## Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Revision Date</th>
<th>Details of Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>February 2020</td>
<td>Update for SR 2.5.2</td>
</tr>
<tr>
<td>15</td>
<td>October 2019</td>
<td>Update for SR 2.5</td>
</tr>
<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................................................................................................................................11
    2.2.3 Driver Fixes...............................................................................................................................11
    2.2.4 Management Software Fixes..................................................................................................14
  2.3 Limitations...................................................................................................................................14
    2.3.1 Firmware Limitations..............................................................................................................14
    2.3.2 UEFI Limitations......................................................................................................................15
    2.3.3 Driver Limitations.....................................................................................................................15
    2.3.4 Hardware Limitations...............................................................................................................15
    2.3.5 Management Software Limitations.........................................................................................16

3 Updating the Board Firmware for PQI Operation............................................................................17
  3.1 Updating Controllers to latest (PQI) Firmware...........................................................................17

4 Installing the Drivers........................................................................................................................18
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 the following table.

Table 1 • Release Summary

<table>
<thead>
<tr>
<th>Solutions Release</th>
<th>2.5.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package Release Date</td>
<td>February 21, 2020</td>
</tr>
<tr>
<td>Firmware Version</td>
<td>2.92 B0(^{1,2}) (basecode 06.05.003.000)</td>
</tr>
<tr>
<td>UEFI Version</td>
<td>1.3.10.2</td>
</tr>
<tr>
<td>Legacy BIOS</td>
<td>1.3.10.2</td>
</tr>
<tr>
<td>Driver Versions</td>
<td></td>
</tr>
<tr>
<td>Windows SmartPQI:</td>
<td></td>
</tr>
<tr>
<td>• Windows 2012/2016/2019: 106.178.0.1009</td>
<td></td>
</tr>
<tr>
<td>• Windows 7/2008: 6.102.0.1026</td>
<td></td>
</tr>
<tr>
<td>Linux SmartPQI:</td>
<td></td>
</tr>
<tr>
<td>• RHEL 6/RHEL 7/RHEL 8/SLES 12/SLES 15: 1.2.12-025</td>
<td></td>
</tr>
<tr>
<td>• Ubuntu 16/18: 1.2.12-025</td>
<td></td>
</tr>
<tr>
<td>• CentOS 6/7/8: 1.2.12-025</td>
<td></td>
</tr>
<tr>
<td>• Debian 8/9: 1.2.12-025</td>
<td></td>
</tr>
<tr>
<td>VMware SmartPQI:</td>
<td></td>
</tr>
<tr>
<td>• VMWare ESXi 6.0/6.5/6.7: 1.0.4.3017</td>
<td></td>
</tr>
<tr>
<td>FreeBSD/Solaris SmartPQI:</td>
<td></td>
</tr>
<tr>
<td>• FreeBSD 11/12: 1.0.4.3017</td>
<td></td>
</tr>
<tr>
<td>• Solaris 11: 1.0.4.3017</td>
<td></td>
</tr>
</tbody>
</table>

arconf/Maxview       B23699

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>X</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 romupdate</td>
<td>The command allows to upgrade/downgrade the firmware and BIOS image to the controller.</td>
<td>Refer to Table 8 • Host Management Utilities on page 8</td>
</tr>
<tr>
<td>maxView firmware upgrade wizard</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 8 • Host Management Utilities on page 8</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>x64</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 for Intel/AMD x64**

<table>
<thead>
<tr>
<th>Drivers</th>
<th>Intel/AMD x64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red Hat Enterprise Linux/CentOS 8.1, 8.0</td>
<td>X</td>
</tr>
<tr>
<td>Red Hat Enterprise Linux/CentOS 7.7, 7.6, 7.5¹, 7.4, 7.3</td>
<td>X</td>
</tr>
<tr>
<td>Red Hat Enterprise Linux/CentOS 6.10, 6.9¹</td>
<td>X</td>
</tr>
<tr>
<td>SuSE Linux Enterprise Server 12¹, SP5, SP4, SP3¹, SP2</td>
<td>X</td>
</tr>
<tr>
<td>SuSE Linux Enterprise Server 15, SP1¹</td>
<td>X</td>
</tr>
</tbody>
</table>
### Drivers

<table>
<thead>
<tr>
<th>Drivers</th>
<th>Intel/AMD x64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Linux 7.5 with UEK4u7 (4.1.12-124)</td>
<td>X</td>
</tr>
<tr>
<td>Oracle Linux 7.6 with UEK5u2 (4.14.35)</td>
<td>X</td>
</tr>
<tr>
<td>Oracle Linux 7.7 with UEK5u2 (4.14.35)</td>
<td>X</td>
</tr>
<tr>
<td>Oracle Linux 8.0 with UEK5</td>
<td>X</td>
</tr>
<tr>
<td>Ubuntu 19.01</td>
<td>X</td>
</tr>
<tr>
<td>Ubuntu 18.04.3, 18.04.2, 18.04.1</td>
<td>X</td>
</tr>
<tr>
<td>Ubuntu 16.04.5, 16.04.4</td>
<td>X</td>
</tr>
<tr>
<td>Debian 10</td>
<td>X</td>
</tr>
<tr>
<td>Debian 9.9</td>
<td>X</td>
</tr>
<tr>
<td>Citrix xenServer 8.0,7.6</td>
<td>X</td>
</tr>
<tr>
<td>Fedora 30</td>
<td>X</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 • Linux SmartPQI Drivers for ARM

<table>
<thead>
<tr>
<th>Drivers</th>
<th>Cavium ThunderX2 ARM x64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red Hat Enterprise Linux/CentOS 8.1, 7.6</td>
<td>X</td>
</tr>
<tr>
<td>SuSE Linux Enterprise Server 15, SP1</td>
<td>X</td>
</tr>
<tr>
<td>Ubuntu 18.04.2, 18.04.1</td>
<td>X</td>
</tr>
<tr>
<td>BCLinux 7.6</td>
<td>X</td>
</tr>
</tbody>
</table>

### Table 7 • FreeBSD, Solaris, and VMware SmartPQI Drivers

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

### Table 8 • 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, Linux x64, VMware ESXi 5.5/6.0, XenServer, FreeBSD x64, Solaris x86, Linux ARM</td>
<td>See the Arcconf download package for the OS-applicable installation executable.</td>
</tr>
<tr>
<td>ARCCONF for UEFI</td>
<td></td>
<td>Included as part of the firmware downloadable image.</td>
</tr>
<tr>
<td>maxView Storage Manager</td>
<td>Windows x64, Linux x64, VMware ESXi 5.5/6.0, XenServer</td>
<td>See the maxView Storage Manager download package for the OS-applicable installation executable.</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>
2 What is New?

2.1 Features

The following table lists features supported for this release.

Table 9 • 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>Out of Band interface selection support of MCTP or PBSI</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Flash Support</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>MCTP BMC Management</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

2.2 Fixes

2.2.1 Firmware Fixes

2.2.1.1 Fixes and Enhancements for Firmware Release 2.92 B0

This release includes the following fixes and enhancements:

- Fixed an issue where a controller would be stuck in survival mode when no temperature sensors were included in the design.
  - Root Cause: During boot, firmware expects to find at least one temperature sensor as required by the hardware reference design. If, for some reason, no temperature sensors are included in the design the firmware would incorrectly use other sensor configuration data to configure the survival mode temperature thresholds. This results in the board falling immediately into survival mode.
  - Fix: When attempting to set up the survival mode thresholds at boot, validate the sensor data being used is associated with a temperature sensor.
  - Risk: Very Low
- Fixed an issue where a bad drive could cause a controller to not be found or an OS to not boot after a reboot.
What is New?

- **Root Cause:** During discovery of the drive after a reboot, the controller firmware was repeatedly attempting to start the drive but the drive was repeatedly reporting 04/xx/xx check conditions (HARDWARE ERROR) to TestUnitReady commands. Parallel to this, the controller has a limit on the amount of time afforded to topology discovery and if that time has been exceeded then the UEFI/Bios driver or OS driver will give up waiting for the controller to become ready. In this case, the error/recovery logic for these devices with fatal errors was causing this timeout to be exceeded.
  - **Fix:** Fail a drive immediately rather than attempt repeated error recovery after it reports the first HARDWARE ERROR status on TestUnitReady because these devices are not expected to suddenly become good again.
  - **Risk:** Low

- Fixed an issue where direct-attach drives not attached to a backplane where no enclosure management schema is defined results in incorrect device location reporting.
  - **Root Cause:** When there is no enclosure management schema defined for a connector, e.g. not SGPIO or some I2C schema, and a direct-attach drive is attached, then code review identified a scenario where this type of device connectivity would result in incorrect device location reporting due to association with a non-existent virtual device associated with the enclosure management schema.
  - **Fix:** Removed logic that was blindly associating devices in this situation with the non-existing enclosure management target.
  - **Risk:** Low

- Fixed a controller lockup issue when toggling one of the path to an enclosure in dual domain configuration.
  - **Root Cause:** During enabling/disabling one of the path to an enclosure in dual domain configuration, an incorrect timeout value is set before the path can be added back.
  - **Fix:** Conditions that set the incorrect timeout value is modified to have the correct value, allowing the path to be added back.
  - **Risk:** Low

- Fixed an issue where wrong connector information is displayed for SATA drives after a cable hot plug event on dual I/O module enclosure configuration.
  - **Root Cause:** In dual I/O module enclosure configuration with SATA drives, when one of the cables is hot removed/added the connector information is not updated properly leading to wrong connector information to be exported for display in host tools.
  - **Fix:** Active connector information is updated correctly for SATA drives when cable is hot added on dual I/O module enclosure configuration.
  - **Risk:** Low

- Fixed an issue where failed drives are physically present in its slot are not reported correctly in installed drive map.
  - **Root Cause:** When a drive fails and is physically present in its slot, corresponding installed drive map bits are not updated to reflect the correct status to be exported for display in host tools.
  - **Fix:** Added more conditions to check for failed drive and its PHY active status to correctly set the installed drive map bit.
  - **Risk:** Low

- Fixed an issue where device drops are observed after multiple create/delete drive zone group configuration commands.
  - **Root Cause:** When multiple create/delete drive zone group configurations are performed, an incorrect out of bound device index value is assigned and causes devices to drop out of configuration.
  - **Fix:** Device index is correctly reinitialized during multiple create/delete drive zone group configurations.
  - **Risk:** Low
• Fixed an issue where status LED doesn’t blink during sanitize operation on expander attached drives configuration.
  ◦ Root Cause: When drives behind expander undergo sanitize operation, controller updates the appropriate bit in SES pages to blink the status LED, but the command fails to reach expander due to incorrect sanity checks.
  ◦ Fix: Sanity checks are refined to allow the SES page update for the drives undergoing sanitize operation behind expander to blink the status LEDs.
  ◦ Risk: Low

• Fixed an issue where all zeros response was sent for ATA PASSTHROUGH commands, such as SMART READ DATA, through Out of Band (OOB) host transport.
  ◦ Root cause: For OOB transport commands transfer length is calculated based on incoming CDBs solely and firmware was parsing the CDBs based on 16 byte/12 byte SCSI CDB which resulted in incorrect transfer length. In addition, when completing OOB commands back to host, firmware was not copying data back for ATA Passthrough since the command direction was set incorrectly.
  ◦ Fix: Firmware has added support in OOB host transport path for parsing the ATA PASSTHROUGH commands' transfer length based on T10 SAT specification. Command direction is calculated based on T_DIR bit incoming CDBs for ATA PASSTHROUGH commands for copying the data back to host accordingly.
  ◦ Risk: Low

## 2.2.2 UEFI Fixes

**Note:** Microsoft signed and secure boot is supported.

### 2.2.2.1 Fixes and Enhancements for UEFI Build 1.3.10.2/Legacy BIOS Build 1.3.10.2

This release includes the following UEFI fixes and enhancements:

• Added option to configure OOB (PBSI/MCTP) settings under Modify Controller Settings.
• Fixed an issue where the Disk Utilities menu shows controller as additional device.
  ◦ Root Cause: Improper logic for filtering controller device from other physical devices.
  ◦ Fix: Corrected logic for filtering controller device from other physical devices.
  ◦ Risk: Low

This release includes the following Legacy BIOS fixes and enhancements:

• Fixed an issue where CTRL A disk utilities shows wrong device port, box, and bay.
  ◦ Root Cause: Check was missing for `alt_paths_phys_conn[0]`.
  ◦ Fix: Verify `alt_paths_phys_conn[0]` against 00 and if it is equal over write the content with `--`.
  ◦ Risk: Low

## 2.2.3 Driver Fixes

### 2.2.3.1 Fixes and Enhancements for Linux Driver Build 1.2.12.025

This release includes the following enhancements:

• Added support for the following: RHEL8u2 Beta 1, RHEL7u8 Snapshot-3, SLES15SP2 Snapshot-2, SLES 12 SP5 GMC, and BCLinux 7u6 ARM only.
• Fixed an issue where when Secure Memory Encryption (SME) is enabled, the smartPQI driver doesn’t work, which results in the failure of kernel boot because it fails to allocate PQI error buffer.
  ◦ Root Cause: The coherent DMA mask value caused the driver to fall back to Software Input Output Translation Lookaside Buffer (SWIOTLB) when SME is active.
• Fix: For correct operation, call the `dma_set_mask_and_coherent()` to properly set the mask for both streaming and coherent, to inform the kernel about the device's DMA addressing capabilities.
  ◦ Risk: Medium

### 2.2.3.2 Fixes and Enhancements for FreeBSD Driver Build 1.0.4.3017

Following are the fixes and enhancements for this release.

- Fixed an issue where OS crashes during physical drive hot removal. The physical drive hot removed while heavy I/O is running on it from FIO and that leads to OS crash.
  ◦ Root Cause: Driver is not waiting for physical drive I/O completions before wiping out the device memory.
  ◦ Fix: Waiting for outstanding commands using atomic operations with maximum of 30 seconds interval.
    ▪ Set IOBypass enable flag to false, if the driver receives IOBypass response error with status as IOBypass path disabled. The flag will stop the further in flight I/O’s in IOBypass path.
  ◦ Risk: Medium

- Fixed an issue where the driver is currently processing "softs->max_outstanding_io -1" outstanding commands instead of "softs->max_outstanding_io" commands.
  ◦ Root Cause: The driver is currently processing one command less than the maximum outstanding IO commands.
  ◦ Fix: Added condition to process all outstanding commands which includes:
    1. Releasing used tags after completing outstanding commands. While unloading a driver it is not necessary but it should be cleaner to have it.
    2. Assigning NULL pointer to device after freeing up memory.
  ◦ Risk: Low

- Fixed an issue where FreeBSD crashes while issuing firmware test lockup during I/Os.
  ◦ Root Cause: Driver was not clearing the RCB structure after I/O completion. When controller went offline, driver was using the RCB structure values to complete the pending I/Os. Since the RCB fields were not cleared, there was an I/O double completion and leading to OS crash.
  ◦ Fix: Reset the RCB before completing I/Os.
  ◦ Risk: Low

- Fixed an issue during dynamic unload of SmartPQI driver.
  ◦ Root Cause: Controller is in complete shutdown while unloading (or) detaching a driver. While unloading a driver, the cache flush is called with PQISRC_SHUTDOWN event type in `smartpqi_shutdown` function and that leads to complete shutdown of the controller. Hence, the next driver load fails.
  ◦ Fix: During driver unload, we are calling Cache flush event type with PQISRC_NONE_CACHE_FLUSH_ONLY. This allows dynamically load of driver.
  ◦ Risk: Medium

### 2.2.3.3 Fixes and Enhancements for Solaris Driver Build 1.0.4.3017

Following are the fixes and enhancements for this release.

- Fixed an issue where OS might crash during physical drive hot removal. The physical drive got removed while heavy I/O is running on it from FIO and that leads to OS crash.
  ◦ Root Cause: Driver is not waiting for physical drive I/O completions before wiping out the device memory.
• Fix: Waiting for outstanding commands using atomic operations with maximum of 30 seconds interval.
  ▪ Set IOBypass enable flag to false, if the driver receives IOBypass response error with status as IOBypass path disabled. The flag will stop the further in flight I/O's in IOBypass path.
• Risk: Medium

2.2.3.4 Fixes and Enhancements for Windows Build 106.178.0.1009

• Fixed an issue where the SRB conversion routine failed to properly identify bidirectional flags which meant it would always fall back to read.
  ▪ Root Cause: An if was used when instead of "else if".
  ▪ Fix: Replaced second if with "else if" in order to ensure the first if, which checks whether bidirectional is used when both IN and OUT flags are set.
  ▪ Risk: Medium. Assumes the RAID stack knows how to handle bidirectional flags.

• Fixed an issue where a bogus CDB in IOCTL SAS Passthru caused unsafe memcpy and double completion at host side.
  ▪ Root Cause: Forgot to add return statement on this error condition.
  ▪ Fix: Added the return statement. Since user provided an unsupported IOCTL, the request cannot be continuously parsed.
  ▪ Risk: Low

• Fixed an issue where the DP WLK—hot replace test is failing; the multi-tag table memory was not getting released.
  ▪ Root Cause: Since OFA was introduced and the multi-tag table is now tied to the operational queues it must always recreate operational queues.
  ▪ Fix: Rework code to always rebuild operational queues and a new multi-tag table. Also, always preserve Records, per lun memory, RAID map data, and I/O error buffer.
  ▪ Risk: Medium

• Fixed an issue where the Static Driver Verifier (SDV) reports null check defects.
  ▪ Root Cause: Driver is not checking if pRecord is a NULL pointer or not after getting a record from MapTraverseTrieStartStop function and it catches in SDV.
  ▪ Fix: Forceful check NULL pointer before accessing device records.
  ▪ Risk: Low

• Fixed an issue related to the static variables. Since the static variables are global to all controller instances, these counters should have been made local to the device extension rather than system-wide.
  ▪ Root Cause: Used static instance rather than controller instance.
  ▪ Fix: Used controller context for tracking this information.
  ▪ Risk: Low

• Fixed an issue where system would crash when doing unnecessary initialization of the multi-tag table after declaring controller lockup.
  ▪ Root Cause: Unnecessary initialization of the multi-tag table.
  ▪ Fix: Removed unnecessary initialization when declaring a controller lockup.
  ▪ Risk: Low

2.2.3.5 Fixes and Enhancements for VMware Driver Build 1.0.4.3017

Following are the fixes and enhancements for this release:

• Added support for VMware 7.0 RC1.
• Fixed an issue with controller offline status not being detected after triggering a controller lockup.
Root Cause: Driver is not detecting controller offline status after triggering a controller lockup when RBOD is connected. Current SmartPQI driver uses common Timer queue for all the timers—heartbeat, rescan and host wellness update. When one of the timer callbacks is stuck, the others will not get a chance to run. Here the rescan timer was executing and the controller went offline (because of test lockup). Rescan timer function was waiting for the response from the firmware. Since the controller is locked up, driver will not get any response. Controller lockup is not getting detected by the driver since the heartbeat timer is not getting chance to run.

Fix: Added new timer queue for heartbeat timer.
Risk: Low

Fixed an issue where firmware was getting lockup during SOB replay.

Root Cause: Heartbeat timer handle will run in each 5 seconds and issue firmware lockup if there is no heartbeat update for 5 seconds. During "sob_replay", firmware updated the heartbeat after 8 seconds. This results in NMI issued from the driver.

Fix: Increase the heartbeat timeout to 10 seconds.
Risk: Low

2.2.4 Management Software Fixes

2.2.4.1 Fixes and Enhancements for Arcconf and maxView Build B23699

This release includes the following fixes and enhancements.

- Added support to configure PBSI/MCTP: arcconf provides support to switch the BMC interface between PBSI and MCTP along with the other related configuration changes. arcconf can also reset or disable the BMC interface. A power cycle will be needed to reflect the newly set BMC interface settings.

2.3 Limitations

2.3.1 Firmware Limitations

2.3.1.1 Limitations for Firmware Release 2.92 B0

This release includes the following firmware limitation:

- 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.

- When I/Os are performed on drives that respond slowly or which do not respond to READ or WRITE commands, and when Secure Erase is performed on other SATA drives, I/Os become stalled for a period of time. The time the I/Os are paused depends directly on the amount of unflushed data in the cache and speed with which the device responds to error recovery.
  - Workaround: None

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 1.3.10.2/Legacy BIOS Build 1.3.10.2

There are no known limitations for this release.

### 2.3.3 Driver Limitations

#### 2.3.3.1 Limitations for Linux Driver Build 1.2.12.025

There are no known limitations for this release.

#### 2.3.3.2 Limitations for Windows Driver Builds 106.178.0.1009

There are no known limitations for this release.

#### 2.3.3.3 Limitations for FreeBSD Driver Build 1.0.4.3017

There are no known limitations for this release.

#### 2.3.3.4 Limitations for Solaris Driver Build 1.0.4.3017

There are no known limitations for this release.

#### 2.3.3.5 Limitations for VMware Driver Build 1.0.4.3017

There are no known 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 B23699

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 refresh 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 14.

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’s product warranty is set forth in Microsemi’s Sales Order Terms and Conditions. Information contained in this publication is provided for the sole purpose of designing with and using Microsemi products. Information regarding device applications and the like is provided only for your convenience and may be superseded by updates. Buyer shall not rely on any data and performance specifications or parameters provided by Microsemi. It is your responsibility to ensure that your application meets with your specifications. THIS INFORMATION IS PROVIDED “AS IS.” MICROSEMI MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL MICROSEMI BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL OR CONSEQUENTIAL LOSS, DAMAGE, COST OR EXPENSE WHATSOEVER RELATED TO THIS INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROSEMI HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROSEMI’S TOTAL LIABILITY ON ALL CLAIMS IN RELATION TO THIS INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, YOU PAID DIRECTLY TO MICROSEMI FOR THIS INFORMATION. Use of Microsemi devices in life support, mission-critical equipment or applications, and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend and indemnify Microsemi from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microsemi intellectual property rights unless otherwise stated.