MPT Windows Driver (SYMMPI) Release Notes

2000, XP, Server 2003

Version 1.30.04

7/7/2009

Compatibility:

LSI53C1020 / LSI53C1020A / LSI53C1030
LSIFC919 / LSIFC919X / LSIFC929 / LSIFC929X / LSIFC949X / LSIFC949E
(The SAS controllers below are not supported by the SYMMPI driver on
Server 2003. The LSI_SAS driver must be used instead.)
LSISAS1064 / LSISAS1068 / LSISAS1078 / LSISAS1064E / LSISAS1068E

Special Notes for 1.30.04:

- Starting with release 1.30.03.00, 4 new no driver entries were added to support new SAS enclosures.
- Driver description strings use the string 'LSI' instead of 'LSI Logic' starting with version 1.25.08. All user documentation has been changed to reflect the new name.
- Starting with version 1.25.07, the Windows driver does not require PCI Port I/O resources. However, the elimination of Port I/O resources is controlled by a registry entry that is not part of the standard driver package. This is not included as it can cause bluescreens if a driver is downgraded to an earlier version which does require Port I/O resources. A special INF file that sets the NoPortIoUsage registry entry can be requested from LSI.
- Driver installation packages have been restructured, with each package supporting a particular OS version/platform combination.

- Versions of SYMMPI at or before 1.24.03 can cause memory corruption under a very specific system environment. See the "Defects fixes" section under version 1.24.04
- The BIOS Config Utility settings for sync, wide, and adapter SCSI ID are the only ones honored by the Windows driver. Other settings are overridden by the Windows OS.
- IOCTL calls cannot be issued to an adapter that has no visible devices on the bus (Windows OS restriction).
- IOCTL calls are not supported on Windows 9X_ME.
- For full driver upgrade functionality in flashless environments, Windows drivers at revision 1.08.12 and above and 1030 F/W at revision 1.00.14.00 or above should be used.
- LSI Pseudo Device support is enabled (via a registry entry) by default on Fibre Channel devices, starting with version 1.09.09. (Not enabled for Server 2003. A system registry entry provides the same function.)
- When using 1030 IR F/W version 1.03.09 or later, driver version 1.09.05 or later must be used for hibernation and crash dump to be supported properly.
- Driver versions from 1.09.07 through 1.09.10 have the PCI-X Multiple Outstanding Split Transactions setting in config space set to 1 for 919X and 929X. This can result in data corruption. Fixed in 1.09.11.
- Driver version 1.09.15 is required for proper operation with MyStorage.
- Driver version 1.09.90 or above is required for 1020A flashless environments.
- Multiple adapters of the same type should all have the adapter BIOS enabled if one of the adapters controls the boot device.
- To enable End-to-End Data Protection (EEDP) support for the 949X/939X devices, the registry entry "EEDP_T10_Enable=1;" must be added to the DriverParameter string.
- Beginning with version 1.20.17, Serial Attached SCSI (SAS) device support for Server 2003 is supplied only by the LSI_SAS.SYS StorPort-based driver.

- For the Windows 2000 operating systems a hotfix is required for proper support of drive hot plug operations. This is described in Knowledge Base article 867818. More information can be found at: http://support.microsoft.com/default.aspx?scid=kb;en-us;867818
- For Fibre Channel adapters, the default setting for host driver persistency table support is now on. To turn this off, place the string LUN_Pers=0; in the DriverParameter string in the registry.
- For SAS adapters (Win2K, XP x86 and IA64), to have a pseudo device be active (with the EnablePsuedoDevice registry entry) with SAS IT F/W version 10.14 or later, version 1.21.30 or later of the Symmpi driver must be used.

Current Errata:

- None

Version 1.30.04

Major Changes From Version 1.30.02:

General Changes

Functionality
- None.

Defect fixes
- None.

Enhancements
- None.

Notes
- The driver version was incremented for general release. This release captures all changes since driver version 1.29.03.

Version 1.30.03

Major Changes From Version 1.30.02:

General Changes

Functionality
- None.

Defect fixes
- None.

Enhancements
- Starting with release 1.30.03.00, 4 new no driver entries were added to support new SAS enclosures.

Version 1.30.02

Major Changes From Version 1.30.01:

General Changes

Functionality

- None.

Defect fixes

- System can bluescreen if connected to a target that supports Broadcast Async Event notification and a task management reset is issued to the target.
  – Seen by: LSI test lab
  – To reproduce: Connect a SAS target that supports Broadcast Async Event notification. Issue I/O's and periodically issue task management resets. If the system bluescreens the problem is reproduced.
  – Root cause: When a Broadcast Async Event (BAE) notification is received the driver uses special MPI message frames to issue config page and task management requests. These frames have a special cotext value (BC_AEN_CONTEXT) so the replies can be trapped by the ISR and routed to the BAE routine. A coding error was allowing some of these special replies to fall into code that used the context value as an index into the IO tracking array.
  – Fix: Added a specific check for BC_AEN_CONTEXT early in the ISR to properly route these replies to the BAE routine.

Version 1.30.01

Major Changes From Version 1.29.03:

General Changes

Functionality

- Added LSI A.1 Expander NoDriver entries to LSINODRV.INF files to suppress asking for drivers when these enclosures are present.
- Added custom OEM tape drive in CD-ROM mode detection for In-Line Data Padding (IDP) (parallel SCSI only).

Defect fixes

SRB_FUNCTION_RESET_BUS was not completing the SRB properly.
- Seen by OEM customer.
- To reproduce: Using a kernel mode service, issue an SRB with SRB_FUNCTION_RESET_BUS. The SRB will not be properly completed back to the service. The SRB will have an SRB status of SRB_STATUS_PENDING.
- Fix: Add code to properly complete the SRB back to the port driver.

Version 1.29.03

Major Changes From Version 1.29.02:

General Changes

Functionality

- Added two more OEM SAS Enclosure NoDriver entries to LSIPSEUD.INF files to suppress asking for drivers when these enclosures are present.

Defect fixes

None.

Version 1.29.02

Major Changes From Version 1.29.01:

General Changes

Functionality

- Added more OEM SAS Enclosure NoDriver entries to LSIPSEUD.INF files to suppress asking for drivers when these enclosures are present.

Defect fixes

- Backed out the change for the problem in the way the driver calculates Global Credits available in the controller.
  – Seen by internal testing.
  – Root cause: The change to count each buffer used actually makes the problem of running out of resources worse, because we are forced to call ScsiPortBusy more often than the old way, and the change doesn't solve the problem it was intended to solve (avoid reset failures due to I/O's stuck in the request buffer not being terminated by the reset).
  – Fix: Backout this change made in v1.29.01.
- The WHQL ChkINF test fails for XP x86, XP IA64 and Win2K.
  – Seen during WHQL ChkINF testing.
  – Root cause: The registry entry for TimeOutValue in SYMMPI.INF has a typo. The line 'HKLM,SYSTEM\CurrentControlSet\Services\Disk\TimeOutValue,0x00010001,0x000000A0' Should be changed to 'HKLM,SYSTEM\CurrentControlSet\Services\Disk,TimeOutValue,0x00010001,0x000000A0'.
  – Fix: Replaced the '\' after 'Disk' with ','.

Version 1.29.01

Major Changes From Version 1.28.03:

General Changes

Functionality
- Removed unused header files from being included in project.

Defect fixes
- Aborted I/O is not being retried and returns GOOD status and causes data corruption.
  - Seen by OEM.
  - Root cause: Driver was not checking the status of an aborted I/O resulting in GOOD status being returned and therefore the I/O was not retried. Some new controllers return the SCSISTAT_TASK_ABORTED status, but this is not checked and a GOOD status is returned.
  - Fix: Check status of aborted I/O to see if it is SCSISTAT_TASK_ABORTED. If so, return SRB_STATUS_ERROR instead of SCSISTAT_GOOD.
- Config Page replies can be processed as Broadcast Async Event message replies.
  - Seen by code review.
  - Root cause: Driver does not separate processing of BAE replies from config page replies.
  - Fix: Isolate processing of BAE messages from config page and TM reset requests by allocating a dedicated static message frame for BAE processing only.
- Problem in the way the driver calculates Global Credits available in the controller.
  - Seen by internal testing.
  - Root cause: The F/W Global Credits value is the number of message frames allocated by F/W to process I/O's. However,

when processing large I/O's with many chain buffers, the F/W will pull all of the chains into controller memory at the start of I/O processing, using additional message frames. The host driver was using Global Credits as the number of concurrent I/O's it can issue to the F/W. With a heavy I/O load of large I/Os many I/O's will be backed up in the H/W Request FIFO. I/O's with shorter timeouts (such as bus enumeration I/O's) may time out due to this backlog. Task Management commands issued to clean up the I/O may not return the timed out I/O if it is still pending in the Request FIFO.

– Fix: Modify the driver to count the number of message frames and chain buffers as a running count of the total number of buffers used, and compare that to the F/W Global Credits. When the value exceeds the Global Credits the driver will stop the new I/O queue until some of the outstanding I/O's complete (reducing the number of outstanding buffers used).

Version 1.28.03

Major Changes From Version 1.28.02:

General Changes

Functionality

- Started logging internal device reset completion event if LogExceptionEvents is turned on.
- Added registry method for entering the four dwords of Product Specific data needed for immediate diagnostic buffers.

Defect fixes

- Modified OEM SAS Enclosure NoDriver entry in LSIPSEUD.INF files.
  – Seen by OEM. Wrong PnP ID was given – OEM made a modification.
  – Root cause: Bad original PnP ID from OEM.
  – Fix: Changed to correct ID.
- In FwDiagPostImmediateBuffers function, 'loop' counter is not big enough to account for maximum loop count.
  – Seen by internal test.
  – Root cause: Used U8 instead of U16 for loop counter.
  – Fix: Changed to use U16 for loop counter.
- Modify OEM timeout value in registry.
  – Seen by OEM.
  – Root cause: Timeout value was not large enough to support some systems.
  – Fix: Changed INF files to increase timeout value from 65 to 160 seconds.

Version 1.28.02

<u>Major Changes From Version 1.28.01:</u>

<u>General Changes</u>

Functionality
- None.

Defect fixes
- INF files are not consistent and some have incorrect OEM information.
  – Seen by OEM.
  – Fix: Change OEM sections of INF files to be correct.
- F/W diagnostic buffer IOCTLs do not return the proper error codes.
  – Seen by internal test.
  – Root cause: When issuing a F/W diagnostic buffer IOCTL with a properly formatted and sized buffer, the IOCTL will succeed, but the ReturnCode in the SRB_IO_CONTROL structure will be 2 instead of 0. Also, in case of an error, a generic IOCTL error is returned instead of the SRB_IO_CONTROL ReturnCode.
  – Fix: Return the proper error code and change the error code value of 2 to 3 so that applications can know that this is a real error. The old value of 2 will tell an application that the error can be ignored if using an older driver. Also, add a new ReturnCode so that if an error occurs the entire buffer is returned so that an application can view the IOCTL error code. If an application uses this new ReturnCode and the IOCTL ReturnCode upon completion is not successful, the entire buffer is returned.

Version 1.28.01

Major Changes From Version 1.27.03:

General Changes

Functionality

- In addition to code 0xDEAD (diag reset), added a bluescreen with code 0xFACE when FW faults. The code to force the bluescreens was changed to work for released builds of the driver as well as checked builds. These bluescreens will occur only if the registry entry "StopOnDiagReset" is active.
- Updated to MPI headers 1.05.19.

Defect fixes

- None.

Version 1.27.03

Major Changes From Version 1.27.02:

General Changes

Functionality
- Updated to MPI headers 1.05.18.

Defect fixes
- Modified OEM SAS Enclosure NoDriver entry in LSIPSEUD.INF files.
  - Seen by OEM. Wrong PnP ID was given – OEM made a modification.
  - Root cause: Bad original PnP ID from OEM.
  - Fix: Changed to correct ID.

Version 1.27.02

Major Changes From Version 1.27.01:

General Changes

Functionality
- Added more OEM SAS Enclosure NoDriver entries to LSIPSEUD.INF files to suppress asking for drivers when these enclosures are present.
- When Persistency Table is full and Log Exception Events is enabled, an error is logged to the Windows System Event log.

Defect fixes
- Bluescreen when reply frame has HBA API flag set and original request did not.
  – Seen by OEM during Windows 7 testing.
  – Root cause: Bad reply frames are returned that have the HBA API flag set, even though the original request did not. When reply is processed, bluescreen occurs because there is no outstanding HBA API request.
  – Fix: Add checks in HbaApiWmiCompletion and CheckHbaApiScsiIo to make sure the request buffer context also has the HBA API flag set.
- Bluescreen during handling of NULL LuExt to complete a context reply.
  – Seen by OEM during Windows 7 testing.
  – Root cause: Port driver issues I/O's to devices with no LuExt allocated, causing bluescreen.
  – Fix: Check for NULL LuExt before processing the reply.

Version 1.27.01

Major Changes From Version 1.26.05:

General Changes

Functionality

- Added 'LargeSrbExt' registry entry to support 800 byte SRB extensions. This affects Fibre Channel controllers only.

Defect fixes

- Changed to return SRB_STATUS_INVALID_REQUEST instead of SRB_STATUS_ERROR if an invalid control code was detected for an IOCTL call.
  – Seen by OEM.
  – This issue is seen only with third-party drivers. To reproduce, use one of these third-party drivers. A driver before this fix will take a long time (20 minutes) to boot.
  – Root cause: Wrong error code is returned for an invalid IOCTL control code.
  – Fix: Return proper error code for an invalid IOCTL control code.
- MPI 1.7 adds new functionality to the DeviceInfo field. Previously, the driver used some of these bits for internal flags. A mask was added to the driver to account for possible interference with the new MPI functionality.

Version 1.26.05

Major Changes From Version 1.26.04:

General Changes

Functionality
- Added support for EEDP BYTCHK bit when set to 1 for Verify command (true data verification).

Defect fixes
- Previous version (1.26.03) was changed to utilize timer recovery routines (replacing all in-line code), to eliminate a system stall while doing an IOCTL hard reset. This fix did not work correctly when using the ImeVal.exe utility when doing write journal testing. The system would still freeze during an IOCTL hard reset and bluescreens would occasionally occur.
  - Seen by LSI test lab.
  - To reproduce, use the Win2k version of the driver. Use the ImeVal.exe utility to perform the Write Journal Test for an attached SAS IM volume. Mouse will not move during parts of the test and may cause blue screen.
  - Root cause: When an IOCTL diag reset was sent, the interrupt back from the HBA would be handled by the ISR and the diag reset would be performed in-line, bypassing the new functionality.
  - Fix: Checks were placed in the ISR to bypass the diag reset (in the ISR) if it came from an IOCTL command. The reset would then be handled using the timer routines.

Version 1.26.04

Major Changes From Version 1.26.03:

General Changes

Functionality

- Added StopOnDiagReset registry entry to cause bluescreen on a Diag Reset for debugging purposes.

Defect fixes

- Fixed failed recovery handshake for PCIe adapters.Fixed problem where PCIe adapters share interrupts with other PCIe parts and this can cause a diag reset to fail. If another device interrupts but we share that interrupt, and we're outside of our recovery routine (waiting the 1/10 second to come back in), our ISR will get called and it will read the doorbell, then it will clear the IntStatus register. This "swallows" the data that is being returned via the doorbell handshake and the PortEnable fails.
  - Seen by OEM.
  - To reproduce: 1) Have multiple 949E chips in an IBM server box. 2) Upgrade firmware and do a chip reset. 3) LSIUTIL gets "stuck" without exit.
  - Root Cause: PCIe adapters share interrupts with other PCIe parts which can cuase a diag reset to fail. If another device interrupts but on the shared interrupt, and driver is waiting the 1/10 second to come back for watchdog, ISR will get called and it will read the doorbell, then it will clear the IntStatus register. This "swallows" the data that is being returned via the doorbell handshake and the PortEnable fails.
  - Fix: Clear the Interrupt Status register only if IOC is not operational OR if the IOC is operational and doorbell interrupts are enabled.

- Changed to utilize timer recovery routines (replacing all in-line code), to eliminate a system stall while doing an IOCTL hard reset.
  – Seen by end customers, OEM customer, reproduced in LSI lab.
  – To reproduce download firmware to controller (using SASflash), or use LSIUtil, option 99.
  – Root cause: All operations for performing a hard reset and re-initialization of the controller were done in-line, causing the driver to hold the interrupt lock for the entire reset process.
  – Fix: Use a timer routine to perform the reset and re-initialization in small steps, leaving and returning to check status. This lets the driver release the interrupt lock and lets other processes execute during the reset and re-initialization.
- Added check for NULL SRB in CheckInqFlagReplies function. Previously, if NULL SRB was seen here this would cause a bluescreen.
  – Seen by OEM.
  – Can only reproduce when firmware sends bad reply.
  – Root cause: Firmware sends bad reply.
  – Fix: Check for NULL SRB, log error, and return from function without processing further.

Version 1.26.03

Major Changes From Version 1.26.02:

General Changes

Functionality

- None.

Defect fixes

- Changed OEM SAS SES PnP ID to remove F/W version.
  - Seen by OEM.
  - To reproduce: Update the OEM SAS SES device F/W to version 02.0. On a system install or reboot the New Hardware Found dialog box will be displayed. Select the option to allow the OS to search for a driver. No driver will be found and the SES device will have a yellow bang (!) in Device Manager.
  - Root Cause: The OEM SAS SES PnP ID in LSIPSEUD.INF contained the F/W revision.
  - Fix: Removed the last 4 characters (F/W revision) from the PnP ID.

Version 1.26.02

<u>Major Changes From Version 1.26.01:</u>

<u>General Changes</u>

Functionality
- Added OEM SAS Enclosure NoDriver entries to LSIPSEUD.INF files to suppress asking for drivers when these enclosures are present.

Defect fixes
- None.

Version 1.26.01

Major Changes From Version 1.25.11:

General Changes

Functionality

- Updated to MPI headers 1.05.17.
- Added support for detecting dual port SAS drives or detecting SATA drives behind a port selector.
- Changed handling of F/W diagnostic buffers such that when a diag reset is done a posted buffer is changed to be released and is available for the diag app to retrieve it.
- FC HBA API was changed to accommodate Vista during WHQL testing.

Defect fixes

- Provide workaround for DMA transfers using 1078 HBA on systems with more than 36 GB of memory.
  – Seen in OEM lab. Reproduce by writing data to at least 10 files at one time.
  – Symptom: System hangs during DMA transfers.
  – Root cause: DMA transfers while using 1078 HBA on systems with more than 36 GB of memory.
  – Fix: Walk through SGL to find problematic addresses. If bad address is found, use POM2 addressing instead of POM3 addressing. Bad addresses are in the ranges:
  0x00000009_00000000 through 0x00000009_1000FFFF
  0x00000009_18000000 through 0x00000009_18000FFF
  0x00000009_1C000000 through 0x00000009_1C000FFF

  This fix checks for the range:
  0x00000009_00000000 through 0x00000009_FFFFFFFF

- Fixed bus/target persistency mapping problem when using generic IOCTL calls or Task Management calls issued through the MPI IOCTL interface. This affects Fibre Channel controllers only

  – Seen in OEM lab.

  – Symptom: Wrong data will be returned when issuing generic IOCTL calls when host driver is in LUN persistency mapping mode.

  – Root cause: Host driver is not remapping bus/target addresses during these calls.

  – Fix: During generic IOCTL calls or task management commands issued through IOCTL interface, remap bus/target addresses if using LUN persistency mapping.

Version 1.25.11

Major Changes From Version 1.25.10:

General Changes

Functionality
- None. Changed version number to be in sync with LSIMPT.

Defect fixes
- None.

Version 1.25.10

<u>Major Changes From Version 1.25.09:</u>

<u>General Changes</u>

Functionality
- None.

Defect fixes
- Fixed looping for getting the NonCachedExtension
  - Detected via code review
  - Symptom: If registry settings are used to enable usage of F/W diagnostic buffers, and the requested buffer size allocations are too large for the OS to support, the Non-Cached Extension allocation will fail. The miniport should loop back and attempt the Non-Cached Extension allocation again to allow the system to boot, but due to a logic error the system will hang.
  - Root cause: A loop flag was not cleared properly on the 2nd allocation attempt.
  - Fix: Clear the loop flag at the start of the loop.
- Fixed FC HBA API for updated DTM tests
  - Seen in OEM labs, reproduced in LSI lab
  - To reproduce: Run the DTM WHQL tests for FC adapters. The FC HBA API test will fail.
  - Root cause: The FC HBA API test has been updated in the DTM WHQL tests. Additional test cases were added which expected the miniport to check SCSI command return data (the data buffer) for additional data lengths. Also, the SendCTPassThru test failed because a test script error treats the MaximumSize field in the CT_IU preamble as little endian instead of big endian.
  - Fix: Add code to ScsiReportLuns and ScsiInquiry tests to check return data buffers and detect additional data available. Added a

workaround for bad MaximumSize field to set the field to the maximum miniport buffer size.

Version 1.25.09

Major Changes From Version 1.25.08:

General Changes

Functionality
- None.

Defect fixes
- Fix possible misaligned data problem on IA64 machines.
  - Seen in Microsoft labs with LSI_SAS driver when using MSI/MSI-X only.
  - To reproduce (on LSI_SAS): On IA64 system, when CheckEnableMSIX is called causes a machine exception bugcheck. Problem does not happen on x86 or x64 systems because these systems can handle these misaligned accesses..
  - Root cause: When using MSI/MSI-X on IA64 system, data is misaligned.
  - Fix: Move PciConfigSpace array so that it is dword/quadword aligned. This was changed to avoid any future data misalignment issues if a dword pointer was used to access the array.

Version 1.25.08

Major Changes From Version 1.25.07:

General Changes

Functionality

- None.

Defect fixes

- Fixed reported status of an inactive volume to be 'Offline' (was reporting OK).

  – Seen by LSI test lab.

  – To reproduce: Roam an IR volume to a different HBA (causing the volume to be Inactive). Run the GetRaidConfig IOCTL. Note that the Inactive volume is show with an OK status.

  – Root cause: The check for an inactive volume was done after the check for an OK status and was not being executed.

  – Fix: Moved check for Inactive volume to first check of volume status.

-

Version 1.25.07

Major Changes From Version 1.25.06:

General Changes

Functionality

- Remove requirement for Port I/O access (mainly 1030 FWDLB), but don't enable via registry entry. (see Special Notes above)
- Remove SAS 6 PnP IDs for custom OEM branding from Windows 2000 (not supported by OEM)

Defect fixes

- Fixed FC HBA API SendCTPassthru DMA problem (handle non-contiguous data buffers).
  - Seen at Microsoft Server 2008 test labs, reproduced by LSI.
  - To reproduce, issue a Fibre Channel Common Transport message via the FC HBA API interface with a response buffer size that crosses a non-contiguous page boundary. The data will be DMA'ed into contiguous memory starting with the first page.
  - Root cause: The driver built only one SG entry each for the request and response buffers (assumed contiguous memory in IOCTL buffer).
  - Fix: The request buffer is copied to a local contiguous buffer. This is required since the MPI F/W interface for issuing a CT command allows only one SG entry for the request buffer. This buffer size is limited to 2048 bytes. Build a full SG list for the response buffer.

Version 1.25.06

Major Changes From Version 1.25.05:

General Changes

Functionality
- None.

Defect fixes
- Fixed GetConnectorInfo IOCTL (was not asking for next request).
  - Seen by OEM customer, reproduced by LSI
  - Using CSMI IOCTL test program issue a GetConnectorInfo IOCTL request twice (back-to-back). The system will hang on the second request and can potentially bluescreen.
  - Root cause: The processing of the GetConnectorInfo IOCTL did not ask for a NextRequest when the IOCTL was completed back to the OS.
  - Fix: Add call to get NextRequest at IOCTL completion.

Version 1.25.05

Major Changes From Version 1.25.04:

Version number change only to stay in sync with LSI_xxx (StorPort-based) drivers.

General Changes

Functionality
- None.

Defect fixes
- None.

Version 1.25.04

Major Changes From Version 1.25.03:

General Changes

Functionality
- Added SAS Broadcast Async Event support.

Defect fixes
- None.

Version 1.25.03

Major Changes From Version 1.25.02:

General Changes

Functionality
- None.

Defect fixes
- Inquiry commands issued to a particular OEM SAS tape drive that returned Check Condition status would have no Sense Data returned.
  – Seen in OEM customer lab, reproduced by LSI.
  – Issue a mal-formed Inquiry command (bad byte in CDB) to the OEM SAS tape drive and note that Check Condition status is returned by no Sense Data is returned.
  – Root cause: Inquiry commands to this OEM SAS tape drive go through special processing to enable SAS TLR. This routine was not detecting a bad SCSI status on an Inquiry command so the Sense Data was not transferred to the OS I/O structure.
  – Fix: Added a check for a bad SCSI status. If so, forward I/O processing to the SCSI IO error reply handling routine.
- Internal Task Management Target Reset changes.
  – Seen in drive manufacturer test lab.
  – During drive power cycle and hot plug testing, the F/W will issue a SAS Device Status Change Event with a Device Not Responding reason code. When the driver receives this event, it must issue a TM Target Reset to help clean up any outstanding I/Os. The drive manufacturer observed that in some cases a 2 second timeout on handshaking the TM message was too short (F/W could take up to 4 seconds to respond to the handshake.) Also, in some cases the TM reset reply was never received, resulting in OS I/O's being blocked forever.

- Fixes: 1) Increased reset message handshake timeout from 2 to 5 seconds. 2) Added setting of the ResetScsiTimer on these internal TM Target Resets. If a TM reply is not received within 15 seconds, the driver will hard reset the controller.

Version 1.25.02

Major Changes From Version 1.24.04:

Version numbers from 1.24.05 through 1.25.01 were skipped to account for the driver packaging restructuing.

General Changes

Functionality
- Send BusChangeDetected notifications if the chip is reset due to a F/W fault (IR device changes may go unscanned).
  – Seen by OEM customer, reproduced in LSI labs.
  – With certain SAS F/W, an IR volume creation will result in a F/W fault before the Volume Create event is issued to the host driver. The OS not seeing the newly created volume since a bus rescan was not performed.
  – Workaround: At any time the host driver detects a F/W fault and will hard reset the chip, issue a BusChangeDetected for all busses of the adapter. This will cause the OS to rescan each bus and any drive changes that occurred before the F/W fault will be detected.
- Restructured driver installation files to flat structures (one for each OS version/platform combination).

Defect fixes
- Add IR PhysDiskCreated/Deleted reason codes to BusChangeDetected notification test (detect change due to create/delete hot spare).
  – Seen by OEM customer, reproduced in LSI labs.
  – If a hot spare drive was created or deleted, the associated OS availability for the physical drive was not detected.
  – Fix: Added the PhysDiskCreated/Deleted reason codes to the check for when to issue a BusChangeDetected.

- Increase default number of reply frames to MaxDevices + 1 for SAS. (Fix starving of reply frames on removing big topologies.)
  – Seen by OEM customer, reproduced in LSI labs.
  – If more than 64 target devices are removed from the SAS topology in one action (such as removing a wide SAS connection from cascaded expanders/targets) the devices are never removed from the OS.
  – Root cause is that the F/W expects to be able to issue all device removal events before processing any EventAck commands. The driver would allocated only 64 reply frames by default.
  – Fix: Increase the default number of reply frames to MaxDevices plus 1 (but limited by the F/W ReplyQueue Depth).

Version 1.24.04

<u>Major Changes From Version 1.24.03:</u>

<u>General Changes</u>

Functionality

- None.

Defect fixes

- Fixed handling of SenseInfoBuffer addresses that are above the lowest 4GB for SCSI I/O's issued through SendIOCTLMessage and custom OEM IOCTLs.
  − Seen by end customers, OEM customer, reproduced in LSI lab.
  − To reproduce the following environment is needed:
    − System is running SYMMPI driver with a version number up to and including 1.24.03.
    − System is configured and is using more than 4GB of memory.
    − An application, agent, or service is calling the LSI proprietary MPI generic IOCTL interface.
    − The MPI messages being issued through the MPI generic IOCTL interface include SCSI IO or SCSI RAID Passthru IO.
    − The SCSI IO or SCSI RAID Passthru IO command receives a check condition and sense data from the target device
  − If all of the above conditions are met it is possible that the LSI driver will cause memory corruption. For this to occur, the IOCTL Srb DataBuffer must reside in physical memory that is above the lowest 4GB boundary.
  − Root cause: A logic error in the driver ignores the upper 32-bits of the physical address of the DataBuffer and will DMA the sense data to the truncated lower 32-bit address.

– Fix: A local buffer (always in the lowest 4GB of memory) is used to DMA sense data, then that buffer is copied to the IOCTL SrbDataBuffer.

Version 1.24.03

<u>Major Changes From Version 1.24.02:</u>

<u>General Changes</u>

Functionality

- Due to a F/W workaround, I/O's can be completed with an IO_CANCELLED_DUE_TO_R_ERR IOCLogInfo code for retry by the OS. This IOCLogInfo code was added to the list to be ignored, so an Event 11 would not be logged in the Windows Event Log.

Defect fixes

- None.

Version 1.24.02

Major Changes From Version 1.24.01:

General Changes

Functionality

- Updated INF and txtsetup.oem files with custom OEM 1068E branding entries.
- Added SAS x28 and x36 Expander NoDriver entries (to suppress asking for drivers when LSI SAS expanders are present).

Defect fixes

- Fixed check for other errors along with a data underrun on an Inquiry command in CheckInqFlagReplies.
  - Seen by LSI developer.
  - If a device returns a check condition on an Inquiry to a non-existent LUN (instead of returning Inquiry data), the LUN would show up in Device Manager as just "SCSI Device".
  - Root cause was treating a data underrun status with another error as a success status.
  - Fix: If another SCSI error status exists along with the data underrun, process the error status.
- Fixed adding of EEDP protection bits in BuildScsiIo32. MS port drivers still put LUN number in CDB byte 1, needs to be masked. (applies to Fibre Channel devices only)
  - Seen by LSI developer.
  - EEDP SCSI IO 32 message frames would not have the proper protection bits set for LUNs other than LUN 0.
  - Root cause is that Microsoft port drivers still insert the LUN value into byte 1 of the CDB. That is now where the protection bits reside.
  - Fix: Mask off the obsolute LUN field in byte 1 before OR'ing in the EEDP protection bits.

Version 1.24.01

Major Changes From Version 1.21.30:

Versions 1.22.xx and 1.23.xx were skipped to align with the LSIMPT driver.

General Changes

Functionality
- Added support for multiple hot spare drives. (Starting with SAS F/W 20.04 two hot spares are supported.)
- Modify test for hot spare (GetHotSpare) to check vendor field to determine drive type for a missing hot spare drive. (Allows missing hot spare drive to be displayed in GetRaidConfig even after a reboot of the system.)
- Add logging of Windows event for Fibre Channel if link speed changes (can be disabled by registry entry).

Defect fixes
- Fixed problem with GetRaidConfig options Device ID and Additional Data when volume is resyncing.
  – Discovered by LSI developer.
  – Symptom was running CSMITest, option 6, suboptions 2 or 3, output would display many lines of "RAID Config Number x", where x is incrementing.
  – Root cause was for suboptions 2 and 3 (when resyncing), GetRaidConfig was not filling in the bDriveCount element before returning the IOCTL.
  – Fix: Update bDriveCount before returning IOCTL.
- Added checks on DomainValidation code to protect from rougue replies (parallel SCSI only).
  – Seen in Microsoft labs.
  – Very intermittently, during ACPI Stress tests a reply would be seen that would have a Domain Validation flag bit set in the

message context, even though Domain Validation was not in progress. Result would be a system bluescreen
– Fix: Additional checks added to ensure that Domain Validation is actually in process before trying to dereference the DvSrb.

- For LsiWmiSendCTPassThru, adjusted minimum output buffer size to be 1052 bytes (12 bytes for parameters plus 16 bytes of preamble plus 1024 bytes output). Also, added checks for CT preample MaximumSize field.
  – Seen in Microsoft labs, reproduced by LSI
  – Using the FC HBA API, send a CT passthrough with a very small response buffer. With older LSI FC929 F/W the CT passthrough will never complete.
  – Fix: Increase minimum application buffer size for CT passthrough to 1052 bytes, to allow for a 1024 byte response buffer length. This also involved adding checks of the CT preamble to ensure the CT message is indicating the correct response buffer size.

Version 1.21.30

Major Changes From Version 1.21.29:

General Changes

Functionality
- None.

Defect fixes
- Fixed assignment of IOCsTotalDevices for SAS. Due to F/W mapping changes the PortSCSIID field must be used instead of MaxDevices
  – Seen by customer. Reproduced by LSI.
  – With SAS IT F/W version 10.14 or later, a pseudo device was not created for Windows 2000 or Windows XP (x86 or IA64). IOCTLs could not be sent to the HBA if no real devices were attached.
  – Root cause: Starting with SAS IT F/W 10.14 the mapping of devices was changed to reserve sections of target IDs, resulting in the PortSCSIID being a higher value than 127.
  – Fix: Change from using MaxDevices field to using PortSCSIID as the largest mapped value that can be returned by the F/W (for SAS only).
- Fixed sizing of SAS address table due to F/W mapping changes.
  – Discovered by developer code review.
  – SAS address table was hard coded to a size of 130 entries. With the changing of the F/W mapping (see above issue) target ID can go well above 130.
  – Fix: Size of the SAS address table is now calculated using the IOCsTotalDevices value.

Version 1.21.29

Major Changes From Version 1.21.28:

General Changes

Functionality
- None.

Defect fixes
- Fixed buffer length check in GetRaidConfig custom OEM IOCTL.
  - Seen by customer. Reproduced by LSI.
  - To reproduce, configure IS volume with 8 drives. Run dumpsas option /rc. System will return a -1 error code instead of displaying the RAID configuration information for the 8 drives.
  - Root cause: GetRaidConfig IOCTL was including the IoctlHeader in the minimum buffer length calculation, when it should not.
  - Fix: Removed IoctlHeader from the length calculation.

Version 1.21.28

Major Changes From Version 1.21.27:

General Changes

Functionality
- Changed driver installation amd64 folder names to x64. LSI was requested by OEM customer to change the folder names to the generic x64 designation, as this denotes both AMD64 and EM64T systems.

Defect fixes
- Fixed drive display error in GetRaidConfig custom OEM IOCTL.
  – Seen by customer. Reproduced by LSI.
  – To reproduce, configure IM volume in Optimal state with a hot spare drive. Run CSMI IOCTL test GetRaidConfig, option 1) Drives. System will display all 3 drives correctly the first time, but will display all zeros for the first drive on subsequent runs.
  – Root cause: GetRaidConfig IOCTL was using an index before it was initialized.
  – Fix: Moved initialization of index to location before it is used.
- Fixed display of RAID volume capacity for volumes of greater than 2 TB capacity.
  – Seen by customer. Reproduced by LSI.
  – To reproduce, configure RAID volume with > 2TB capacity. Issue GetRaidConfig IOCTL. Volume capacity will be incorrect (much smaller than actual).
  – Root cause: RAID volume capacity calculation was not using the MaxLBAhigh field of RAID Volume Page 0
  – Fix: Modified calculation to use MaxLBA and MaxLBAhigh fields.
- Fixed GetLocation IOCTL when a RAID volume physical disk is missing.

– Seen by customer. Reproduced by LSI.

– To reproduce, configure IM volume. Issue GetLocation IOCTL. Both physical disks will have location data. Remove one of the RAID physical disks. Issue GetLocation IOCTL. IOCTL will be returned with a Failed status.

– Root cause: A config page access for the missing physical disk was returning an error, causing the IOCTL to be returned with Failed status.

– Fix: Check physical disk status for a missing status. If missing, skip getting location data for this disk and go on to next disk.

Version 1.21.27

Major Changes From Version 1.21.26:

General Changes

Functionality

- None.

Defect fixes

- Fixed errors in GetRaidConfig custom OEM IOCTL.
  - Seen by customer. Reproduced by LSI.
  - To reproduce, configure IM volume in Optimal state. Remove one drive to put volume in Degraded state. Add a new drive to replace removed drive, volume will begin resyncing. Run CSMI IOCTL test GetRaidConfig, option 1) Drives. System will either bluescreen or data for many drives will be displayed
  - Root cause: GetRaidConfig IOCTL was using an uninitialized pointer. Also, callback for GetHotSpare routine was not getting called correctly.
  - Fix: Moved HpSasGetHotSpare callback check to top of list (so it is called before other callbacks). Fixed uninitialized RAID Vol Page 0 pointer in GetRaidConfig.

Version 1.21.26

Major Changes From Version 1.21.25:

General Changes

Functionality
- None.

Defect fixes
- Modify check for PortSCSIID > SYM_MAX_TARGETS. Log error only for parallel SCSI. For FC or SAS if check fails set initiator ID to 255.
  - Reported by LSI test lab. With SAS IT F/W that supports > 126 target devices, an Event 11 with offset 0x10 data of 0xAD010019 will be logged on every boot.
  - Check for PortSCSIID needs to be made only for parallel SCSI, where the Initiator ID is used in the bus protocol.
  - Fix is to log the event only for parallel SCSI. For SAS and FC, if the PortSCSIID is > 255, set the OS Initiator ID to 255.
- Fixed GetRaidConfig, GetLocation, and GetRaidElement IOCTLs to check for proper hot spare compatibility.
  - Reported by OEM customer. Information for hot spare disks for these IOCTLs was not being reported.
  - Fix is to use additional checks of hot spare data in MPI config pages to obtain hot spare info and verify compatibility with the IR volume.
- Fixed GetLocation IOCTL to return location info for non-IR SATA disks.
  - Reported by OEM customer. GetLocation data was not returned for a non-IR SATA drive.
  - Fix is to add the DeviceInfo mask for SATA_DEVICE to all checks for target devices in the driver internal SAS mapping table.

Version 1.21.25

Major Changes From Version 1.21.24:

General Changes

Functionality
- None.

Defect fixes
- Subtract 1 from NumberOfDiscoveredPorts in FillPortAttributes (FC HBA API).
  - Seen by OEM customer. Run FC HBA API application and observe that the NumberOfDiscoveredPorts for the LSI FC HBA is 1 more than what is expected.
  - Root cause is that LSI FC F/W logs into itself (the initiator) and the initiator is counted as a discovered port. FC HBA API applications don't expect that.
  - Fix is to subtract one from the number of discovered ports reported in FC Port Page 0.

Version 1.21.24

Major Changes From Version 1.21.23:

General Changes

Functionality

- INF File Update Only
- Add custom OEM INF entries for branded 1068E entries
- Update/add INF files for OEM SAS SES No Driver support.

Defect fixes

- None.

Version 1.21.23

Major Changes From Version 1.21.22:

General Changes

Functionality
- Added support for CSMI IOCTLs 0.90 (Phases 1 & 2).

Defect fixes
- Fixed FwDiagPostImmediateBuffers to read the reply frame for each post.
  - Seen by customer with their customized driver. If initial F/W Diag Buffers are enabled via the registry, subsequent messages would receive confusing reply frames.
  - Root cause was that the driver was posting initial diag buffers, but not reading the reply frames from those posts.
  - Fix was to add code in the initial buffer post routine to read the reply frames from the post messages.
- Added workaround for 106xE D3 to D0 transition issue (doorbell handshake would fail).
  - Seen everywhere when doing power management or driver upgrades with 106xE devices. Adapter would not be functional after a power transition or a driver upgrade.
  - Root cause is a H/W errata which leaves the doorbell reading Ready after a PCI D3 to D0 power state transistion. This allows the host driver to perform doorbell accesses before the hardware and firmware are actually Ready.
  - Workaround is to perform a diagnostic reset on the device during driver initialization (start-of-day) and when resuming from any power transition state.
- Added deferred processing of UpdatePersistencyTable if a reset is active. (Fibre Channel only)
  - Seen in LSI test lab. Heavy I/O stress testing with a large

number of LUNs. F/W would get 0x1314 fault codes.
– Root cause is that a persistency table update would be performed while a Task Management request was outstanding to the F/W.
– Fix is to check for an outstanding Task Management request, and if so, set a flag to defer the persistency table update until the TM request has completed.

- Fixed hibernation failure introduced in 1.21.22.
    – Found in LSI test lab. Hibernation would fail or hang.
    – Root cause is the default setting of host persistency active would cause the driver to try to allocate too much non-paged pool (NonCached Extension) and driver init would fail.
    – Fix is to not set host persistency on in crash dump mode.

Version 1.21.22

<u>Major Changes From Version 1.21.21:</u>

<u>General Changes</u>

Functionality
- Changed the default for Fibre Channel LUN Persistency from OFF to ON.

Defect fixes
- None.

Version 1.21.21

<u>Major Changes From Version 1.21.20:</u>

<u>General Changes</u>

Functionality
- None

Defect fixes
- Backed out CSMI 0.89 IOCTL changes due to improper header file structure definitions. Restoring CSMI 0.83 functionality.
  – Seen in LSI labs and reported by OEM (owner of CSMI header file).
  – Certain CSMI IOCTLs would fail due to application buffer length being too small.
  – New structure definitions caused compiler-generated padding, resulting in larger buffer sizes than implicit structure definition
  – Fix is to revert back to previous header file until fixed definitions are available.
- Backed out changes for ScsiRestartAdapter processing (added in 1.21.20) due to OEM/Microsoft not wanting this workaround. Instead, they went to the HD Audio card vendor for a fix.
- Added code to clean up an active IOCTL at end of Link Reset.
  – Seen in LSI lab.   On Fibre Channel adapters only, if Link Reset option is set in the registry, an active IOCTL may not be cleaned up after a Link Reset.

Version 1.21.20

<u>Major Changes From Version 1.21.19:</u>

<u>General Changes</u>

Functionality
- Changed GetRaidConfig IOCTL to get metadata size from IOC Page 6, if available.
- Changed GetConnectorInfo IOCTL to retrieve information from Manufacturing Page 7.
- Added fields to GetRaidInfo and GetRaidConfig IOCTLs. Added GetRaidFeatures and SetRaidControl IOCTLs.

Defect fixes
- Moved all ScsiRestartAdapter processing to a timer routine to avoid blocking interrupts while adapter is initializing.
  - Found in OEM lab, reproduced in LSI lab.
  - Interrupts are blocked during adapter initialization in ScsiRestartAdapter. If adapter initialization takes too long, it may cause other parts to function incorrectly.
  - Fix: Moved ScsiRestartAdapter processing to a timer routine.
- Added changes to handle I/O's left active after a bus reset.
  - Found in OEM lab.
  - Added routine to catch situations where firmware is not completing all I/Os on a bus reset, and changed method for issuing task management bus resets.
- Fixed erroneous access to RAID Physdisk Page 0 with IT firmware
  - Found in LSI test lab.
  - Request for RAID Physdisk Page 0 header without checking if the page was supported by the firmware resulted in an event in the Windows system event log.
  - Changed code to not request the RAID Physdisk Page 0 header if the current firmware is IT.

Version 1.21.19

Major Changes From Version 1.21.18:

General Changes

Functionality
- Removed support for 1066 and 1066E.

Defect fixes
- Added checks to not request IOC Pages 3-6, Raid Volume Pages 0-1, and Raid Phys Disk Pages 0-1 for IT firmware.
  - Seen by LSI developer.
  - Due to limited space, firmware has removed support for the above pages in IT firmware. Requesting these pages in non-IR firmware results in incorrect results for various IOCTLs.
  - Fix: Check IOC Page 2 capabilities flag for Raid support in the firmware. If Raid is not supported, do not request the unsupported pages.

Version 1.21.18

Major Changes From Version 1.21.17:

General Changes

Functionality
- None

Defect fixes
- Added checks to not request a rescan on certain SAS Device Status Change events.
  - Seen by LSI developer.
  - When certain SAS Device Change Events were received (such as ReasonCode INTERNAL_DEVICE_RESET, a rescan was being requested, but should not be. This results in unnecessary bus rescans.
  - Fix: Add checks to ensure that only ReasonCodes that require a bus rescan request one.

Version 1.21.17

Major Changes From Version 1.21.16:

General Changes

Functionality
- Added support for SAS nexus loss delay functionality.

Defect fixes
- Added check for NULL LuExt in CheckInqFlagReplies.
  - Seen in LSI test lab. Issue of Inquiry command which isn't a port driver enumeration request to a non-existent device would result in a bluescreen.
  - ScsiPortGetLuExtension was returning NULL since device didn't exist.
  - Fix is to check for NULL LuExt and exit routine to let ISR complete command.
- Added NextRequest notification to CleanUpAfterHardReset to prevent stalls of I/O's after a hard reset IOCTL call.
  - Seen in LSI test lab. Stress test running repeated hard reset IOCTLs would eventually result in no I/O's being issued to adapter.
  - Fix is to send a NextRequest notification after completing the hard reset IOCTL to ensure ScsiPort will issue I/Os.

Version 1.21.16

Major Changes From Version 1.21.15:

General Changes

Functionality

- None.

Defect fixes

- Fixed how the GetRaidConfig Ioctl retrieves a disk serial number.
  – Seen by customer. Significant characters are truncated from the drive serial number while trailing blanks are left alone.
  – Firmware changed the location of where it is storing the drive serial number.
  – Code was changed to retrieve the disk serial number from the new firmware location.
- Fixed STP/SSP passthrough calculation of minimum input buffer size.
  – Seen in LSI lab.
- Fixed clean up of task management when issued via Ioctl.
  – Seen in LSI lab. When issuing a target reset via an Ioctl, the LSI Windows driver hangs.
  – Added code to clean up the reset state when a task management completes.

Version 1.21.15

<u>Major Changes From Version 1.21.14:</u>

<u>General Changes</u>

Functionality
- None.

Defect fixes
- Added queuing of target resets.
  - Seen in LSI lab. With multiple drives including the boot drive attached to a HAB, and a stress test running to multiple non-boot drives, pulling the cable to the non-boot drives results in Windows hanging.
  - The driver was receiving target reset requests for all of the drives that were disconnected, but the driver was only sending down the first target reset it received.
  - Code was changed to queue target resets for later completion if a reset is already occurring.

Version 1.21.14

Major Changes From Version 1.21.13:

General Changes

Functionality

- Added OEM SAS tape device recognition and enabling of TLR.

Defect fixes

- Added SAS AEN notification when a SMART error is detected and an event is added to the MPI event queue.

Version 1.21.13

Major Changes From Version 1.21.12:

General Changes

Functionality

- Added ShutdownNotification to all INF files. Driver binary version change only.

Defect fixes

- None

Version 1.21.12

Major Changes From Version 1.21.10:

Version 1.21.11 was skipped to stay in sync with LSIMPT drivers.

General Changes

Functionality
- Added LSISAS1078 support.
- Added LSIFC949E support.
- Add SMART detection for all SAS devices, SEP message to light Predictive Fault LED for custom OEM SAS adapters only.

Defect fixes
- None

Version 1.21.10

Major Changes From Version 1.21.08:

Version 1.21.09 was skipped to stay in sync with LSIMPT drivers.

General Changes

Functionality
- None

Defect fixes
- Fixed GET_DEVICE_ADDRESS IOCTL (bug detecting IR phys disks). Mask off LogInfoValid bit when saving ConfigPageStatus.
  - Reported by OEM customer, reproduced by LSI developer.
  - IOCTL would return a SAS address for all target IDs after the last true existing target, instead of NoDeviceAddress error.
  - Fixed logic error in detecting IR physical disk and test of true IOCStatus for an invalid config page.

Version 1.21.08

Major Changes From Version 1.21.07:

General Changes

Functionality

- Changed default number of large sense buffers to 256 (for FC). This is to prevent throttling of I/O's when applications issue SCSI pass-thru commands with a sense buffer size larger than 18 bytes (OS default). This applies to Windows 2000 only.

Defect fixes

- Moved UpdatePersistencyTable from BringIocToOperationalState to HwInitialize (needs to run after EnableEvents, affects FC only).
  - Reported by LSI test lab during error injection tests.
  - Pending events could block use of doorbell handshake.
- Added option (via registry entry) to use a Link Reset in the place of a bus reset for FC (avoids dueling bus resets in multi-initiator configurations).
  - Reported by customer, reproduced by LSI. Applies to cluster configurations only.
  - Bus resets terminate all I/O's on a target for all initiators. This can result in another initiator timing out I/O's and issuing a bus reset, resulting in ping-ponging resets. A link reset terminates I/O's only for the issuing initiator.
- Added HWResetTimer for FC to avoid returning selection timeout if devices take a long time to log back in after controller reset.
  - Reported by LSI test lab during error injection tests.
  - Some fabric controllers require too long for devices to log in, therefore the initially report only a subset of devices after a controller reset occurs. This results in DeviceNotThere returns until all devices have logged in.

- Fix is to add a delay in returning DeviceNotThere status back to the OS until 30 seconds after a controller reset.

Version 1.21.07

Major Changes From Version 1.21.06:

General Changes

Functionality

- Remove debouncing of WMI AEN's being sent (requested by RSA).
  – StorLib was missing events when debouncing was used, but gets all events when debouncing is not used.

Defect fixes

- Added checks to CSMI GetRaidConfig IOCTL to not include a "foreign" (SAS vs. SATA) or "too small" hot spare drive as part of the volume.
  – Reported by OEM lab, reproduced by LSI.
  – To reproduce, create a SAS IR volume with hot spare, switch the hot spare drive with a SATA hot spare drive that is smaller than the IR volume physical disks. Run CSMITest GetRaidConfig IOCTL. Hot spare drive will appear in volume configuration.
  – Fix is to check hot spare drive type against volume drive type and check that hot spare size is adequate.

Version 1.21.06

Major Changes From Version 1.21.05:

General Changes

Functionality
- Added Phase III features to SAS_PHY_CONTROL IOCTL.

Defect fixes
- None

Version 1.21.05

Major Changes From Version 1.21.04:

General Changes

Functionality

- Added logging of device add/remove for LogExceptionEvents. (requested for debug purposes)

Defect fixes

- Changed GUID for EVENT_DATA_IR in ca_sas.mof and MEGASAS_EVENT in ca_wmi.c for StorLib WMI AEN support.
  – Found by RSA test lab. StorLib would get confused when both SCG and RSA SAS adapters were installed in a system, due to both drivers using the same GUID for the AEN Event structure.
  – Fix was to use a different GUID in the SYMMPI driver.
- Fixed input buffer length calculation in SSP and STP passthrus.
  – Found by LSI test lab. CSMItest program uses an exact buffer size for SSP and STP tests. Calculation was off by 1 byte due to a U8 data[1] element at the end of the IOCTL structure that wasn't accounted for.
- Fixed GetDeviceAddress and GetLocation IOCTLs.
  – Found by OEM lab. IOCTLs were using IR physical drives as if OS could see them. IR physical drives should not be reported with an address or location of their bus/target is passed in as inputs. For GetLocation, if the IR volume bus/target is passed in, locations are returned for all IR physical disks.
- Changed EventAck message to use EventNotify reply frame. Increased number of reply frames to number of request frames plus 16 (FC only).
  – Found by LSI test lab. These changes were needed to handle extreme conditions when many I/O's are terminated and EventAck's get stuck behind the I/O's on FC.

- Fixed saving of IoctlSrb in SMP Passthrough.
  – Found in OEM lab, reproduced in LSI lab. SMP passthrough commands would timeout, causing delays in system boot.
  – Problem was routine saving Srb for comparision before an element was modified, so comparison failed on completion and IOCTL was not completed back to OS.
  – Fix was to save Srb after element was modified.

Version 1.21.04

Major Changes From Version 1.21.03:

General Changes

Functionality

- None

Defect fixes

- Fixed SasGetScsiAddress for immediate completion. Added clear of CSMI IOCTL Srb for FW download and SMP passthru completions. Clear IOCTLActive flag for IOCTL TM requests.
  – OEM customer lab saw occasional IOCTL timeouts with PSuite when running Scripts that injected resets.
  – Above actions were to clean up tracking of outstanding IOCTLs.
- Set PortEnable timeout to 300 seconds for SAS (IT and IR F/W).
  – Seen by OEM customer with large SAS topology. Host driver would time out PortEnable after 30 seconds.
  – Debugging showed that large SAS topologies (many expanders and PHYs) can take significantly longer than 30 seconds to do SAS discovery (and receive a reply from PortEnable). Increased the PortEnable timeout to 300 seconds for all SAS F/W types.

Version 1.21.03

Major Changes From Version 1.21.02:

General Changes

Functionality

- For FC hard disk devices, force Inquiry data to support CmdQue, sync, and wide16 (some FC devices don't set these).

Defect fixes

- Modified SAS GetLocation IOCTL to check bus/target mapping and return error if not mapped.
  - Seen by LSI developer
  - Fix prevents request for SAS Device Page 0 for a non-existing page.
  - Return proper CSMI code of NO_DEVICE_ADDRESS.
- Modified check of RAID Phys Disk Change event to add a replaced physical disk SAS address to the cross-reference table.
  - Seen by OEM lab, reproduced by LSI lab.
  - Symptom: an IR physical drive which is replaced by a different physical drive will not display the new drive's SAS address in Get Raid Configuration and SSP Passthru's will not work.
  - Fix is to update the driver internal cross-reference table whenever a RAID Phys Disk Status Change event with a disk status of on-line is received.

Version 1.21.02

Major Changes From Version 1.21.01:

General Changes

Functionality
- Add custom OEM specific pseudo device support.

Defect fixes
- None

Version 1.21.01

<u>Major Changes From Version 1.20.20:</u>

<u>General Changes</u>

Functionality
- None

Defect fixes
- Added check for resync percentage reply with total blocks = 0.
  - Reported by OEM lab, reproduced by LSI lab
  - During IR hot plugging the RAID Action Indicator Struct reply can have Total Blocks = 0 (would cause a bluescreen due to a Divide-by-Zero error).
- Numerous changes for robust recovery if errors, hangs, faults occur while CSMI IOCTLs are active.
  - Reported by OEM lab and LSI lab
  - System hangs/bluescreens seen during IR hot plug testing.
  - Added multiple checks for outstanding IOCTLs during all error handling/recovery actions.
- Added read of IOC Page 1 to get slotNumber in ReadAndSetSASPages.
  - Reported by OEM lab, reproduced by LSI developer
  - Slot number was always being reported as 0. Will now be value populated in IOC Page 1 by LSI SAS BIOS.
- Disabled quiescing/unquiescing of IR physical disks when sending a SSP or STP passthrough IOCTL.
  - When quiescing for each passthrough, IR F/W exhibits occasional anomalies in replies (no reply to SSP [IOCTL hang], incorrect RAID Action reply). Seen as IOCTL hangs.
  - Analysis determined that quiescing was not necessary.
- Added code to populate SasAddressTable for new IR physical disk.

– Reported by OEM lab, reproduced by LSI developer

– If an IR physical disk is not present at system boot, then hot plugged later, a passthrough command to that disk will fail with an invalid parameter error.

– SAS Address of the newly added IR disk was not in the driver's internal SAS address to SCSI bus/target cross-reference table.

– Fix is to key off of the RAID IR Phys Disk Status Change event to populate the table.

- Fixed SSP Pasthrough bug which truncated returned data.

– Reported by OEM lab, reproduced by LSI developer

– Issue seen on 64-bit systems as last 4 bytes of expected data were not returned.

– Calculation for data length returned for entire IOCTL request was incorrect.

Version 1.20.20

Major Changes From Version 1.20.19:

General Changes

Functionality

- Updated to MPI headers 1.05.10.
- Added ScsiMaxLunLimit registry entry to allow user to override 32 LUN limit for parallel SCSI.
- Added support for 1064E and 1066E.
- Added new PCI Config adapter info page and WMI AEN support for SAS (StorLib support).
- Added custom OEM IOCTLs Get Location, Get Connector Info, and Phy Control IOCTLs.

Defect fixes

- Added blocking of new I/O's while any Task Mgmt request is outstanding (per MPI specification).

Version 1.20.19

Major Changes From Version 1.20.18:

General Changes

Functionality

- None

Defect fixes

- Fixed issue that prevented the driver from supporting EEDP in T10 mode.

Version 1.20.18

Major Changes From Version 1.20.17:

General Changes

Functionality

- None

Defect fixes

- Fixed issue with U320 event driven DV (was due to an OS bus scan at the same time). Qualified RAID volume event processing to be done only for SAS devices.
  – Seen in LSI test labs. System would bluescreen on insertion of a RAID volume physical disk (secondary or hot spare).

Version 1.20.17

Major Changes From Version 1.20.16:

General Changes

Functionality

- Added support for the SAS 8-port PCI-Express device (1068E).
- Removed SAS device support from Server 2003 drivers. SAS support for Server 2003 is provided by the LSI_SAS.SYS StorPort-based driver.
- Modified driver enabling of EEDP (End-to-End Data Protection) to follow the Fibre Channel F/W setting in IOC Page 1.
- Added a BusType registry entry to all INF files (Windows 2000 and above).

Defect fixes

- Fixed a bug in calculating Max SG elements for SCSI IO 32 (was getting a data underflow due to 96-byte U320 message frame size). This affected U320 adapters only.
  – Detected in Microsoft labs, verified by LSI developer.
  – Calculations would underflow, causing the driver to provide ScsiPort with very large bogus numbers for NumberOfPhysicalBreaks and SrbExtensionSize.
  – Symptoms observed included bluescreens and hangs, but not on all systems.
- Added Task Mgmt Target Reset after a SAS device not responding event received (required by F/W to complete the removal of device).
  – Detected in LSI labs. In some cases, when a SAS or SATA drive was hot removed, then another drive was plugged into the same PHY, the new drive was not detected by the F/W.

Version 1.20.16

Major Changes From Version 1.20.15:

General Changes

Functionality
- None

Defect fixes
- Fixed issue with overlapped commands during initial bus scan.
  – Seen in multiple LSI labs with Fibre Channel devices. Symptom was unusually long OS boot times and multiple Event 9's in the Event Log.
  – Root cause was a change in version 1.20.07 where the miniport ConfigInfo->MultipleRequestPerLu setting was changed from FALSE to TRUE. This was done to allow custom CSMI IOCTLs to not be throttled by outstanding I/Os for hard drive testing. However, setting MultipleRequestPerLu to TRUE could result in overlapped commands being issued.
  – Fix is to only enable MultipleRequestPerLu for SAS adapters (F/W handles queuing) and disable it for FC and parallel SCSI.

Version 1.20.15

Major Changes From Version 1.20.14:

General Changes

Functionality
- Added PERSISTENCY_TABLE_FULL to HandleEventNotification to detect full persistency table for SAS devices.
- Added capability to issue CSMI SSP and STP passthrough I/Os to IR physical disks.
- Added handling of IR volume events to allow OS to detect volumes arriving or leaving.

Defect fixes
- Fixed Standby problem, set flag to prevent WatchdogTimer routine from resetting controller when coming out of Standby.
  – Seen is LSI test lab. Some systems would not resume when coming out of Standby.
- Fixed problem with config page accesses in HwInitialize due to events being enabled already (could read improper header info).
  – Seen in LSI test lab. If events were enabled before the reading of config pages during driver init, an event being received during the reading of config pages could result in improper data being read for a config page header or config page data.
- For IOC_TERMINATED status and IOCLogInfo of aborted command, changed return status from BUSY to ERROR (OEM request, BUSY returns are retried by ScsiPort and aren't returned to the sending application).
- Fixed bug introduced when EEDP support was added (incorrectly building SG list for non-SCSI I/O messages).
  – Seen in developer testing. A non-SCSI I/O message that required a SG list would have that list built in the wrong place in the message.

Version 1.20.14

Major Changes From Version 1.20.13:

General Changes

Functionality
- Added 939X/949X 4GB FC support.
- Added End-to-End Data Protection (EEDP) support for 949X/939X Fibre Channel devices.

Defect fixes
- Added call to clean up allocated resources in SetupSenseBuffer if we run out of cached sense buffers.
  - Defect found in code review. Problem could occur only on NT/2K and only if an application was issuing SCSI pass through commands with a sense buffer size > 18 bytes.
- In CheckMpiIoctlReply, set default ReturnCode to 0.
  - Found in LSI developer testing. The addition of SAS IOCTLs resulted in the SIC->ReturnCode being set to non-zero for generic MPI IOCTLs. This impacted utilities that use the generic MPI IOCTL.
- Added Hot Spare physical disks to GetRaidConfig IOCTL.
  - The list of physical disks in a RAID volume will now include all hot spare physical disks.
- Modified Windows 2000 txtsetup.oem file to workaround a Windows 2000 RTM Setup bug and allow a single install package to work with both the RTM version and an integrated Service Pack 4 version.

Version 1.20.13

Major Changes From Version 1.20.12:

General Changes

Functionality

- For IOC_TERMINATED status and IOCLogInfo of aborted SAS command, changed return status from ABORTED to BUSY (OEM request). Also, place IOCLogInfo value into Srb->TimeOutValue field to allow determination of reason for abort by test application.
- Added HWFaultLimit capability to catch SRAM fault codes and block adapter resets after user defined count of faults. (For FC SRAM checking, enabled via HWFaultLimit registry entry.)

Defect fixes

- Check to make sure we don't overwrite an active timer, when we fail to reset the SCSI bus (ResetScsiTimer)
  - Bug in Watchdog Timer code added in 1.20.10. Can result in adapter hang due to ResetActive not being cleared.
- Use 5 second timeout on config page requests via doorbell handshake (vs. 30)
  - Reduces error recovery time if doorbell handshake fails.
- When updating persistence table, quit if any handshake fails.
  - OEM testing saw an adapter hang due to handshake fail.

Version 1.20.12

Major Changes From Version 1.20.11:

General Changes

Functionality

- On F/W download boot performed during StopAdapter, don't wait for F/W to return to Ready state, leave immediately. Also, increased timeout in getting to Ready state in FindAdapter from 10 to 20 seconds.

Defect fixes

- None

Version 1.20.11

Major Changes From Version 1.20.10:

General Changes

Functionality

- None

Defect fixes

- Fixed hibernation and crash dump for SAS/U320 (was accessing FC config pages).
  – Found in LSI test lab. The addition of the FC WMI HBA API functionality (in version 1.20.10) required adding reads of FC config pages. In crash dump (and hibernate) mode, the driver defaults to FC chip type. The fix is to qualify the FC config page reads so they aren't done when in crash dump mode.

Version 1.20.10

Major Changes From Version 1.20.09:

General Changes

Functionality

- Added a WatchdogTimer check for the firmware going non-operational (once per second). This will detect if the firmware faults must faster than waiting for an I/O timeout.
- Added support for WMI and HBAAPI for FC devices (required by Microsoft for all Fibre Channel drivers beginning with Server 2003 Service Pack 1).

Defect fixes

- Fixed UpdatePersistencyTable to try reading FCDevicePage0 using BUS_TID before invalidating a currently-valid entry; if that is successful, then leave the entry valid.
  - For FC devices only. Problem seen in OEM and LSI test labs. Occasionally, FC devices would be unmapped if they did not login fast enough after a controller hard reset.
- Fixed SAS IOCTL code that was returning improperly on errors.
  - Seen in developer testing. For some IOCTLs, if an IOCTL returned an error, the SRB was being completed twice (once in the low level routine and again in an upper dispatch routine). This did not cause a problem since both were done on the same execution pass in the miniport.
- Always clear CpqTaskType in LSImpiReset.
  - Some I/Os terminated by an OS issued Bus Reset could be returned with an Abort status vs. a Bus Reset status.
- Fixed update of SasAddressTable on add/remove events.
  - Hot added SAS devices could not be accessed properly by some SAS IOCTL commands.

Version 1.20.09

Major Changes From Version 1.20.08:

General Changes

Functionality
- None

Defect fixes
- Added a check of IOCLogInfo on SCSI_IOC_TERMINATED to determine whether an Abort or Reset was done, and return proper SrbStatus.

Version 1.20.08

Major Changes From Version 1.20.07:

General Changes

Functionality

- Added RAID support to GetControllerConfig.
- Enabled QueryTask Task Management request for SAS IOCTLs.
- Changed PseudoDeviceSupport to only use LSI PSEUDO DEVICE (requested by multiple OEM customers).

Defect fixes

- Zero out redundant ROM fields in GetControllerConfig.
- Added setting of bDataPresent in NonSCB_Func_handler for SAS TM replies.
- Fixed copy of response data for SSP passthroughs in CheckMpiIoctlReply.
- Fixed test for end device in GetScsiAddress.

Version 1.20.07

<u>Major Changes From Version 1.20.06:</u>

<u>General Changes</u>

Functionality
- Updated to CSMI .83 header.
- Added support for new bSSPStatus field to task management and SSP Passthru.
- Add a SAS_FW_CACHE_SIZE define and dynamically set the F/W cache size according to device type (SAS – 160K, others – 75K).

Defect fixes
- Change return status for CSMI clear task set and abort task set operations to SRB_STATUS_ABORTED.
  – Found in OEM testing. Injection of Bus Resets would result in I/O's being completed with Aborted status.
- Change the tracking mechanism for SRBs so that CSMI abort IOCTLs look for the SRB to abort in the IO tracking array. Also, include a member in the IO tracking array to store the IO path/target/lun/tag information so that the driver can match this information locally.
  – Found in OEM testing. Task Management Abort commands would only rarely be seen on the bus. Routine to determine if an I/O was still outstanding had a logic error that would not catch some outstanding I/Os.
- Set MultipleRequestPerLu to TRUE in HwInitializationData (this is suspected to fix an IO flushing issue).
  – This setting was set to FALSE at the suggestion of Microsoft to avoid a ScsiPort port driver bug which would reduce performance. This bug was fixed in Win2K SP3 and is not in XP or Server 2003. With MultipelRequestPerLu set to TRUE, ScsiPort does not wait for

all outstanding I/Os to complete before issuing an IOCTL to the miniport.

- Fixed GetPhyInfo IOCTL to correctly get device type.
  – Code that set the PHY connected device type was using the field as a bitfield instead of an enumerated value.

Version 1.20.06

Major Changes From Version 1.20.05:

General Changes

Functionality
- Added support for SAS 1068/1066 devices.
- Enabled support for ClearTaskSet in custom OEM IOCTLs

Defect fixes
- Set CpqTaskType in custom OEM IOCTL to ensure that the return value for an aborted IO is correct.
- Changed INF device description strings (shortened slightly, EMC PowerPath display problem)

Version 1.20.05

Major Changes From Version 1.20.04:

General Changes

Functionality
- None

Defect fixes
- Set PortEnable timeout according to F/W type (30 seconds for non-IR, 300 seconds for IR). Eliminate 5 minute delay on non-IR F/W if errors occur during PortEnable.

Version 1.20.04

Major Changes From Version 1.20.03:

General Changes

Functionality

- Added code to allow a Diagnostic Reset IOCTL to be executed even if the NoBoardReset flag is set.

Defect fixes

- Fixed custom OEM GetScsiBusData IOCTL to properly handle 64-bit data buffer address.
- Modified LsiMpiReset to not count unanswered resets (use timer instead)
  - With high numbers of outstanding I/O's, multiple resets are issued by the port driver. The unanswered reset count was going above the threashhold, causing a hard reset of the port. On dual-channel adapters, this could cause ping-ponging resets.
  - Observed in LSI lab, reproduced by issuing a hard reset with LsiUtil under I/O load on a dual-channel adapter.
  - Fix is to remove the unanswered reset count. This was added long ago to trap failed F/W. A timer has already been added which will detect failed F/W and will reset the port.
- Enable doorbell interrupts for all devices except 919X/929X
  - Previously, doorbell interrupts were disabled for all Fibre Channel devices. However, the 919X and 929X devices are the only ones with a H/W errata that can cause a PCI PERR if the IntStatus register is read.
  - Driver modified to use doorbell interrupts except for 919X/929X.
- Change generic MPI IOCTL to issue Task Management requests through the doorbell.
  - Task Management requests must be issued through the

doorbell.
– This change allows utilities to issue any type of Task Management request through the generic MPI IOCTL I/F.

- Fixed OEM custom GetDeviceData IOCTL to properly report U320 device speed.
  – Driver was reporting a device speed of async for U320 devices.
  – Table index in searching for speed value was incorrect.

- Set ReplySize and ConfigReplySpace to size of IOCFactsReply.
  – Size was hard coded to 64 bytes, then in 1.20.03 it was raised to 80 bytes for Fibre Channel and SAS devices.
  – To enable compatibility with future MPI changes to IOCFacts, the size for FC and SAS is now determined by the size of the IOCFactsReply in the MPI headers used to build the driver.
  – The size for 1020/1030 will remain 64 bytes for compatibility with older applications that have this size hard coded.

- Fixed processing of SASDeviceChangeStatus event to notify OS to scan the bus when a device is removed.
  – Observed in customer lab, OS detection of a removed SAS/SATA device did not occur when the device was removed, only when another device was installed.
  – Driver now issues a BusChangeDetected notification on both a removal and arrival of a SAS/SATA device.

- Added code to clean up all outstanding I/O's when the device is hard reset.
  – Observed hang condition with StorPort driver, long delays with ScsiPort driver. Reproduced by issuing a hard reset via LsiUtil while under I/O load.
  – If the port needs to be hard reset, outstanding I/O's will not be completed back to the OS, resulting in hangs or long timeouts.
  – Driver will now complete all outstanding I/O's back to the OS with a SCSI bus reset status 1 second after a hard reset has been done.

Version 1.20.03

Major Changes From Version 1.20.02:

General Changes

Functionality

- Added Serial Attached SCSI (SAS) device support.
- Added support for version MPI 1.5 interface.
- Added support for Firmware Diagnostic Buffers.
- Added support for custom OEM SAS IOCTLs.
- Added an additional OEM SSVID to enable setting the Initiator ID for U320 adapters at 255 (required for support of OEM SAF-TE device).

Defect fixes

- Fixed custom OEM GetScsiBusData IOCTL to properly handle 64-bit data buffer address.
- Changed default size of MPI reply frames to be 64 for all U320 devices and 80 for Fibre Channel and SAS devices.
  - Problem observed in LSI labs at in OEM customer testing.
  - Some system management applications hard coded the reply frame size to be 64 bytes. These versions are used only with U320 devices.
  - SAS devices require a reply frame size of 80 bytes due to the increased size of the IOCFacts reply.

Version 1.20.02

Major Changes From Version 1.20.01:

General Changes

Functionality

- None

Defect fixes

- Changed MPIOMode handling to work with multi-path filter drivers in both multiple path and single path modes.
  - Observed in LSI labs, Fibre Channel adapters in a multi-pathing configuration.
  - Failure mode: If paths have been degraded such that there is only a single path available to a device, then LIPs are done, the testing application will receive I/O errors due to devices being marked as missing.
  - Fix: For non-Microsoft MPIO multi-pathing drivers, do not enable the extended FCP status processing. This processing would return a Selection Timeout status due to the FCP status return on a LIP. This is MPIMode = 1. For the Microsoft MPIO driver, this Selection Timeout status is necessary, and MPIOMode = 2 should be used to enable FCP status.

Version 1.20.01

Major Changes From Version 1.10.00:

(Version label bumped from 1.10.00 to 1.20.01 for future SAS support activities.)

General Changes

Functionality
- Added support for Extended 64 processors (AMD64, IA32e).

Defect fixes
- Added more robust error handling of outstanding IOCTLs.
  - Observed by LSI during FC failover testing
  - System setup: Multiple FC adapters to multi-ported targets, heavy I/O stress tests, system management programs issuing IOCTLs, periodic FC cable breaks.
  - System may hang or bluescreen after 2 to 12 hours of testing. IOCTL was being completed twice back to the OS.
  - Added checks for IOCTL already completed and blocked issuing of IOCTLs during task management resets.
- Added timer to Task Management Reset reply.
  - Observed by MS test lab during PnP Stress testing.
  - System setup: 1030 on motherboard (flashless) with F/W version below 1.03.10. PnP Stress test running
  - F/W will very rarely die during I/Os or reset processing. If during a reset, there was no event to detect they F/W was dead.
  - Added timer to reset reply. If no reply is received within 15 seconds, the F/W is assumed to be dead, a H/W reset is done and the TM request is retried.
- Added U320 tape workaround (tape detect and IDP bit).
  - Due to 1030 H/W errata, a special In-line Data Padding mode is

required for both the 1030 and tape drive F/W to operate in U320 packetized mode.

– Driver detects tape drive at U320 speed and sets IDP bit for F/W.

- Modified 919X/929X MOST workaround for rev. 0 only.

  – MOST (Multiple Outstanding Split Transactions) set to 0 for 919X/929X rev.0 due to H/W errata.

  – Workaround modified to allow later revisions to use default MOST setting.

Version 1.10.00

Major Changes From Version 1.09.90:

(GCA label version change only from 1.09.90)

General Changes

Functionality
- None

Defect fixes
- None

Version 1.09.90

Major Changes From Version 1.09.14:

(Version was bumped to 1.09.90 for 1020A test cycle.)

General Changes

Functionality

- None

Defect fixes

- Modify routine to enable the WriteSeq register for F/W Download Boot.
  - Seen in OEM customer integration test, verified by LSI
  - System with 1020A in a flashless environment would fail any Windows power management event (which performs a F/W download boot). System would hang on shutting down from Standby.
  - System would not resume from Standby (blank screen), had to be cold power cycled.
  - 1020A behaves slightly different from previous MPI devices. Diagnostics register is left enabled even after a H/W reset via the Diagnostics register. Fix is to write a "junk" byte to the WriteSeq register to ensure that the Diagnostics register is disabled, then write the magic sequence to enable the Diagnostics register.

Version 1.09.14

Major Changes From Version 1.09.12:

(Version 1.09.13 was used for other functionality.)

General Changes

Functionality
- Added support for Fibre Channel InBand Management.
- Added driver based Fibre Channel bus/target persistency (registry enabled).
- Added read of OEM I2C device in custom OEM IOCTL.

Defect fixes
- Limit number of reply frames requested via a regsitry entry to not overrun the maximum supported by the F/W.
  - Seen only in LSI test lab.
  - If user specified a value for NumberOfReplyBuffers registry entry that was larger than the F/W maximum the F/W will hang when the host driver seeds the ReplyFIFO and the adapter will fail.
  - Results in non-functioning adapter or a 0x7B bluescreen if the adapter is the boot device.
  - Fix is the compare request value against F/W maximum and limit to F/W maximum.
- Fixed IOCTL completion processing with ResetActive (was accessing the reply frame even with a turbo reply).
  - Seen only in LSI test lab.
  - System must be running MyStorage (or similar CIM program) that issues IOCTLs periodically, running an I/O stress test, and performing cable breaks or causing similar exceptions.
  - During bus reset processing, an IOCTL can complete with a

context reply. IOCTL clean-up code when in reset processing mode was accessing a reply frame even with a context reply (no valid reply frame).

– System will bluescreen due to invalid memory access.

– Fix is to check for a context reply before accessing reply frame.

Version 1.09.12

Major Changes From Version 1.09.11:

General Changes

Functionality
- Reversed change to LSImpiReset in 1.09.07. Microsoft decided to fix the Driver Verifier test and not have us poll in the reset routine.
- Removed pseudo device support for Server 2003 and above (a new registry entry provides a LUN at the initiator ID).

Defect fixes
- Added error recovery to HwInitialize when EnableEvents or ReadPortPages fails.
  - – Seen in Microsoft lab during PnP Stress testing.
  - – F/W initializes but fails during config page accesses.
  - – HwInitialize routine will perform a H/W reset and retry if F/W fails.
- Changed initial handshake message timeout to just 2 seconds for a reset message (to detect hung F/W faster).
  - – Faster detection of failed F/W for above error recovery.
- Added 100 usec delay after setting ResetAdapter bit during F/W download boot (fixed issue where F/W would not get restarted).
  - – Seen on one OEM platform.
  - – Symptom: Channel not functional after F/W download boot, driver upgrade, or hang during Windows Setup.
  - – Need to delay before reading diagnostic register after setting ResetAdapter bit.
- Fixed ScsiPortNotification(NextLuRequest) call for IOCTL calls when processing OtherChannelReset (need to specify path/ID/LUN).
  - – Seen when system management program would issue an IOCTL

immediately after doing a H/W reset or F/W update.

– Symptom: IOCTL will not complete and will time out, resulting in an Event 9 in the Event Log.

– NextLuRequest was passing wrong number of parameters when returning a Busy status while F/W was reinitializing.

- Fixed custom OEM SCSI Pass Thru IOCTL to use a cached sense info buffer (failed on IA64 systems, application buffer can have a physical address > 4GB).

Version 1.09.11

Major Changes From Version 1.09.10:

General Changes

Functionality
- None

Defect fixes
- Changed workaround for Multiple Outstanding Split Transactions (MOST) setting from 1 to 0 for 919X and 929X devices.
  – Under heavy I/O load data corruption was observed during customer testing and verified in LSI labs.
  – System and setup to reproduce: FC919X or FC929X host adapters driving heavy I/O load to multiple FC devices.
  – Original description of bug symptoms and behavior prior to change: Data corruption was observed.
  – Fix: Change workaround to set MOST value to 0 (at most only one outstanding split transaction).

Version 1.09.10

<u>Major Changes From Version 1.09.09</u>:

<u>General Changes</u>

Functionality
- None

Defect fixes
- Added back in the workaround for too many "errors" being returned on FC cable pulls/failovers. Active for FC devices only, not dependent on MPIO mode.
  - Issue seen by FC customers when doing failover testing.
  - System and setup to reproduce issue: System with FC adapters and devices with multiple paths and failover capability, or just testing FC cable pulls.
  - Original description of bug symptoms and behavior prior to change: Failover activity results in no I/Os being issued to one or more FC devices. (This is a ScsiPort bug on how it handles multiple error returns from devices.)
  - Fix: On an IOCSTATUS_SCSI_EXT_TERMINATED error return, return Busy instead of Bus Reset status (Busy is not considered an error by ScsiPort).
- Added unmasking of interrupts in LSImpiIsr when processing a doorbell interrupt (fix hang during MS PnPstress testing).
  - Issue seen in Microsoft lab
  - System and setup to reproduce issue: System with flashless 1030 device, running MS PnPStress test.
  - Original description of bug symptoms and behavior prior to change: During PnPStress test (constant disable/enable of adapters) an adapter would "hang" and not re-enable.
  - Fix: Doorbell interrupt processing was leaving interrupts disabled in some timing situations. Added code to insure

interrupts are re-enabled.
- Microsoft RAID# 817517
- Added test for F/W Ready state in HandleAppRequest (need to bring IOC back to Operational state if other channel did a diagnostic reset).
  - Issue seen by LSI CIM developer
  - System and setup to reproduce issue: System running MyStorage app with FC adapters, downloading new F/W (requires a diagnostic reset).
  - Original description of bug symptoms and behavior prior to change: After download of F/W (and diagnostic reset), an IOCTL to the other channel would timeout (due to F/W being reset to Ready state, not Operational).
  - Fix: Added test of F/W state in HandleAppRequest (IOCTL processing). If in Ready state, driver brings IOC to Operational state before processing IOCTL.
- Qualified doorbell interrupts/processing for parallel SCSI only (to avoid a 929X errata when reading IntStatus).
  - Issue seen by OEM customer (documented as a 929X errata).
  - System and setup to reproduce issue: System with 929X running heavy I/O load, fails very infrequently. Symptom dependent on how system handles a PCI PERR.
  - Original description of bug symptoms and behavior prior to change: 929X errata Item#1, read of IntStatus register can cause a PERR due to timing anomaly in 929X. Driver was reading IntStatus to determine if a Doorbell register interrupt occurred.
  - Fix: Doorbell interrupts were enabled to assist in 1030 flashless power management handling. FC devices don't need doorbell interrupts enabled, so they have been disabled for FC devices.

Version 1.09.09

Major Changes From Version 1.09.08:

General Changes

Functionality

- Modified INF files to enable pseudo device for Fibre Channel, set MaximumSGList to 0xFF (1 MB maximum I/O size), NumberOfRequests to 0xFF (255 maximum concurrent requests). These changes allow a single INF file to have the default settings for both Fibre Channel and parallel SCSI adapters. (LSI internal request)

Defect fixes

- Moved setting of defaults for IntCoalTimeout and IntCoalDepth so they are set even with NULL ArgumentString.
  – Issue identified by developer (code review)
  – System and setup to reproduce issue: Not applicable
  – Original description of bug symptoms and behavior prior to change: Not applicable (no symptoms)
  – If no DriverParameters string specified, interrupt coalescing would be disabled. LSI standard installation files always specify a DriverParameters string so this defect will not be seen.
- Added checks for NULL pointers on all cached MPI config pages.
  – Issue identified by Microsoft storage labs
  – System and setup to reproduce issue: Occurred on only one Tiger4 system during PnP Stress test. After running many hours, one of the adapter initialization sequences would work properly except the first access to a config page would fail. The driver attempts no more accesses after a failure and has NULL pointers for cached config pages.
  – Microsoft RAID# 812606 and 814878

Version 1.09.08

Major Changes From Version 1.09.07:

General Changes

Functionality

- None

Defect fixes

- Changed AdapterRestart routine to perform Domain Validation on parallel SCSI. (ensures externally powered U320 drives run at U320 speed on Resume from Standby)

Version 1.09.07

Major Changes From Version 1.09.06:

General Changes

Functionality

- When IOCSTATUS_BUSY is returned, check for timeout value of 36000 (issued by custom OEM test program). If so, change timeout to 35999. (Allows test program to detect that this was IOCSTATUS_BUSY.)
- Check SCSI Port Page 2 DV bits to either disable DV or limit DV to Inquiry only checks.
- Changed LSImpiReset routine to always poll on ISR until reset reply is received. (Required by Microsoft since Driver Verifier expects all I/O's to be completed before returning from Reset routine.)

Defect fixes

- Fixed MOST workaround for 919X/929X (workaround was setting MOST to a bitvalue of 0 instead of 1). [resulted in lower write performance on 919X/929X]

Version 1.09.06

Major Changes From Version 1.09.05:

General Changes

Functionality

- Added 1035 support to driver installation (INF) files.

Defect fixes

- Added I/O throttling per device to avoid overrunning device queue depths (eliminate multiple Queue Full returns and Event 11/15 errors in event log).

Version 1.09.05

Major Changes From Version 1.09.04:

General Changes

Functionality
- None.

Defect fixes
- Changed previous fix for crash dump & hibernate to make it more robust (was still failing when the resyncing IR volume was the boot device). Moved fix into LSImpiReset.

Version 1.09.04

Major Changes From Version 1.09.03:

General Changes

Functionality

- None.

Defect fixes

- During crash dump (& hibernate), added check in HwInitialize for reply from task management reset message. (Need to wait for reply to prevent Busy status return on Inquiry.) Fixed hibernation failure when an IR volume is resyncing.

Version 1.09.03

Major Changes From Version 1.09.02:

General Changes

Functionality

- Qualified the NoBoardReset flag to be active only for Fibre Channel adapters. (Some Fibre Channel customers need NoBoardReset active, but the driver must be able to do a board reset when a flashless 1020/1030 is also in the system.)

Defect fixes

- Fixed problem with access to Manufacturing Page 0 if no NVDATA available (would result in a bluescreen if valid NVDATA was not available).
- Added check for valid event data on EventNotificationReply (Fibre Channel adapters log link CRC errors using an Event Notification Reply with no event data, just LogInfo).
- Added MsgUnitReset to StopAdapter routine. Need to disable Event Notifications before disabling interrupts (Fibre Channel devices can try to send events during adapter disable, hanging up the F/W).
- Fixed data buffer address calculation for MPI pass through writes on 64-bit platforms (would cause improper data to be written on MPI IOCTL calls).

Version 1.09.02

Major Changes From Version 1.09.01:

General Changes

Functionality

- None.

Defect fixes

- Fixed doorbell timeouts to give Port Enable a 5 minute timeout.

Version 1.09.01

Major Changes From Version 1.08.23:

- Version 1.08.23 was a GCA build so the version number for this build was set to 1.09.01.

General Changes

Functionality
- Added support for 1035 device (MOST workaround not enabled).
- Added pseudo device support for FC devices (default disabled).
- Reworked MOST workaround logic to not do workaround for 1030 C.0

Defect fixes
- Changed I/O completion while reset active to complete I/Os individually (vs. using CompleteRequest to complete all at once) to workaround bugs in FC multi-path filter drivers.
- Added MPI_FCPORTPAGE1_FLAGS_VERBOSE_RESCAN_EVENTS to enabling of MPIO capability (needed since other OEMs used other 2 flags).
- Fixed virtual buffer address calculation for 64-bit platforms in SendIOCTLMessage.
- Added reread of IOC Page 3 on any RAID event DV or after a diagnostic reset.
- Fixed problem with accessing SCSIPortPage0 on FC device.

Version 1.08.23

<u>Major Changes From Version 1.08.22</u>:

<u>General Changes</u>

Functionality

- Added capability to put Initiator ID at 255 for certain OEM devices (allows support for custom OEM system processor at initiator ID).

Defect fixes

- None.

Version 1.08.22

Major Changes From Version 1.08.21:

General Changes

Functionality
- Backed out workaround for bad data direction on commands. Did not work properly and 3rd party application will be fixed to use the correct data direction on the Mode Select command.

Defect fixes
- None.

Version 1.08.21

Major Changes From Version 1.08.20:

General Changes

Functionality

- Added workaround for bad data direction on commands. The INVALID_SGL error will cause a retry of the command with the CMD_DETERMINES_DATA_DIR flag set in the MsgFlags field.

Defect fixes

- Added custom OEM IOCTL GetResetId support for new F/W functionality that reports the ID of the target on the bus in SCSI Port Page 0.

Version 1.08.20

Major Changes From Version 1.08.19:

General Changes

Functionality

- Changed StartExecSCB to use always issue tagged commands to devices that support tagged if custom OEM CIM support enabled.

Defect fixes

- Added code to handle custom OEM Abort or BDR IOCTL hang due to SCSI bus hang (issue MsgUnitReset to clear everything and reset the bus).
- Fixed check of ImageSize for F/W download boot in MPI_DIAG_RESET.
- Fixed periodic hang/reset when custom OEM Storage Agents running (was not returning reply frame on non-turbo custom OEM SCSI pass-thru).

Version 1.08.19

Major Changes From Version 1.08.18:

General Changes

Functionality
- None

Defect fixes
- Added handling for getting a check condition on an OS issued request sense command (no SenseInfoBuffer).
- Change custom OEM IOCTL devPhysWidth back to an enumerated value, set on a device basis per Inquiry data.

Version 1.08.18

For change history prior to version 1.08.18, see the Release Notes for a version prior to 1.20.10.

NOTES, ISSUES and DETAILS

1. Known issues and restrictions

- The BIOS Config Utility settings for sync, wide, and adapter SCSI ID are the only ones honored by the Windows driver. Other settings are overridden by the Windows OS.
- IOCTL calls cannot be issued to an adapter that has no visible devices on the bus (Windows OS restriction).
- IOCTL calls are not supported on Windows 9X_ME.
- For full driver upgrade functionality in flashless environments, Windows drivers at revision 1.08.12 and above and 1030 F/W at revision 1.00.14.00 or above should be used.
- LSI Pseudo Device support is enabled (via a registry entry) by default on Fibre Channel devices, starting with version 1.09.09. (Not enabled for Server 2003. A system registry entry provides the same function.)
- When using 1030 IR F/W version 1.03.09 or later, driver version 1.09.05 or later must be used for hibernation and crash dump to be supported properly.
- Driver versions from 1.09.07 through 1.09.10 have the PCI-X Multiple Outstanding Split Transactions setting in config space set to 1 for 919X and 929X. This can result in data corruption. Fixed in 1.09.11.
- Driver version 1.09.15 is required for proper operation with MyStorage.
- Driver version 1.09.90 or above is required for 1020A flashless environments.
- Multiple adapters of the same type should all have the adapter BIOS enabled if one of the adapters controls the boot device.
- To enable End-to-End Data Protection (EEDP) support for the 949X/939X devices, the registry entry "EEDP_T10_Enable=1;" must be added to the DriverParameter string.

- Beginning with version 1.20.17, Serial Attached SCSI (SAS) device support for Server 2003 is supplied only by the LSI_SAS.SYS StorPort-based driver.
- For the Windows 2000 and Windows Server 2003 operating systems a hotfix is required for proper support of drive hot plug operations. This is described in Knowledge Base article 867818. More information can be found at: http://support.microsoft.com/default.aspx?scid=kb;en-us;867818.
- Versions of SYMMPI at or before 1.24.03 can cause memory corruption under a very specific system environment. See the "Defects fixes" section under version 1.24.04