

Exhibit A1

EXHIBIT A1

Invalidity Claim Chart for U.S. Patent No. 8,891,298 Based on U.S. Patent No. US 8,634,240 to Gavens et al. (“Gavens”)

Claim Language	Citations to Prior Art
[1pre] A system for storing data comprising:	<p>Gavens discloses a system for storing data.</p> <p>For example, Gavens discloses a flash memory device for storing data.</p> <p>Fig. 1 ref. 90, col. 8:15-16: The host 80 sends data to be stored at the flash memory device 90.</p>
[1a] at least one MLC non-volatile memory module comprising a plurality of individually erasable blocks;	<p>Gavens discloses at least one MLC non-volatile memory module comprising a plurality of individually erasable blocks.</p> <p>For example, Gavens discloses non-volatile memory including a second portion of MLC memory that stores multiple bits per cell and is divided into blocks of cells that are erased together.</p> <p>Col. 1:18: Non-volatile memory.</p> <p>Col. 4:32-34, 17:21-23: The first portion has each memory cell storing one bit of data and the second portion has each memory cell storing more than one bit of data.</p> <p>Col. 5:15-17, 50-53, 16:22-24, 20:31-34: The high density storage portion (D3) has each memory cell storing 3 bits of data. The low density storage portion (D1) has each memory cell storing 1 bit of data.</p> <p>Fig. 14A, col. 16:15-22: The array of memory cells 200 is partitioned into a first portion 410 of lower density and a second portion 420 of high density having each cell storing multiple bits.</p>

Claim Language	Citations to Prior Art
	<p>Abs., col. 6:18-21, 15:62-66, 16:16-22: The memory has a first portion operating with a lower density and a second portion operating with a higher density.</p> <p>Fig. 8, col. 12:11-12, 31-33: A multi-state memory or multi-level cell (MLC) memory has a population of cells with each cell in one of eight possible states.</p> <p>Fig. 15 ref. 500, col. 17:27-29: Configuring memory into a first portion having memory cells operating with a margin of error larger than that of a second portion.</p> <p>Fig. 6, col. 11:9-10: A memory array organized in erasable blocks.</p> <p>Col. 11:4-8: The array of memory cells is divided into blocks of memory cells. Each block contains the minimum number of memory cells that are erased together.</p> <p>Col. 8:19-21: The memory chip 100 includes a memory array 200 of memory cells with each cell capable of being configured as a multi-level cell (MLC) for storing multiple bits of data.</p>
<p>[1b] at least one SLC non-volatile memory module comprising a plurality of individually erasable blocks; and</p>	<p>Gavens discloses 2-state use of a partition of the MLC module comprising a plurality of individually erasable blocks which obviously may be substituted with an SLC module comprising a plurality of individually erasable blocks. (See Sutardja, Moshayedi and Lee '794 under element [1g] below).</p> <p>For example, Gavens discloses non-volatile memory including a first portion of MLC memory sometimes referred to in the patent as "SLC," that is configured to store one bit per cell and is divided into blocks of cells that are erased together.</p> <p>Col. 1:18: Non-volatile memory.</p> <p>Col. 4:32-34, 17:21-23: The first portion has each memory cell storing one bit of data and the second portion has each memory cell storing more than one bit of data.</p>

Claim Language	Citations to Prior Art
	<p>Col. 5:15-17, 50-53, 16:22-24, 20:31-34: The high density storage portion (D3) has each memory cell storing 3 bits of data. The low density storage portion (D1) has each memory cell storing 1 bit of data.</p> <p>Fig. 14A, col. 16:15-22: The array of memory cells 200 is partitioned into a first portion 410 of lower density and a second portion 420 of high density having each cell storing multiple bits.</p> <p>Abs., col. 6:18-21, 15:62-66, 16:16-22: The memory has a first portion operating with lower density storage and a second portion operating with a higher density.</p> <p>Fig. 7, col. 11:55-56, 12:6-7: A binary memory or single-level cell (SLC) memory has a population of cells with each cell in one of two possible states.</p> <p>Fig. 15 ref. 500, col. 17:27-29: Configuring memory into a first portion having memory cells operating with a margin of error larger than that of a second portion.</p> <p>Fig. 6, col. 11:9-10: A memory array organized in erasable blocks.</p> <p>Col. 11:4-8: The array of memory cells is divided into blocks of memory cells. Each block contains the minimum number of memory cells that are erased together.</p>
<p>[1c] a controller coupled to the at least one MLC non-volatile memory module and the at least one SLC non-volatile memory module wherein the controller is adapted to:</p>	<p>Gavens discloses a controller coupled to the at least one MLC non-volatile memory module and the at least one SLC non-volatile memory module (as substituted).</p> <p>For example, Gavens discloses a controller that is coupled to a memory chip that includes a memory array that includes the second portion and the first portion.</p> <p>Fig. 1 ref. 102, fig. 14A, col. 8:17-21, 8:32-35, 16:14-24: The controller 102 manages one or more memory chips 100 in the memory device 90. The memory chips 100 includes a memory array 200. The memory array 200 includes the first portion 410 and the second portion 420. The controller 102 manages higher-level memory operations, for example, read and write.</p>

Claim Language	Citations to Prior Art
	Col. 2:56-58: Controller performs intelligent and higher level memory operations and interfacing.
<p>[1d] maintain an address map of at least one of the MLC and SLC non-volatile memory modules, the address map comprising a list of logical address ranges accessible by a computer system, the list of logical address ranges having a minimum quanta of addresses, wherein each entry in the list of logical address ranges maps to a similar range of physical addresses within either the at least one SLC non-volatile memory module or within the at least one MLC non-volatile memory module;</p>	<p>Gavens discloses the controller is adapted to maintain an address map of at least one of the MLC and SLC (as substituted) non-volatile memory modules, the address map comprising a list of logical address ranges accessible by a computer system, the list of logical address ranges having a minimum quanta of addresses, wherein each entry in the list of logical address ranges maps to a similar range of physical addresses within either the at least one SLC non-volatile memory module (as substituted) or within the at least one MLC non-volatile memory module.</p> <p>For example, Gavens discloses that the controller implements a memory block management system that maintains a directory of ranges of logical addresses, which are accessible by a host and have a minimum quantum of addresses corresponding to a block or a page, mapped to similar ranges of physical addresses in the first portion or the second portion of the memory array.</p> <p>Col. 4:64-67, 8:30-39, 17:17-20, 53-56, 18:47-52, 63-67: The host 80 sends data to be written to the memory array 200 in logical sectors. A memory block management system implemented in the controller 102 maps the logical sectors to the physical structure of the memory array 200.</p> <p>Fig. 15 ref. 550, col. 17:39-40: Identifying the last written copy as valid data for subsequent read.</p> <p>Col. 11:19-22: The group of cells erasable together consists of one or more addressable erase units. The erase unit or a block stores one or more pages of data. A page is a minimum unit of programming and reading.</p> <p>Col. 8:40-43: The entire disclosure of Paley US2010/0172180 is incorporated by reference. Paley US2010/0172180 para. 155: The memory manager 300 in the controller 100 manages the storage and retrieval of host logical sectors of data among metablocks of the flash memory 200. The memory manager</p>

Claim Language	Citations to Prior Art
	<p>300 manages erase, read, and write operations of the metablocks. The memory manager 300 maintains system control data and directory data associated with its operations.</p>
<p>[1e] determine if a range of addresses listed by an entry and mapped to a similar range of physical addresses within the at least one MLC non-volatile memory module, fails a data integrity test, and, in the event of such a failure, the controller remaps the entry to the next available equivalent range of physical addresses within the at least one SLC non-volatile memory module;</p>	<p>Gavens discloses the controller is adapted to determine if a range of addresses listed by an entry and mapped to a similar range of physical addresses within the at least one MLC non-volatile memory module, fails a data integrity test, and, in the event of such a failure, the controller remaps the entry to the next available equivalent range of physical addresses within the at least one SLC non-volatile memory module (as substituted).</p> <p>For example, Gavens discloses that the controller determines if a range of logical addresses of data mapped to a similar range of physical addresses within the second portion fails a data integrity test, and in the event of such a failure, remaps the range of logical addresses to an equivalent range of physical addresses within the first portion.</p> <p>Col. 4:51-54, 15:66-16:3, 17:9-11, 21:35-36: A first copy of the incoming data is written to the second portion, and the first copy in the second portion is read back to check for excessive error bits. If the error bits exceed a predetermined amount, then a second copy of the incoming data is written to the first portion.</p> <p>Col. 5:1-7: After writing to a high density storage portion, a post-write read checks if the error rate is within a prescribed limit. If post-write read checks results in an error rate above the prescribed limit, then the data is rewritten to another, less error-prone portion that has each memory cell storing less data bits per cell.</p> <p>Abs., col. 6:22-25: After a copy of data is written to the second portion, the data is read from the second portion and checked for errors. If the copy in the second portion has excessive error bits, then a copy is written to the first portion.</p> <p>Col. 4:19-22, 15:67-16:3: After the data is written to the second portion, the data is read back to check of excessive error bits. If error bits exceeds a predetermined amount, then the data is rewritten at the less error-prone first portion.</p>

Claim Language	Citations to Prior Art
	<p>Col. 6:4-6, 24:45-50: Post-write reading every memory cell to check what has been written.</p> <p>Title: Post-write read to manage errors.</p> <p>Col. 4:55-67: The first copy in the second portion and the cached copy in the first portion are loaded into a set of data latches on an integrated circuit ship and compared to check for error bits. After the post-write read has detected an excessive amount of error in the first copy, a directory in a block management system embodied in the firmware of the controller is updated to render the first copy in the second portion obsolete and to direct subsequent access to the cached copy in the first portion. The cached copy in the first portion replaces the first copy in the second portion as the valid data.</p> <p>Col. 16:46-17:20: The first copy of the data page is read back in a post-write read to determine if there are any errors by comparing with the original copy which was cached. After the post-write read detects that the number of error bits in the data page has exceeded the predetermined amount, a second copy of the data page is made from the original data which was cached, and the second copy is rewritten to the first portion. A directory in a block management system embodied in the firmware of the controller is updated to render the first copy in the second portion obsolete and to direct subsequent access to the second copy in the first portion. The second copy in the first portion replaces the first copy in the second portion as the valid data.</p> <p>Fig. 15 refs. 520, 530, 540, col. 17:32-38: Reading the first copy from the second portion. Does the error exceed a predetermined number of error bits? If yes, then programming a second copy of the group of input data in the first portion.</p> <p>Col. 12:67-13:4, 29:57-59: Data is read from the second portion, latched in the data latches 430, and shifted out to the controller 102. The controller 102 compares the ECC in the read data to a second version of ECC computed from the read data.</p> <p>Col. 23:45-56: The controller performs checking EDC (error detection code).</p> <p>Col. 13:42-43: Post write E_{pw} (N_{cyc}) is bit errors that are present after the page is written.</p>

Claim Language	Citations to Prior Art
	<p data-bbox="516 269 1801 337">Fig. 14B: If post-write read failed, then rewrite input data page as a second copy of data page in first portion 410 of lower density.</p> <p data-bbox="516 378 1866 483">Col. 4:41-46: If the post-write read has detected an excessive amount of error in the first copy, the cached copy in the first portion will replace the first copy in the second portion as valid data. The first copy will become obsolete and subsequent access will be directed to the cached copy.</p> <p data-bbox="516 524 1850 667">Col. 17:50-56: If the post-write read has detected an excessive amount of error in the first copy in the second portion, then the cached copy in the first portion will replace the first copy in the second portion. The first copy in the second portion will become obsolete and a directory in a block management system will be updated to direct subsequent access to the cached copy in the first portion.</p> <p data-bbox="516 708 1877 813">Col. 18:30-44: The memory array includes data latches onto which the first copy in the second portion and the cached copy in the first portion can be loaded and compared. The comparison is not made at the controller.</p> <p data-bbox="516 854 1864 997">Col. 18:53-67: After a post-write read detects excessive number of error bits, a second copy of the data page is take from the cached copy and rewritten to the second section of the first portion. The directory in the block management system embodied in the firmware of the controller is updated to direct subsequent access to the second copy.</p>
<p data-bbox="205 1037 495 1383">[1f] determine which of the blocks of the plurality of the blocks in the MLC and SLC non-volatile memory modules are accessed most frequently by maintaining a count of the number of times each one of the</p>	<p data-bbox="516 1037 1881 1143">Gavens discloses the controller is adapted to determine which of the blocks of the plurality of the blocks in the MLC and SLC (as substituted) non-volatile memory modules are accessed most frequently by maintaining a count of the number of times each one of the blocks is accessed.</p> <p data-bbox="516 1216 1864 1321">For example, Gavens discloses that the controller determines which physical blocks in the second portion and the first portion are erased/programmed most frequently by maintaining a hot count of the number of erase/program cycles each physical block has undergone.</p>

Claim Language	Citations to Prior Art
<p>blocks is accessed; and</p>	<p>Fig. 19 refs. 720-730, col. 20:18-22: Tracking the age of each block by maintaining a hot count that records the number of erase/program cycling each block has undergone. Is the hot count of a memory block greater than a predetermined hot count threshold?</p> <p>Col. 5:41-45, 19:47-51: A hot count is maintained with each erase block of memory cells. The hot count tracks the number of times the erase block has been cycled through erase and program operations.</p> <p>Col. 14:46-47, 15:10-15, 19:62-64: The number of program-erase cycles an erase block has experienced is measured by a cycle count N_cyc (also known as hot count).</p> <p>Col. 15:2-6: Flash memory has an endurance problem as it ages with use. The error rate increase with the number of program-erase cycling.</p> <p>Col. 17:57-67: The entire disclosure of Lee US5,930,167 is incorporated by reference. Lee US5,930,167 col. 2:54-3:7: The usual desire to evenly wear the memory. Avoid using one portion of memory much more than other portions. Spread out heavy use function across the memory. Maintain separate counts of the number of times that each sector has been erased and programmed to even the wear among the sectors.</p> <p>Col. 16:29-40: The entire disclosure of Chen US6,456,528 is incorporated by reference. Chen US6,456,528 col. 11:7-10: Maintain counts of the number of erase/programming cycles of the individual blocks.</p> <p>Col. 20:53-59: The entire disclosure of Gorobets U.S. Appl. Ser. No. 12/642,584, published as US2011/0153912, is incorporated by reference. Gorobets US2011/0153912 para. 126: Leveling erase/rewrite counts among all blocks.</p> <p>Col. 8:40-43: The entire disclosure of Paley US2010/0172180 is incorporated by reference. Paley US2010/0172180 para. 542: The endurance history of a block in the main memory is tracked by a hot count.</p>

Claim Language	Citations to Prior Art
<p>[1g] allocate those blocks that receive the most frequent writes by transferring the respective contents of those blocks to the at least one SLC non-volatile memory module.</p>	<p>Gavens, alone or in combination with other reference(s), renders obvious the controller is adapted to allocate those blocks that receive the most frequent writes by transferring the respective contents of those blocks to the at least one SLC non-volatile memory module (as substituted).</p> <p>For example, Gavens discloses enabling an enhanced error management on blocks whose erase/program count is above a threshold so that a failure of the post-write read test will cause a transfer of data in those blocks to the first portion.</p> <p>Fig. 19 refs. 730-740, col. 20:21-29: If the hot count of a memory block is greater than a predetermined threshold, then enable the error management for the rest of the life of the memory.</p> <p>Fig. 20C ref. 7, col. 22:1-3: The new D3 block cannot be used and the file system will direct access to corresponding data that reside in D1 instead.</p> <p>Fig. 22A ref. 870, col. 22:66-67: The data in the D3 block is deemed bad, so the data must be accessed from the original copy in D1.</p> <p>Fig. 15 refs. 520-540, col. 17:32-38: If there is an error in reading the first copy from the second portion, then program a second copy in the first portion.</p> <p>Col. 5:45-48, 19:51-54: Whenever a hot count of an erase block passes a predetermined hot count threshold, the enhanced post-write-read error management will commence.</p> <p>Col. 16:34-37: It is desirable to have longer operational life of memory cells.</p> <p>Col. 17:57-67: The entire disclosure of Lee US5,930,167 is incorporated by reference. Lee US5,930,167 col. 2:54-3:7: The usual desire to evenly wear the memory. Avoid using one portion of memory much more than other portions. Spread out heavy use function across the memory. Maintain separate counts of the number of times that each sector has been erased and programmed to even the wear among the sectors.</p>

Claim Language	Citations to Prior Art
	<p>Col. 16:29-40: The entire disclosure of Chen US6,456,528 is incorporated by reference. Chen US6,456,528 col. 10:41-60: Memory cell blocks to which data are written most frequently are operated in two states while the remaining blocks that are rewritten less frequently are operated in multi-state. Data that is rewritten frequently is directed by the controller to a block operating in two states. Storing the most frequently written data in SLC and less frequently written data in MLC avoids the need to replace blocks that reach their endurance limit.</p> <p>Col. 20:53-59: The entire disclosure of Gorobets U.S. Appl. Ser. No. 12/642,584, published as US2011/0153912, is incorporated by reference. Gorobets US2011/0153912 para. 126: Leveling erase/rewrite counts among all blocks.</p> <p>Col. 8:40-43: The entire disclosure of Paley US2010/0172180 is incorporated by reference. Paley US2010/0172180 para. 542: A block in the MLC portion with the highest hot count is selected to be reallocated to the SLC portion.</p> <p>U.S. Prov. Appl. No. 61/509,257 ('298 patent admitted prior art as stated in the "Description of the Prior Art" section) para. 7: Wear leveling algorithms ensure that hot blocks (i.e., blocks that are frequently written) are not rendered unusable much faster than other blocks.</p> <p>Gavens discloses transferring data to the first portion in response to two conditions being met: (1) the hot count is above the threshold, and (2) the post-write read check fails. However, it would have been obvious to a person of ordinary skill in the art to transfer data to the first portion in response to only the first condition being met in order to avoid the post-write read check failures, achieve more efficient read/write operations, level the wear across the blocks, and prolong the lifetime of the flash memory device.</p> <p>US2008/0140918 ("Sutardja") discloses the controller is adapted to allocate those blocks that receive the most frequent writes by transferring the respective contents of those blocks to the at least one SLC non-</p>

Claim Language	Citations to Prior Art
	<p>volatile memory module (disclosed as higher-cost, faster memory distinct from another lower-cost, slower memory [0141]).</p> <p>See Exhibit A2 claim element 1g.</p> <p>US2009/0327591 (“Moshayed”) discloses the controller is adapted to allocate those blocks that receive the most frequent writes by transferring the respective contents of those blocks to the at least one SLC non-volatile memory module (explicitly disclosed as distinct from an MLC non-volatile memory module).</p> <p>See Exhibit A3 claim element 1g.</p> <p>Duann, Nelson, “SLC & MLC Hybrid,” Silicon Motion Flash Memory Summit, 2008 (“Duann”) discloses the controller is adapted to allocate those blocks that receive the most frequent writes by transferring the respective contents of those blocks to the at least one SLC non-volatile memory module.</p> <p>See Exhibit A4 claim element 1g.</p> <p>US 8,078,794 Lee et al. (“Lee ‘794’”) discloses allocation of logical blocks pre-determined to have most frequent writes (updates, such as system data and user directories) to a distinct SLC module and less frequent writes (user data) to a distinct MLC module.</p> <ul style="list-style-type: none"> • Lee ‘794 Fig. 2 refs. 226, 228, col. 5:10-13: The SLC flash memory array 226 may include at least one flash memory chip. Likewise, the MLC flash memory array 228 may include at least one MLC flash memory chip. • Lee ‘794 col. 6:53-7:39: allocation of logical blocks pre-determined to have most frequent writes (updates, such as system data and user directories) to a distinct SLC module and less frequent writes (user data) to a distinct MLC module.

Claim Language	Citations to Prior Art
<p>[3] The system of claim 1, wherein the minimum quanta of addresses is equal to one page.</p>	<p>Gavens discloses the minimum quanta of addresses is equal to one page.</p> <p>Fig. 4 ref. 70, col. 10:19-20, 41-47, 20:2-4: A page 70 is a group of memory cells enabled to be sensed or programmed in parallel.</p> <p>Col. 3:21-22: A page of memory elements are read or programmed together.</p> <p>Col. 11:20-29: A page is a minimum unit of programming and reading.</p> <p>Col. 9:36-40: The memory array 200 is arranged in rows and columns of memory cells, addressable by word lines and bit lines.</p> <p>Col. 16:29-40: The entire disclosure of Chen US6,456,528 is incorporated by reference. Chen US6,914,823 col. 10:62-65: Controller receives an address to write data to. The address is mapped to a particular page.</p> <p>Col. 20:53-59: The entire disclosure of Gorobets U.S. Appl. Ser. No. 12/642,584, published as US2011/0153912, is incorporated by reference. Gorobets US2011/0153912 para. 17: The host assigns files unique addresses, which are mapped to pages of the physical blocks of memory.</p> <p>Col. 8:40-43: The entire disclosure of Paley US2010/0172180 is incorporated by reference. Paley US2010/0172180 para. 20: The host assigns files unique addresses, which are mapped to pages of the physical blocks of memory.</p>
<p>[4] The system of claim 1, wherein the MLC non-volatile memory module is NAND flash memory.</p>	<p>Gavens discloses the MLC non-volatile memory module is NAND flash memory.</p> <p>Fig. 4, col. 9:49-50: A page of memory cells in the memory array 200 is organized in the NAND configuration.</p> <p>Col. 1:18-19, 4:14-18: Flash memory having a first portion and a second portion storing higher density</p>

Claim Language	Citations to Prior Art
	<p>compared to the first portion.</p> <p>Col. 2:65-66, 8:65-67: Flash memory devices with NAND string structures.</p> <p>Fig. 1 ref. 90: Flash memory device.</p>
<p>[5] The system of claim 1, wherein the SLC non-volatile memory module is NAND flash memory.</p>	<p>Gavens discloses the SLC non-volatile memory module (as substituted) is NAND flash memory.</p> <p>Fig. 4, col. 9:49-50: A page of memory cells in the memory array 200 is organized in the NAND configuration.</p> <p>Col. 1:18-19, 4:14-18: Flash memory having a first portion and a second portion storing higher density compared to the first portion.</p> <p>Col. 2:65-66, 8:65-67: Flash memory devices with NAND string structures.</p> <p>Fig. 1: Flash memory device 90.</p>
<p>[11] The system of claim 1, wherein the controller causes the transfer of content on a periodic basis.</p>	<p>Gavens, alone or in combination with other reference(s), renders obvious the controller causes the transfer of content on a periodic basis.</p> <p>For example, Gavens discloses that the controller performs operations in the background on a periodic basis.</p> <p>Col. 15:39-41: Data retention errors can be alleviated by periodically refreshing the threshold levels of the cells in a “read scrub” operation.</p> <p>Col. 17:57-67: The entire disclosure of Lee US5,930,167 is incorporated by reference. Lee US5,930,167</p>

Claim Language	Citations to Prior Art
	<p>col. 2:7-10, 30-37: New data is initially stored in SLC. Later, preferably in the background without slowing down other operations, the data is read out of SLC and reprogrammed into MLC.</p> <p>Sutardja discloses the controller causes the transfer of content on a periodic basis. See Exhibit A2 claim 11.</p> <p>US8,612,804 (“Kang”) discloses the controller causes the transfer of content on a periodic basis. See Exhibit A3 claim 11.</p>