOL.Start().
  - Risk: Low

- Fixed an issue where PollMem function of PCI I/O protocol is taking longer time than expected.
  - Root Cause: PollMem implementation of the platform is taking a longer time than the expected timeout.
  - Fix: Replaced PollMem with event timer functions for calculating command timeout.
  - Risk: Low

- Fixed an issue where the Controller Family ID is displayed as blank in the consolidated display for controller name.
  - Root Cause: White space gets added to the controller name if the family ID is blank.
  - Fix: Ignore family ID if blank while creation of consolidated controller name.
  - Risk: Low

- Fixed an issue where the tape autoloader device reported twice in the disk utilities menu.
  - Root Cause: Device enumeration function consider tape device under disk enumeration as well as non disk enumeration.
  - Fix: Ignore other disk types while enumerating physical drives.
  - Risk: Medium

- Fixed an issue where platform boot hang is observed while booting with a multi lun device connected.
  - Root Cause: Invalid memory access due to mismatch in actual memory allocated and requested size for a SCSI command.
  - Fix: Corrected SCSI request size to match allocated size.
  - Risk: Medium
2.2.3 Driver Fixes

2.2.3.1 Fixes and Enhancements for Linux Driver Build 1.2.8-026

The fixes and enhancements in this release.

- Fixed an issue during the force reboot (reboot -f), with outstanding commands that causes a controller firmware lockup.
  - **Root Cause**: Driver does not drain all commands before issuing a PQI reset.
  - **Fix**:
    - During system shutdown, the driver waits until all the outstanding I/O completes before issuing a PQI reset.
    - During kdump, the driver performs a soft reset of the controller before proceeding with normal driver initialization.
  - **Risk**: Medium

- Fixed an issue where there is inconsistent systool/sysfs output between SmartPQI and AACRAID drivers.
  - **Root Cause**: Request for consistent sysfs information to be exposed by the SmartPQI and AACRAID drivers.
  - **Fix**: Modified/added sysfs entries in such a way that they are consistent. In SmartPQI vendor, model, and serial number is added. The firmware version has been made a separate sysfs attribute.
  - **Risk**: Low

- Added a module parameter "expose_ld_first" to cause the driver to expose logical drives before physical drives to the OS.
- Added bay and enclosure identifier fields by populating the `pqi_sas_get_enclosure_identifier` and `pqi_sas_get_bay_identifier` routines so that they can be digested by sysfs output.
- Added a module parameter “hide_vsep” to hide the controller VSEP.
- Fixed an issue where CCISS_REGNEWD returns good completion, but device files for physical devices were not created.
  - **Root Cause**: When device scan is going on, if CCISS_REGNEWD is received from the application, driver queues the rescan worker and returns SUCCESS.
  - **Fix**: When the rescan worker is queued, return -EINPROGRESS instead of returning SUCCESS, if a rescan worker is in progress.
  - **Risk**: Low

- Fixed an issue where driver writes corrupted timestamp to controller log in certain Linux variants.
  - **Root Cause**: When driver is loaded, `pqi_write_current_time_to_host_wellness` is executed. Timestamp is corrupted for the kernel that does not support the `ktime_get_real_seconds` API.
  - **Fix**: Read 32-bit variable through timeval structure for the kernels that do not support the `ktime_get_real_seconds` API.
  - **Risk**: Low

- Fixed an issue where during the force reboot (reboot -f), there are outstanding commands while processing PQI reset, that causes firmware controller lockup.
  - **Root Cause**: Driver does not block driver initiated RAID path requests before issuing PQI reset.
  - **Fix**: During system shutdown, driver will disable the events and block synchronous commands before issuing a PQI reset.
  - **Risk**: Low
2.2.3.2 Fixes and Enhancements for FreeBSD Driver Build 1.0.3-2323

Following are the fixes and enhancements in this release.

- Added FreeBSD 12.0 support for the SmartPQI driver.
- Fixed an issue when allocating memory using malloc, the tag name and description does not match with the driver name.
  - Root Cause: While allocating memory using malloc, the tag name and description did not match with the driver name.
  - Fix: Replaced the tags SMARTRAID with SMARTPQI when allocating memory using malloc.
- Fixed an issue where the tag name and description does not match with the driver name.
  - Root Cause: While allocating memory using malloc, the tag name and description did not match with the driver name.
  - Fix: Replaced the tags SMARTRAID with SMARTPQI when allocating memory using malloc.
- Fixed an issue where a low memory condition causes the system to freeze.
  - Root Cause: The low memory condition was not handled properly in a particular scenario.
  - Fix: Handled the low memory condition by unmapping the request.
- Fixed an issue where inquiry command was failing during device discovery.
  - Root Cause: The firmware was returning an aborted response for inquiry commands and the driver did not retry the commands.
  - Fix: Retry inquiry three times before returning a failure.
- Fixed a PSOD issue that occurs due to incorrect inbound queue selection logic for Task Management requests.
  - Root Cause: The driver was using an incorrect inbound queue to submit a Task Management request for an outstanding command. If that outstanding command had already completed, then the index for the inbound queue would be NULL and that results in a PSOD.
  - Fix: Always use inbound queue 0 for submitting a Task Management request.

2.2.3.3 Fixes and Enhancements for Solaris Driver Build 1.0.3-2323

Following are the fixes and enhancements for this release.

- Added support for Solaris 11.4 in this release.
- Fixed an issue of controller lockup due to outstanding I/O during PQI reset.
  - Root Cause: Before issuing PQI reset, the host should ensure that there is no pending I/O. Any outstanding I/O during PQI reset will lead to firmware lockup.
  - Fix: Wait for outstanding I/O to finish before issuing a PQI reset.
- Fixed an issue where inquiry command was failing during device discovery.
  - Root Cause: The firmware was returning an aborted response for inquiry commands and the driver did not retry the commands.
  - Fix: Retry inquiry three times before returning a failure.
- Fixed a PSOD issue that occurs due to incorrect inbound queue selection logic for Task Management requests.
  - Root Cause: The driver was using an incorrect inbound queue to submit a Task Management request for an outstanding command. If that outstanding command had already completed, then the index for the inbound queue would be NULL and that results in a PSOD.
  - Fix: Always use inbound queue 0 for submitting a Task Management request.
- Fixed an issue with warning message after driver Installation: “Signature verification failed for SmartPQI”
- Root Cause: Unsigned driver binary will show this message when the driver is loaded.
- Fix: The driver will be signed by the elfsign utility.

• Fixed an issue where the system crashes when the Solaris driver is loaded.
  - Root Cause: While passing the scsi_device structure to scsi_hba_probe() to get the SCSI inquiry data, the following pointer dev_info_t *sd_dev is NULL. Device is still not added to the OS. During the INQUIRY packet completion this pointer is accessed by the upper layer. This created kernel panic in Solaris 11.4.
  - Fix: Instead of calling scsi_hba_probe() to get the SCSI inquiry data, now the function calls internal PQI SCSI inquiry function.

2.2.3.4 Fixes and Enhancements for Windows Builds 106.100.0.1014/6.100.0.1014

The fixes and enhancements in this release.

• Fixed an issue where the Windows PNP WHQL tests are failing.
  - Root Cause: Unwanted PQI reset triggering the controller post in SIS mode and due to that all the PNP WHQL test cases failed.
  - Fix: PQI reset is moved out from inappropriate place.

• Fixed an issue where the system freezes during repetition of DC Off/On test.
  - Root Cause: There is a polling routine that may fail if the controller takes too long to respond, causing a potential race condition in MemAlloc.
  - Fix: Rearranged the order of these host commands and made them all use polling instead of three of them using interrupts.

2.2.3.5 Fixes and Enhancements for VMware Driver Build 1.0.3-2323

This release provides the following fixes and enhancements:

• Fixed an issue where a LUN reset completion with a service response failure results in a PSOD.
  - Root Cause: When a "LUN reset" completed with service response failure, the driver was treating it as a SCSI command completion rather than a Task Management Function (TMF) completion. The driver then tried accessing the SCSI command structure, which is not valid for a LUN reset request and caused the PSOD.
  - Fix: Set the TMF flag for LUN reset request.

• Fixed an issue of controller lockup due to outstanding I/O during PQI reset.
  - Root Cause: Before issuing PQI reset, the host should ensure that there is no pending I/O. Any outstanding I/O during PQI reset will lead to a firmware lockup.
  - Fix: Wait for outstanding I/O to finish before issuing a PQI reset.

• Fixed an issue during device discovery where an inquiry command was failing.
  - Root Cause: The firmware was returning an aborted response for inquiry commands and the driver did not retry the commands.
  - Fix: Retry inquiry three times before returning a failure.

• Fixed a PSOD issue when a TMF request timed out.
  - Root Cause: When a TMF request timed out, the driver completed the TMF as failed and cleared the internal request structure for that TMF. If the firmware completed that TMF at a later point in time, the driver will try accessing the internal request structure members and that leads to a page fault.
  - Fix: Do not free the internal resources if the TMF request times out.

• Fixed an issue where I/O tags are exhausted when TMF fails.
  - Root Cause: Tags were not freed during TMF failures, which leads to tag exhaustion condition.
• Fix: Free the tag when a TMF fails.
• Fixed a PSOD issue that occurs due to incorrect inbound queue selection logic for Task Management requests.
  ◦ Root Cause: The driver was using an incorrect inbound queue to submit a Task Management request for an outstanding command. If that outstanding command had already completed, then the index for the inbound queue would be NULL and that results in a PSOD.
  ◦ Fix: Always use inbound queue 0 for submitting a Task Management request.
• Fixed an issue where controller locks up when “arcconf getconfig” is executed.
  ◦ Root Cause: The current smartPQI driver has maximum of 10 minutes timeout for all internal and TMF commands. If the firmware does not complete the request within 10 minutes, the driver forcefully completes it and frees up the resource. This resulted in re-using the tags. Also, the driver was issuing more than what the controller’s maximum outstanding I/O request was.
  ◦ Fix: Kept infinite timeout for all internal requests (the driver initiates device discovery, cache flush, management requests, and so on) and fixed 10 minutes timeout for the Task Management functionality.
• Fixed an issue where interrupt ACK was not done properly for legacy interrupt.
  ◦ Root Cause: Incorrect implementation of interrupt ACK.
  ◦ Fix: Corrected interrupt ACK.

### Management Software Fixes

#### 2.2.4.1 Fixes and Enhancements for Arcconf/maxView Build B23600

This release includes the following fixes and enhancements for arcconf/maxView:

• Added support for Central Management of Systems in maxView.
  ◦ Root Cause: Implement feature to have Central Management of Systems in maxView.
  ◦ Fix: Implemented the exporting and importing of remote systems to be managed with maxView.
  ◦ Exposure: All previous releases
  ◦ Risk: Low.
• Fixed an issue where SMART stats are not displayed in maxView in Standalone mode.
  ◦ Root Cause: maxView in Standalone mode, was referring to the wrong address location for consuming the SMART stats buffer that resulted in a blank page.
  ◦ Fix: Added changes to maxView’s Standalone mode to refer to the right address location for consuming SMART stats buffer.
  ◦ Exposure: All previous releases
  ◦ Risk: Low.
• Fixed issue where remote management in maxView is not working when there is no network interface card is present.
  ◦ Root Cause: maxView was always looking for local IP address for communication with Redfish server which caused the problem when no network interface card is configured.
  ◦ Fix: Added changes to fallback to loop-back address where Redfish is not reachable via local IP address.
  ◦ Exposure: All previous releases
  ◦ Risk: Low.
• Fixed an issue where Active/Redundant Path Information not updated under runtime in maxView when enclosure is hot-plugged/removed.
  ◦ Root Cause: maxView is not updating the configuration of active/redundant path settings when enclosure is hot-plugged/removed.
2.3 Limitations

2.3.1 Firmware Limitations

2.3.1.1 Limitations for Firmware Release 2.30 B0

This release includes the following limitations:

- Firmware 2.30 B0 could potentially become unresponsive when a SCSI pass-through command directed towards a drive/SEP device gets lost in 'Loss of synchronization'/'Loss of Signal'.
  - **Workaround:** None
- SATA drives attached to a non-Microsemi expander may get into a failed state when upgrading the controller firmware from previous releases to this release due to the expander not clearing STP affiliation.
  - **Workaround:** Power cycle the expanders to clear the STP affiliation.

2.3.1.2 Limitations for Firmware Release 1.32 Build 0

- Firmware release 1.32b0 may become unresponsive while attempting to flash firmware or execute other RAID logical volume operations.
  - **Description:** Refer to entry "Fixed an issue where firmware may become unresponsive while attempting to flash firmware or execute other RAID logical volume operations" in the Firmware fixes section.
  - **A fix for this issue is available in the 1.60 B0 firmware release.** If a firmware flash failure is occurring, try the following workarounds:
    - **Workaround:** If there are no target devices (expanders or drives) attached to the controller, attach a target device to the controller and try the host management operation again.
    - **Workaround:** If the system is operating using UEFI, the HII tool can be used to flash the firmware to this release as outlined in the *Microsemi SmartIOC 2100/SmartROC 3100 Installation and User's Guide (ESC-2170577)*, appendix entry "Updating the SmartIOC 2100/SmartROC 3100 Controller Firmware".
    - **Workaround:** If there are target devices attached to the controller and this issue occurs or none of the workarounds can be used, contact Microsemi Support.

2.3.2 UEFI Limitations

2.3.2.1 Limitations for UEFI Build 1.3.8.2 /Legacy BIOS Build 1.3.8.1

There are no known limitations for this release.

2.3.3 Driver Limitations

2.3.3.1 Limitations for Linux Driver Build 1.2.8-026

There are no known limitations for this release.
2.3.3.2 Limitations for Windows Driver Builds 106.100.0.1014
There are no known limitations for this release.

2.3.3.3 Limitations for FreeBSD Driver Build 1.0.3-2323
There are no known limitations for this release.

2.3.3.4 Limitations for Solaris Driver Build 1.0.3-2323
This release includes the following limitations:
• UEFI Secure boot is not supported in this release.

2.3.3.5 Limitations for VMware Driver 1.0.3-2323
There are no limitations for this release.

2.3.4 Hardware Limitations
This release includes the following hardware limitations:
• Two Wire Interface (TWI) address conflicts can cause system DDR memory to not be discovered.
  • Description: The HBA1100 boards include two TWI targets on the host-facing SMBUS interface with the following slave addresses:
    ▪ 0xA0 – Field Replaceable Unit (FRU) SEEPROM
    ▪ 0xDE – PBSI (default)
    According to the JEDEC specification, the default TWI addresses for the DDR SPD is 0xA0-0xAE (the spec uses 7 bit addressing which is 0x50-0x57). On platform system board designs with SMBUS wiring that has both PCIe slots and DDR slots shared on the same TWI bus, the TWI devices for the DDR and Smart controller are exposed to address conflicts which can result in the system memory not being discovered. The Smart controller PBSI interface defaults to a value of 0xDE (0x6F in 7-bit addressing) and is not a problem unless it is changed to an address that conflicts with the JEDEC defined values. The Smart controller FRU SEEPROM is hardwired to 0xA0.
  • Workaround: None available. If this issue is encountered, contact your Microsemi support engineer to determine the next steps for your system.
  • Performance with workaround: Not applicable
  • Performance without workaround: Not applicable

2.3.5 Management Software Limitations

2.3.5.1 Limitations for Arcconf and maxView Build B23600
There are no known limitations for this release.
3 Updating the Board Firmware for PQI Operation

This section describes how to update the board's firmware components to the latest release.

3.1 Updating Controllers to latest (PQI) Firmware

This procedure describes how to prepare your board to be programmed with the latest board PQI firmware.

**Note:** Complete these procedures exactly as described for proper functionality. If you do not follow all of the steps correctly, you could encounter unusual runtime behavior.

**Flashing the board to the latest PQI firmware:**

This section describes how to update all the firmware components on HBA 1100 Adapter boards to the latest release.

If the controller is currently running 1.60 b0 firmware or newer, follow these steps:

1. **Mandatory:** Flash the target with the provided "SmartFWx100.bin" image with arcconf/maxView software.
2. **Mandatory:** Cold boot the system to刷新 all components.

If the controller is currently running 1.32 b0 firmware, follow these steps:

1. **Mandatory:** Flash the target with the provided “SmartFWx100.bin” image with arcconf/maxView software.
   - If the arcconf/maxView software becomes unresponsive or hangs then power cycle the system to recover and refer to firmware limitation section *Limitations for Firmware Release 1.32 Build 0* on page 17.
2. **Mandatory:** If flashing completes, cold boot the system to refresh all components.

If the controller is currently running 1.04 b0 firmware, follow these steps:

1. **Mandatory:** Flash the controller with the provided “SmartFWx100_v1.29_b314.bin” image with arcconf/maxView software.
2. **Mandatory:** Reboot the system to refresh all components.
3. **Mandatory:** Flash the target with the provided "SmartFWx100.bin" image with arcconf/maxView software.
4. **Mandatory:** Cold boot the system to refresh all components.

At this point, the controller would be updated and would be ready to use. Install the SmartPQI driver and the latest version of the Arcconf/maxView management utility to monitor and configure the controller.

**Note:** Downgrading firmware could lead to unexpected behavior due to an incompatibility in SEEPROMs between this release and the prior release.
4 Installing the Drivers

See the "Microsemi Adaptec® HBA 1100 Series Host Bus Adapters Installation and User's Guide (ESC-2161232)" for complete driver installation instructions.
Microsemi makes no warranty, representation, or guarantee regarding the information contained herein or the suitability of its products and services for any particular purpose, nor does Microsemi assume any liability whatsoever arising out of the application or use of any product or circuit. The products sold hereunder and any other products sold by Microsemi have been subject to limited testing and should not be used in conjunction with mission-critical equipment or applications. Any performance specifications are believed to be reliable but are not verified, and Buyer must conduct and complete all performance and other testing of the products, alone and together with, or installed in, any end-products. Buyer shall not rely on any data and performance specifications or parameters provided by Microsemi. It is the Buyer’s responsibility to independently determine suitability of any products and to test and verify the same. The information provided by Microsemi hereunder is provided “as is, where is” and with all faults, and the entire risk associated with such information is entirely with the Buyer. Microsemi does not grant, explicitly or implicitly, to any party any patent rights, licenses, or any other IP rights, whether with regard to such information itself or anything described by such information. Information provided in this document is proprietary to Microsemi, and Microsemi reserves the right to make any changes to the information in this document or to any products and services at any time without notice.