

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SK HYNIX INC., SK HYNIX AMERICA INC., and SK HYNIX MEMORY
SOLUTIONS INC.,

Petitioners,

v.

NETLIST, INC.

Patent Owner

Patent No. 9,535,623

Issued: January 3, 2017

Filed: June 1, 2016

Inventors: Hyun Lee

Title: Memory Module Capable of Handshaking with a Memory Controller
of a Host System

Inter Partes Review No. IPR2018-00303

**DECLARATION OF DR. DONALD ALPERT REGARDING U.S. PATENT
NO. 9,535,623**

TABLE OF CONTENTS

I.	INTRODUCTION	1
A.	Engagement	1
B.	Background and Qualifications	1
C.	Compensation and Prior Testimony	3
D.	Information Considered.....	4
II.	LEGAL STANDARDS FOR PATENTABILITY	4
A.	Anticipation	6
B.	Obviousness.....	7
III.	The 623 Patent.....	13
A.	Effective Filing Date of the 623 Patent.....	13
B.	Person Of Ordinary Skill in the Art	14
C.	Overview of The 623 Patent.....	14
D.	The Prosecution History of The 623 Patent	18
	1. 745 Application.....	18
	2. 115 Application	19
	3. 721 Application.....	23
	4. 339 Application.....	24
E.	Challenged Claims of the 623 Patent	26
F.	Construction of Terms Used in the 623 Patent Claims	34
	1. “mode”	35
	2. “notification signal”	37
	3. “training”	39
	4. “indicating.....status”.....	41
IV.	OVERVIEW OF THE PRIOR ART	43
A.	U.S. Patent Publication No. 2008/0098277 to Hazelzet (Ex. 1014).....	43
B.	U.S. Patent Publication No. 2008/0155378 to Amidi (Ex. 1015).....	45

C.	U.S. Patent Publication No. 2008/0147897 to Talbot (Ex. 1016).....	46
D.	U.S. Patent No. 8,139,430 to Buchmann (Ex. 1017)	48
E.	U.S. Patent No. 7,525,860 to Hokenmaier (Ex. 1018).....	50
F.	U.S. Patent No. 8,359,521 to Kim (Ex. 1019)	51
V.	COMPARISON OF THE PRIOR ART TO THE CLAIMS OF THE 623 PATENT.....	52
A.	Claims 1-29 Are Unpatentable.....	52
1.	Claim 1 is Unpatentable.....	52
2.	Claim 2 is Unpatentable.....	117
3.	Claim 3 is Unpatentable.....	119
4.	Claim 4 is Unpatentable.....	121
5.	Claim 5 is Unpatentable.....	124
6.	Claim 6 is Unpatentable.....	128
7.	Claim 7 is Unpatentable.....	129
8.	Claim 8 is Unpatentable.....	131
9.	Claim 9 is Unpatentable.....	135
10.	Claim 10 is Unpatentable.....	139
11.	Claim 11 is Unpatentable.....	146
12.	Claim 12 is Unpatentable.....	149
13.	Claim 13 is Unpatentable.....	154
14.	Claim 14 is Unpatentable.....	154
15.	Claim 15 is Unpatentable.....	155
16.	Claim 16 is Unpatentable.....	155
17.	Claim 17 is Unpatentable.....	156
18.	Claim 18 is Unpatentable.....	156
19.	Claim 19 is Unpatentable.....	157
20.	Claim 20 is Unpatentable.....	157
21.	Claim 21 is Unpatentable.....	158
22.	Claim 22 is Unpatentable.....	161

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

23. Claim 23 is Unpatentable.....	161
24. Claim 24 is Unpatentable.....	161
25. Claim 25 is Unpatentable.....	162
26. Claim 26 is Unpatentable.....	162
27. Claim 27 is Unpatentable.....	163
28. Claim 28 is Unpatentable.....	163
29. Claim 29 is Unpatentable.....	163

I. INTRODUCTION

A. Engagement

1. I have been retained by counsel for SK Hynix as an expert witness in the above-captioned proceeding. I have been asked to provide my opinion about the state of the art of the technology described in U.S. Patent No. 9,535,623 (“the 623 Patent”) (Ex. 1001) and on the patentability of the claims of this patent. The following is my written declaration on these topics.

B. Background and Qualifications

2. I am an independent consultant with Camelback Computer Architecture, LLC. My residence and place of business is at 365 Santa Ynez Way, Sacramento, CA 95816. I am over the age of eighteen, and I am a citizen of the United States.

3. I have 40 years of academic and industrial experience in applying, designing, studying, teaching, and writing about microprocessors and computer systems. I received an Electrical Engineering Ph.D. degree in 1984 from Stanford University. I earlier received an Electrical Engineering B.S. degree from MIT in 1973 and an Electrical Engineering M.S. degree from Stanford University in 1978. I have taught classes in computer architecture at Stanford, Tel Aviv, and Arizona State Universities.

4. From 1976 to 1977, I worked at Burroughs Corporation, where I designed peripheral interface controllers, including those for serial data

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

communications based on Intel 8080 microprocessor components. From 1980 to 1989, I was the lead architect for the design of three high-performance microprocessors at Zilog and National Semiconductor. Later, at Intel, I was the lead architect of the Pentium® Processor from 1989 to 1992 and of the 815 chipset from 1999 to 2000, both of which became the most widely used PC components of their time. The 815 chipset comprised two components: (1) a memory controller hub (MCH) that included a graphics controller and memory controller with interfaces to the CPU, 133 MHz SDRAM system memory modules, an optional, external graphics controller and (2) an I/O controller hub (ICH) that included various I/O controllers (e.g., network, hard drive, USB) for system peripheral devices. Additionally, I served as co-manager for the Itanium processor design from 1993-1997.

5. I am a Senior Member of the Institute of Electrical and Electronics Engineers (IEEE), and served as the chair of the IEEE Technical Committee on Microprocessors and Microcomputers from 1999 to 2000. I was the keynote speaker at the first Cool Chips conference, dedicated to the study of low-power microprocessors and systems. I have given invited lectures at several universities, and published ten papers in various professional journals and conference proceedings. My paper entitled “Architecture of the Pentium Processor,” was selected as best paper in IEEE Micro for 1993. I am a named inventor on over 30

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

U.S. patents that pertain to microprocessors, computer systems. and related technology.

6. I have reviewed the 623 Patent, and I am familiar with the patent's subject matter, which is within the scope of my education and professional experience. Based at least on my background in academia, industry, and consulting, I am familiar with the issues and technology relating to processors, chipsets, memory, and peripheral devices for computer systems. I have personally analyzed, developed, and tested such computer components and systems. More specifically, the Pentium® Processor and 815 chipset for which I was the lead architect at Intel implemented various features for initializing, calibrating, and testing the memory system, which comprised a memory controller and standard SDRAM modules.

7. My Curriculum Vitae, including my publications and patents, is submitted herewith as Ex. 1004.

C. Compensation and Prior Testimony

8. I am being compensated at a rate of \$550.00 per hour for my study and testimony in this matter. I am also being reimbursed for reasonable and customary expenses associated with my work and testimony in this investigation. My compensation is not contingent on the outcome of this matter or the specifics of my testimony.

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

9. I have previously submitted declarations in five IPR proceedings, namely IPR2014-00970, IPR2014-00971, IPR2015-00148, IPR2015-00191, and IPR2017-00548. Additionally I have testified as an expert witness in Federal Court and the ITC, as summarized in Exhibit 1004 at 6-7.

D. Information Considered

10. My opinions are based on my years of education, research and experience, as well as my investigation and study of relevant materials. In forming my opinions, I have considered the materials I identify in this report and those listed in the Petition's Attachment B.

11. I may rely upon these materials and/or additional materials to respond to arguments raised by the Patent Owner. I may also consider additional documents and information in forming any necessary opinions — including documents that may not yet have been provided to me.

12. My analysis of the materials produced in this investigation is ongoing and I will continue to review any new material as it is provided. This report represents only those opinions I have formed to date. I reserve the right to revise, supplement, and/or amend my opinions stated herein based on new information and on my continuing analysis of the materials already provided.

II. LEGAL STANDARDS FOR PATENTABILITY

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

13. In expressing my opinions and considering the subject matter of the claims of the 623 Patent, I am relying upon certain basic legal principles that have been explained to me.

14. First, I understand that for an invention claimed in a patent to be found patentable, it must be, among other things, new and not obvious from what was known before the invention was made.

15. I understand the information that is used to evaluate whether an invention is new and not obvious is generally referred to as “prior art” and generally includes patents and printed publications (*e.g.*, books, journal publications, articles on websites, product manuals, *etc.*).

16. I understand that in this proceeding the Petitioner has the burden of proving that the claims of the 623 Patent are anticipated by or obvious from the prior art by a preponderance of the evidence. I understand that “a preponderance of the evidence” is evidence sufficient to show that a fact is more likely true than it is not.

17. I understand that in this proceeding, the claims must be given their broadest reasonable interpretation consistent with the specification. The claims after being construed in this manner are then to be compared to the information in the prior art.

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

18. I understand that in this proceeding, the information that may be evaluated is limited to patents and printed publications. My analysis below compares the claims to patents and printed publications that are prior art to the claims.

19. I understand that there are two ways in which prior art may render a patent claim unpatentable. First, the prior art can be shown to “anticipate” the claim. Second, the prior art can be shown to have made the claim “obvious” to a person of ordinary skill in the art. My understanding of the two legal standards is set forth below.

A. Anticipation

20. I understand that the following standards govern the determination of whether a patent claim is “anticipated” by the prior art.

21. I have applied these standards in my evaluation of whether the claims of the 623 Patent would have been anticipated by the prior art.

22. I understand that the “prior art” includes patents and printed publications that existed before the earliest filing date (the “effective filing date”) of the claim in the patent. I also understand that a patent will be prior art if it was filed before the effective filing date of the claimed invention, while a printed publication will be prior art if it was publicly available before that date.

23. I understand that, for a patent claim to be “anticipated” by the prior art, each and every requirement of the claim must be found, expressly or inherently, in a single prior art reference as recited in the claim. I understand that claim limitations that are not expressly described in a prior art reference may still be there if they are “inherent” to the thing or process being described in the prior art. For example, an indication in a prior art reference that a particular process complies with a published standard would indicate that the process must inherently perform certain steps or use certain data structures that are necessary to comply with the published standard.

24. I understand that if a reference incorporates other documents by reference, the incorporating reference and the incorporated reference(s) should be treated as a single prior art reference for purposes of analyzing anticipation.

25. I understand that it is acceptable to consider evidence other than the information in a particular prior art document to determine if a feature is necessarily present in or inherently described by that reference.

B. Obviousness

26. I understand that a claimed invention is not patentable if it would have been obvious to a person of ordinary skill in the field of the invention at the time the invention was made.

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

27. I understand that the obviousness standard is defined in the patent statute (35 U.S.C. § 103(a)) as follows:

28. A patent may not be obtained although the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be denied by the manner in which the invention was made.

29. I understand that the following standards govern the determination of whether a claim in a patent is obvious. I have applied these standards in my evaluation of whether the asserted claims of the 623 Patent would have been considered obvious as of the priority date of the patent.

30. I understand that to find a claim in a patent obvious, one must make certain findings regarding the claimed invention and the prior art. Specifically, I understand that the obviousness question requires consideration of four factors (although not necessarily in the following order):

- The scope and content of the prior art;
- The differences between the prior art and the claims at issue;
- The knowledge of a person of ordinary skill in the pertinent art; and
- Whatever objective factors indicating obviousness or non-obviousness may be present in any particular case.

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

31. In addition, I understand that the obviousness inquiry should not be done in hindsight, but must be done using the perspective of a person of ordinary skill in the relevant art as of the effective filing date of the patent claim.

32. I understand the objective factors indicating obviousness or non-obviousness may include: commercial success of products covered by the patent claims; a long-felt need for the invention; failed attempts by others to make the invention; copying of the invention by others in the field; unexpected results achieved by the invention; praise of the invention by those in the field; the taking of licenses under the patent by others; expressions of surprise by experts and those skilled in the art at the making of the invention; and the patentee proceeded contrary to the accepted wisdom of the prior art. I also understand that any of this evidence must be specifically connected to the invention rather than being associated with the prior art or with marketing or other efforts to promote an invention. I am not presently aware of any evidence of “objective factors” suggesting the claimed methods are not obvious, and reserve my right to address any such evidence if it is identified in the future.

33. I understand the combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results. I also understand that an example of a solution in one field of endeavor may make that solution obvious in another related field. I also understand that market

demands or design considerations may prompt variations of a prior art system or process, either in the same field or a different one, and that these variations will ordinarily be considered obvious variations of what has been described in the prior art.

34. I also understand that if a person of ordinary skill can implement a predictable variation, that variation would have been considered obvious. I understand that for similar reasons, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using that technique to improve the other device would have been obvious unless its actual application yields unexpected results or challenges in implementation.

35. I understand that the obviousness analysis need not seek out precise teachings directed to the specific subject matter of the challenged claim, but instead can take account of the “ordinary innovation” and experimentation that does no more than yield predictable results, which are inferences and creative steps that a person of ordinary skill in the art would employ.

36. I understand that sometimes it will be necessary to look to interrelated teachings of multiple patents; the effects of demands known to the design community or present in the marketplace; and the background knowledge possessed by a person having ordinary skill in the art. I understand that all these

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

issues may be considered to determine whether there was an apparent reason to combine the known elements in the fashion claimed by the patent at issue.

37. I understand that the obviousness analysis cannot be confined by a formalistic conception of the words “teaching, suggestion, and motivation.” I understand that in 2007, the Supreme Court issued its decision in *KSR Int'l Co. v. Teleflex, Inc.*, 550 U.S. 398 (2007), where the Court rejected the previous requirement of a “teaching, suggestion, or motivation to combine” known elements of prior art for purposes of an obviousness analysis as a precondition for finding obviousness. It is my understanding that *KSR* confirms that any motivation that would have been known to a person of skill in the art, including common sense, or derived from the nature of the problem to be solved, is sufficient to explain why references would have been combined.

38. I understand that a person of ordinary skill attempting to solve a problem will not be led only to those elements of prior art designed to solve the same problem. I understand that under the *KSR* standard, steps suggested by common sense are important and should be considered. Common sense teaches that familiar items may have obvious uses beyond the particular application being described in a reference, that if something can be done once it is obvious to do it multiple times, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle. As such, the prior art

considered can be directed to any need or problem known in the field of endeavor and can provide a reason for combining the elements of the prior art in the manner claimed. In other words, the prior art does not need to be directed towards solving the same problem that is addressed in the patent. Further, the individual prior art references themselves need not all be directed towards solving the same problem.

39. I understand that an invention that might be considered an obvious variation or modification of the prior art may be considered non-obvious if one or more prior art references discourages or lead away from the line of inquiry disclosed in the reference(s). A reference does not “teach away” from an invention simply because the reference suggests that another embodiment of the invention is better or preferred. My understanding of the doctrine of teaching away requires a clear indication that the combination should not be attempted (*e.g.*, because it would not work or explicit statements saying the combination should not be made).

40. I understand that a person of ordinary skill is also a person of ordinary creativity.

41. I further understand that in many fields, it may be that there is little discussion of obvious techniques or combination, and it often may be the case that market demand, rather than scientific literature or knowledge, will drive design trends. When there is such a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

ordinary skill has good reason to pursue the known options within their technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense. In that instance the fact that a combination was obvious to try might show that it was obvious. The fact that a particular combination of prior art elements was “obvious to try” may indicate that the combination was obvious even if no one attempted the combination. If the combination was obvious to try (regardless of whether it was actually tried) or leads to anticipated success, then it is likely the result of ordinary skill and common sense rather than innovation.

III. The 623 Patent

A. Effective Filing Date of the 623 Patent

42. I understand that the 623 Patent claims priority to U.S. Provisional Application No. 61/186,799, filed June 12, 2009. Ex. 1001 (623 Patent) at 1. Because the prior art discussed in my declaration was either filed or published before the filing of the provisional application, whether the 623 Patent is entitled to the filing date of the provisional application is irrelevant to my analysis. Therefore, I assume for purposes of these proceedings that the effective filing date of the 623 Patent is **June 12, 2009**. Ex. 1008 (U.S. Provisional Application No. 61/186,799) at 1.

B. Person Of Ordinary Skill in the Art

43. Based on my experience and the prior art I discuss herein, I believe a person of ordinary skill in the art in the field of the 623 Patent would be someone who was familiar with computer memory systems and basic CPU architecture documented in the computer/memory architecture literature and generally available in commercial systems as of 2009. Such literature and commercial systems would have included techniques related to how computer components access a computer's memory, including the role of a memory controller, the basic operation of memory modules and devices, and the techniques used to couple memory devices to the other components of the computer system. They would also include schemes for initialization of computer systems, including initialization of memory components. Such a person would also have been familiar with the standards that govern the operation of memory devices. The person of ordinary skill in the art would have had a bachelor's degree in computer engineering, or a related field, and several years of additional experience working with computer memory systems.

C. Overview of The 623 Patent

44. The 623 Patent states that its "disclosure relates to the operation of memory modules," and specifically "relates to a memory module capable of handshaking with a memory controller of a host system." Ex. 1001 (623 Patent) at

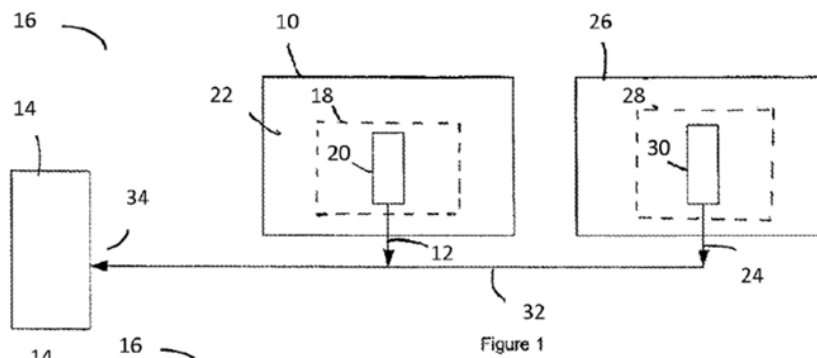
1:23-26. The 623 Patent explains that such “initialization sequence may comprise one or more training sequences.” Ex. 1001 (623 Patent) at 6:53-55.

45. According to the 623 Patent, before the patent “[i]n general, there [was] *no existing method of handshaking* between the MCH (e.g., system memory controller) and a memory subsystem (e.g., memory module) during initialization.” Ex. 1001 (623 Patent) at 3:49-52 (emphasis added). The patent states that “in conventional systems, the system memory controller does not monitor the error-out signal from the memory subsystem.” *Id.* at 3:52-54. The 623 Patent states that “typical” servers of the prior art would instead, after commencing an initialization function in a memory module, simply wait a predetermined amount of time and then assume the memory subsystem had completed initialization. Ex. 1001 (623 Patent) at 3:55-4:7.

46. The 623 Patent suggests that use of “handshaking” between memory controller and memory module (i.e., exchanging information about the completion of the initialization procedure) would be a better idea. Ex. 1001 (623 Patent) at 3:43-48. It states that, generally, “handshaking can be implemented in at least two ways; polling and notifying.” Ex. 1001 (623 Patent) at 4:27-28. The 623 Patent claims that polling is “generally inefficient because the system memory controller does not know exactly when the memory subsystem will have completed the required or requested operation,” thus causing the controller to “wait longer than

necessary to poll the memory subsystem, thereby delaying the overall initialization process.” Ex. 1001 (623 Patent) at 4:33-39. It further states that “the notifying method is an advantageous handshaking method between the MCH and the memory subsystem controller.” Ex. 1001 (623 Patent) at 4:48-50.

47. The patent provides only limited detail on what it purports to be the inventive system. Figure 1, for example (below), depicts the preferred embodiment. That figure shows a computer system including memory module 10 coupled to a memory controller 14 via “output 12” which is driven by notification circuit 20 of controller circuit 18.



Ex. 1001 (623 Patent) at 5:8-26, FIG. 1. Memory module 26 is also shown, and has essentially in the same structure as module 14, having “output 24” driven by notification circuit 30 of controller circuit 28. *Id.* at 7:7-28.

48. In one specific example of handshaking described in the patent, the interface between the memory controller (MCH) and memory subsystem controller (of the memory module) is implemented using a technique called open drain

signaling. Ex. 1001 (623 Patent) at 4:64-5:1; *see also id.* at 10:30-36, FIGS. 2-3. Specifically, output 12 of module 10 and output 24 of module 26 are both tied to the same bus 32 using an open drain configured transistor. Ex. 1001 (623 Patent) at 4:64-5:7, 10:34-54, FIGS. 2-3.

49. In that configuration, when starting an initialization procedure each memory module will drive its output active, placing an active signal on line 32. When the initialization is complete, the memory module can de-assert the signal on line 32. However, because all of the outputs of multiple memory modules are tied to the same line 32 in an open drain configuration, the state of line 32 will remain active until each memory module de-asserts its output. Ex. 1001 (623 Patent) at 10:30-11:33. In this manner, the system memory controller may be provided the status of an initialization procedure (*e.g.*, either that it is currently executing or complete) using only a single line from multiple memory modules. The patent further states that line 32 could be tied to a standard-defined (such as JEDEC) error output pin that is not required by the standard to be used during initialization. Ex. 1001 (623 Patent) at 8:54-9:16.

50. The patent further states that output 12 can be operatively coupled to the error-out pin of the memory module 10, which is connected to a multiplexor 42 driving the transistor 36 with either of a `task_in_progress` signal 44 or an error signal 46 (*e.g.*, parity error signal). Ex. 1001 (623 Patent) at 12:2-6, FIG. 3.

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

According to the 623 Patent, “the memory module 10 can be advantageously configured to both perform the standard (*e.g.*, JEDEC-specified) error reporting functionality via the error-out pin during the operational mode and provide the status notification functionality during the system initialization mode.” Ex. 1001 (623 Patent) at 12:12-17.

D. The Prosecution History of The 623 Patent

1. 745 Application

51. The 623 Patent issued from U.S. Patent Application Serial No. 15/169,745 (“745 Application”), which is a continuation of U.S. Patent Application Serial No. 15/088,115 (“115 Application”), filed on April 1, 2016, which is a continuation of U.S. Patent Application Serial No. 13/942,721 (“721 Application”), filed on July 16, 2013, now U.S. Patent No. 9,311,116, which is a continuation of U.S. Patent Application Serial No. 12/815,339 (“339 Application”), filed on July 14, 2010, which is now U.S. Patent No. 8,489,837, which in turn claims priority to U.S. Provisional Application 61/186,799 filed June 12, 2009. Ex. 1001 (623 Patent) at 1.

52. The 745 Application presented 37 claims. Ex. 1002 at 37-43. (Application at 22-28). On July 29, 2016, the examiner rejected claims 1-34 on the ground of nonstatutory double patenting as being unpatentable over claims 1-20 of U.S. Patent No. 9,311,116. Ex. 1002 (623 Patent File History) at 54-59.

53. In response, the applicants of the 745 Application cancelled claims 5, 6, 15, 18, 25, 28, 29, and 36, and filed a terminal disclaimer to overcome the nonstatutory double patenting rejections. Ex. 1002 at 63-71. The applicants further amended claims 2, 7, 13, 14, 30, and 37 for “minor issues.” Ex. 1002 (623 Patent File History) at 64-71.

54. On November 23, 2016, the examiner issued a notice of allowance, allowing claims 1-4, 7-14, 16-17, 19-24, 26-27, 30-35, and 37 of the 745 Application (issued as claims 1-29 of the 623 Patent). Ex. 1002 (623 Patent File History) at 100-107.

2. 115 Application

55. The 115 Application presented 20 claims, 1-20. Ex. 1005 (115 Application File History) at 30-33 (Application at 20-23). On May 20, 2016, the examiner rejected claims 1-20 on the ground of nonstatutory double patenting as being unpatentable over claims 1-20 of U.S. Patent No. 9,311,116 and claims 1-18 of U.S. Patent No. 8,489,837. Ex. 1005 (115 Application File History) at 49-57. Claims 1-20 were rejected also under 35 U.S.C. 103(a) as being unpatentable over Tanguay (Ex. 1009, US 2010/0042778) in view of Roohparvar (Ex. 1010, US 2003/0115427). *Id.*

56. In response, the applicants of the 115 Application cancelled claims 1-20, and added new claims 21-42. Ex. 1005 at 68-81. The applicants also filed a terminal disclaimer to overcome the nonstatutory double patenting rejections. *Id.*

57. On November 25, 2016, the examiner rejected claims 21-42 as being unpatenable over Moshayedi (Ex. 1011, U.S. 2010/0202240) in view of Fahr (Ex. 1012, U.S. 2008/0256281). Ex. 1005 (115 Application File History) at 100-107.

58. On April 25, 2017, the applicants of the 115 Application filed a Response. Ex. 1005 (115 Application File History) at 132-144. In the Response, the applicants amended claims 21 and 35 to “address issues with antecedent basis and for better clarity.” *Id.* at 138.

59. The applicants further argued that the Applicant is left to guess what the Examiner’s rationale is in rejecting the claims under Moshayedi in view of Fahr, because Moshayedi shows several outputs including “CACHE_DIRTY, DRAM_AVAILABLE, NVDIMM_READY, TEST_TX, and VBACK_RESET,” that cannot be “an open drain output” outputting “a signal indicating a parity error.” Ex. 1005 at 139-141. The applicants further argue that Moshayedi fails to disclose “configured to receive from the memory controller via the second edge connections address and command signals associated with one or more memory read, write, pre-charge, or refresh operations and to control the memory elements in accordance with the address and command signals,” *id.* at 142, and “wherein the

module controller is further configured to process one or more training sequences unrelated to any memory read, write, pre-charge, or refresh operation, and wherein the module controller is further configured to drive a notification signal associated with the one or more training sequences to the memory controller via the open drain output of the module controller and the error edge connection.” *Id.* at 143.

60. The applicants further argued that Fahr does not make up for the deficiency of Moshayedi, because the Fahr describes “two open-drain outputs for two different error signals, /Error (CE) 120 and /Error (UE) 121, respectively,” while the claim requires “that the module controller ‘outputs to the memory controller a signal indicating a parity error’ AND ‘is further configured to drive a notification signal associated with the one or more training sequences to the memory controller’ **via the same open-drain output.**” Ex. 1005 (115 Application File History) at 143.

61. On May 25, 2017, the applicants of the 115 Application further filed an Amendment, amending claims 21, 22, 28-29, 34-36, and 42, responding to the Final Office Action dated November 25, 2016. Ex. 1005 (115 Application File History) at 167-179. In this Amendment, the applicants made similar arguments as in its April 25, 2017 Response. *Id.*

62. On May 31, 2017, the applicants of the 115 Application conducted an interview with the Examiner, discussing “the AFCP 2.0 filed 04/25/17 and

potential claim amendments to overcome the art of record.” Ex. 1005 (115 Application File History) at 182.

63. On June 16, 2017, the examiner rejected claims 21-42. Ex. 1005 (115 Application File History) at 184-191. In particular, the examiner rejected claims 21-42 under 35 U.S.C. 112(a), as failing to comply with the written description requirement, because the newly added limitations to claims 21, 29, and 35 (*i.e.*, “wherein the one or more training sequences are not associated with any memory read or write operation,” “while the memory module is not performing any memory read or write operation,” “while the memory module is not performing any memory read or write operation,” respectively) are not supported by the original disclosure and constitute new matter. Ex. 1005 (115 Application File History) at 186-187. The examiner further rejected claims 21-42 as being unpatentable over Moshayedi (Ex. 1011, U.S. 2010/0202240) in view of Fahr (Ex. 1012, U.S. 2008/0256281) and Goishi (Ex. 1013, U.S. 2010/0142383). Ex. 1005 (115 Application File History) at 188-190.

64. On July 14, 2017, the applicants of the 115 Application conducted an interview with the Examiner, discussing “the rejection of claims 21-42 under 35 USC 112 first paragraph and 35 USC 103 over Moshayedi (20100202240), Fahr (20080256281), and Goishi (20100142383), as well as potential claim amendments

to overcome the pending rejections.” Ex. 1005 (115 Application File History) at 201.

65. On September 18, 2017, the applicants of the 115 Application filed an Amendment, amending claims 21, 27-29, 33-35, and 41-42, responding to the Non-Final Office Action dated June 16, 2017. Ex. 1005 (115 Application File History) at 203-216. In this Amendment, the applicants made similar arguments as in its April 25, 2017 Response with respect to Mosheyedi and Fahr. Ex. 1005 (115 Application File History) at 210-216. The applicants further argued that Goishi does not make up for the deficiencies of Moshayeddi and Fahr without any substance. Ex. 1005 (115 Application File History) at 216.

66. On October 12, 2017, the examiner issued a notice of allowance, allowing claims 21-42 of the 115 Application. Ex. 1005 (115 Application File History) at 228-232.

3. 721 Application

67. The 721 Application presented 15 claims, 1-15 (Application at 21-23) Ex. 1006 (721 Application File History) at 27-29.

68. On August 5, 2014, the applicants of the 721 Application filed a Preliminary Amendment amending claims 1 and 11-15, canceling claims 4 and 8-10, and adding new claims 16-24. Ex. 1006 (721 Application File History) at 54-60.

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

69. On May 22, 2015, the examiner rejected claims 1-7 and 11-24 on the ground of nonstatutory double patenting as being unpatentable over claims 1-18 of U.S. Patent No. 8,489,837. Ex. 1006 (721 Application File History) at 64-69.

70. On August 24, 2015, the applicants of the 721 Application filed a terminal disclaimer to overcome the double patenting rejection. Ex. 1006 (721 Application File History) at 74-80, 86.

71. On November 16, 2015, the examiner issued a Final Office Action maintaining the double patenting rejection noting that “the person who signed the terminal disclaimer is not an attorney or agent of record, is not recognized as an officer of the assignee, and has not been established as being authorized to act on behalf of the assignee.” Ex. 1006 (721 Application File History) at 92.

72. On January 21, 2016, the applicants of the 721 Application filed a “properly signed terminal disclaimer” to overcome the double patenting rejection. Ex. 1006 (721 Application File History) at 97, 99-104.

73. On February 24, 2016, the examiner issued a Notice of Allowance, allowing claims 1-3, 5-7, and 11-24 of the 721 Application. Ex. 1006 (721 Application File History) at 115-119.

4. 339 Application

74. The 339 Application presented 20 claims, 1-20. Ex. 1007 (339 Application File History) at 28-31 (Application at 21-24). On September 14, 2012,

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

the examiner rejected claims 1-3, 5, 7, 11, 14, and 19, and allowed claims 4, 6, 8-10, 12, 13, 15-18, and 20. Claims 5 and 11 were rejected under 35 U.S.C. 112, first paragraph. Ex. 1007 (339 Application File History) at 60-61. Claims 1-3, 5, 7, 11, 14, and 19 were rejected under 35 U.S.C. 103(a) as being unpatenable over Tanguay (Ex. 1009, US 2010/0042778) in view of Roohparvar (Ex. 1010, US 2003/0115427). Ex. 1007 (339 Application File History) at 61-67.

75. In response, the applicants of the 339 Application noted that the examiner's 112 rejection must have referred to claims 5 and 14 not 5 and 11, and cancelled claims 5 and 14. Ex. 1007 (339 Application File History) at 90.

76. The applicants of the 339 Application also amended claims 1, 4, 11, and 19. Claim 1 was amended to add "wherein the at least one notification signal triggers the memory controller to execute an interrupt routine," claim 4 was amended to remove "and the memory controller is responsive the at least one notification signal indicating completion of the at least one initialization sequence by triggering execution of an interrupt routine by the memory controller." Ex. 1007 at 85. According to the applicants, the amendment to claim 1 rendered claims 1-3 patentable over Tanguay in view of Roohparvar. Ex. 1007 (339 Application File History) at 91.

77. On April 29, 2013, the examiner issued a notice of allowance, allowing claims 1-4, 6-13, and 15-20 of the 339 Application. Ex. 1007 (339 Application File History) at 121.

E. Challenged Claims of the 623 Patent

78. This declaration relates to claims 1-29 of the 623 Patent. Those claims read as follows:

1. A memory module configured to fit into a corresponding slot of a host system to operate with a memory controller of the host system, the memory module comprising:

a module controller having an open drain output, the module controller generating a parity error signal and driving the parity error signal to the memory controller of the host system via the open drain output while the memory module operates in a first mode, the parity error signal indicating a parity error having occurred in the memory module while the memory module operates in the first mode, wherein the module controller is configured to cause the memory module to enter a second mode in response to a command from the memory controller of the host system, the module controller generating a notification signal indicating at least one status of one or more training sequences while the memory module is in the second mode and outputting the notification signal to the memory controller of

the host system via the open drain output while the memory module is in the second mode; and

a printed circuit board having a first set of edge connections for communicating address and control signals from the memory controller of the host system, a second set of edge connections for communicating data signals between the memory module and the memory controller of the host system while the memory module operates in the first mode, and an error edge connection coupled to the open drain output of the module controller, the memory module communicating to the memory controller of the host system via the error edge connection the parity error signal while the memory module operates in the first mode and the notification signal while the memory module is in the second mode.

2. The memory module of claim 1, further comprising:

a plurality of synchronous dynamic random access memory elements, wherein, while the memory module is in the first mode, the module controller receives and processes address and control signals corresponding to read or write commands from the memory controller of the host system and transmits processed address and control signals to the plurality of synchronous dynamic random access memory elements.

3. The memory module of claim 1, wherein, while the memory module is in the second mode, the module controller drives the notification signal to a logic low

level via the open drain output to indicate a status of the one or more training sequences.

4. The memory module of claim 1, wherein the first mode is an operational mode of the memory module, and the second mode is a training mode of the memory module.

5. The memory module of claim 1, wherein the one or more training sequences are executed by the module controller in response to a command or a signal received from the memory controller of the host system.

6. The memory module of claim 1, wherein the module controller comprises one or more integrated circuit.

7. The memory module of claim 1, wherein the module controller includes a notification circuit comprising one or more transistors each having an open drain coupled to the open drain output.

8. The memory module of claim 7, wherein the notification circuit generates the notification signal by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.

9. The memory module of claim 7, wherein the notification circuit generates the parity error signal by driving a gate of each of the one or more transistors to a

logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.

10. The memory module of claim 7, wherein the notification circuit generates the notification signal using one or more OR logic elements.

11. The memory module of claim 1, wherein the module controller implements the one or more training sequences in response to a signal or a command received from the memory controller of the host system.

12. A memory module to be coupled to a memory controller of a host system, the memory module comprising:

- a plurality of synchronous dynamic random access memory elements;
- a printed circuit board (PCB) having a first set of edge connections for communicating address and control signals from the memory controller of the host system, and a second set of edge connections for communicating data signals between the memory module and the memory controller of the host system while the memory module operates in a first mode, the PCB further including an error edge connection in addition to the first set of edge connections and the second set of edge connections;

- a module circuit controlling the plurality of synchronous dynamic random access memory elements in response to read and write commands received from the memory controller of the host system via the first set of edge connections while

the memory module operates in the first mode, the module circuit generating a parity error signal indicative of a parity error having occurred in the memory module while the memory module operates in the first mode, the module circuit causing the memory module to enter a second mode in response to a command from the memory controller of the host system, the module circuit generating a notification signal in response to one or more training sequences while the memory module is in the second mode, the notification signal indicating a status of the one or more training sequences; and

a notification circuit having an open drain output coupled to the error edge connection of the PCB, the notification circuit driving the parity error signal to the memory controller of the host system via the open drain output and the error edge connection while the memory module operates in the first mode, the notification circuit driving the notification signal to the memory controller of the host system via the open drain output and the error edge connection while the memory module is in the second mode.

13. The memory module of claim 12, wherein the memory module executes the one or more training sequences in response to a signal or a command from the memory controller of the host system.

14. The memory module of claim 12, wherein the first mode is an operational mode of the memory module, and the second mode is a training mode of the memory module.

15. The memory module of claim 12, wherein the notification circuit generates the notification signal using one or more OR logic elements.

16. The memory module of claim 12, wherein the notification circuit is a physically separate circuit from the module circuit.

17. The memory module of claim 12, wherein the module circuit includes the notification circuit.

18. The memory module of claim 12, wherein the notification circuit includes one or more transistors each having an open drain coupled to the open drain output.

19. The memory module of claim 18, wherein the notification circuit generates the notification signal by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.

20. The memory module of claim 18, wherein the notification circuit generates the parity error signal by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.

21. A memory module operable in at least a first mode and a second mode, the memory module comprising:

- a plurality of synchronous dynamic random access memory elements;
- a printed circuit board (PCB) having a first set of edge connections for communicating address and control signals from the memory controller of the host system, and a second set of edge connections for communicating data signals between the memory module and the memory controller of the host system while the memory module operates in the first mode, the PCB further including an error edge connection in addition to the first set of edge connections and the second set of edge connections, wherein the PCB is configured to fit into a corresponding slot connector of the host system; and

- a module controller having an open drain output coupled to the error edge connection, the module controller being configured to transmit address and control signals to the plurality of synchronous dynamic random access memory elements in response to read or write commands received from the memory controller of the host system via the first set of edge connections while the memory module operates in the first mode, to generate a parity error signal indicating a parity error having occurred in the memory module while the memory module is in the first mode, and to transmit the parity error signal to the memory controller of the host system via the open drain output and the error edge connection while the memory

module is in the first mode, the module controller being further configured to cause the memory module to enter the second mode in response to a command from the memory controller of the host system, to generate a notification signal indicating a status of one or more training sequences while the memory module is in the second mode, and to transmit the notification signal to the memory controller of the host system via the open drain output and the error edge connection while the memory module is in the second mode.

22. The memory module of claim 21, wherein the first mode is an operational mode of the memory module, and the second mode is a training mode of the memory module.

23. The memory module of claim 21, wherein the memory module executes the one or more training sequences in response to a command or a signal received from the memory controller of the host system.

24. The memory module of claim 21, wherein the module controller includes a notification circuit comprising one or more transistors, each of the one or more transistors having an open drain coupled to the open drain output.

25. The memory module of claim 24, wherein the notification circuit is configured to generate the notification signal by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.

26. The memory module of claim 24, wherein the notification circuit is configured to generate the parity error signal by [sic] driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.

27. The memory module of claim 24, wherein the notification circuit is configured to generate the notification signal using one or more OR logic elements.

28. The memory module of claim 24, wherein the module controller comprises one or more integrated circuits.

29. The memory module of claim 21, wherein the module controller is further configured to implement the one or more training sequences in response to a signal or a command received from the memory controller of the host system.

F. Construction of Terms Used in the 623 Patent Claims

79. I understand that in an *inter partes* review proceeding the claims of the patent are to be given their broadest reasonable interpretation in light of the specification.

80. I also understand that where a patent applicant provides an explicit definition of a claim term in the specification that definition may control the interpretation of that term in the claim.

81. I also understand that if no explicit definition is given to a term in the patent specification the claim terms must be evaluated using the ordinary meaning

of the words being used in those claims, evaluated from the perspective of a person of ordinary skill in the art.

1. “mode”

82. I believe a person of ordinary skill in the art would conclude that the broadest reasonable construction of “*mode*” would be in accordance with its plain and ordinary meaning.

83. The 623 Patent uses the term “mode” in accordance with the plain and ordinary meaning. Ex. 1001 at Abstract, 1:43-3:14, 5:13-26, 6:50-65, 7:29-40. In the related IPR proceeding involving the parent 837 Patent, I understand Patent Owner argued that the term “*mode*” should be construed to mean “*a distinct behavioral state that a system may be switched to,*” IPR2017-00548, Patent Owner Response at 3. However, I also understand that Patent Owner’s expert later admitted that the 837 Patent (and hence the 623 Patent) uses the term “mode” in accordance with how a person of ordinary skill in the art would understand that term to mean, Ex. 1026 at 9:23-10:2, and that Patent Owner’s proposed interpretation is not in accordance with the plain and ordinary meaning, Ex. 1026 at 77:14-25.

84. Patent Owner’s understanding of its interpretation was inconsistent with the specification of the 837 and 623 Patents, since it precluded different modes from using the same memory operations, even when the specification

explicitly states the opposite. Ex 1001 at 7:34-40 (“For example, the system memory controller 14 may cause the memory module 10 to perform standard operations such as memory read/write, pre-charge, refresh, etc., while in operational mode, although it will be appreciated that one or more of these operations can also be performed by the memory module 10 while in initialization mode in certain embodiments.”); Ex. 1026 at 24:24-25:18, 25:24-29:25.

85. I understand that the Board concluded in the Institution Decision involving the parent 837 Patent that the term “mode” need not be interpreted at all. IPR2017-00548, Institution Decision. However, to the extent the Board now thinks otherwise, I agree that the word “state” – when understood without all of the implicit limitations Patent Owner sought to have read into the claims in the 837 Patent IPR – is a synonym for “mode” in the context of the 623 Patent. That word “state” is consistent with the disclosures of the 623 Patent and the dictionary Patent Owner has provided in the IPR for the parent 837 Patent. Ex. 1028 at 3. While Patent Owner’s dictionary uses the phrase “operational state,” Ex. 1028 at 3, it is clearly using “operational” more broadly than the 623 Patent, since the patent explicitly equates the disclosed “operational mode” with normal memory operations and the dictionary includes no such limitation. Ex. 1001 at 7:29-34; Ex. 1028 at 3. Construing “mode” to mean “state” therefore captures the ordinary meaning of “mode” and avoids any confusion from the inconsistent use of the

word “operational.” To be clear, I do not believe interpreting “mode” to mean “state” alters the analysis set forth in my declaration, since in my view the two words mean the same thing to a person of skill. As discussed below, the prior art discloses the first and second “modes” under any construction.

2. “notification signal”

86. I believe a person of ordinary skill in the art would conclude that the broadest reasonable construction of “notification signal” would at least include a “signal notifying the memory controller of the status of the training sequence, either in response to polling or without polling.”

87. This construction is consistent with both the intrinsic and extrinsic evidence. The claim language states that the notification signal “indicat[es] at least one status of one or more training sequences” to the memory controller. *See, e.g.*, Ex. 1001 (623 Patent) at 15:39-43. The specification similarly describes the notification signal as “indicating at least one status of one or more training sequences ... to the memory controller,” Ex. 1001 (623 Patent) at 1:55-60. The specification further describes the notification signal as “indicating at least one status of the at least one initialization sequence” to the memory controller, Ex. 1001 (623 Patent) at 5:20-26, 7:41-47, and explains that such “initialization sequence may comprise one or more training sequences.” Ex. 1001 (623 Patent) at 6:53-55.

88. I understand that Petitioners have taken the position before an Administrative Law Judge of the United States International Trade Commission that the specification of the parent 837 Patent disparages “polling” as “delaying the overall initialization process,” which results in “inefficiency,” while describing the “notifying method” as an “advantageous handshaking method,” Ex. 1001 (623 Patent) at 4:33-56, and that therefore under the *Phillips* claim construction standard this claim term should be interpreted to exclude polling, Ex. 1025 (337-TA-1023, Initial Determination) at 82-84. I also understand that the ALJ did not adopt that analysis, but instead concluded that “one of ordinary skill in the art arriving at the ‘plain and ordinary’ meaning of the term ‘notification signal’ in the context of the ’837 patent would recognize that it does not utilize polling.” Ex. 1025 (337-TA-1023, Initial Determination) at 85.

89. Nevertheless, in the Institution Decision of IPR2017-00548 (involving the parent 837 Patent), the Board agreed, at least for purposes of institution, that the “polling” embodiment of the prior art LeClerc patent satisfies the claimed “notification signal” under the broadest reasonable construction of that term. See IPR2017-00548, Institution Decision, at page 9 (“LeClerc discloses that ‘each memory module is . . . polled’ and may ‘indicate[] that it has completed its initialization’ or ‘that the module has not completed the initialization process.’ Ex. 1005 ¶¶59, 60. In other words, LeClerc discloses that the memory module provides

a notification signal that indicates a ‘status’ (of whether initialization has been completed) of the initialization sequence.”). I understand that Patent Owner has not disputed that conclusion and, indeed, its expert has declined to interpret “notification signal” as excluding polling. Ex. 1026 (Murphy Deposition Transcript in IPR2017-00548) at 63:17-21 (“Q Okay. So I take it then that you have not offered an opinion in this proceeding that the claims of the '837 patent should be construed in a manner to exclude polling? A That's correct.”).

90. Accordingly, I believe a person of ordinary skill in the art would conclude that the broadest reasonable construction “notification signal” would at least include a “signal notifying the memory controller of the status of the training sequence, either in response to polling or without polling.” As discussed below, however, the prior art Hazelzet, Buchman, and Amidi references each disclose a “*notification signal*” whether or not polling is covered by this claim term. On the other hand, Talbot discloses a “*notification signal*” under the broadest reasonable construction set forth here.

3. “training”

91. I believe a person of ordinary skill in the art would conclude that the broadest reasonable construction of “training” would not be limited to “training of a memory module” but would also include “training of a memory system, including, among others, a memory module and/or a memory controller.”

92. With respect to the 837 Patent, which shares the same specification as the 623 Patent, I understand that Netlist argued in the ITC that the term “initialization” (or “training”) is broad enough to cover “initialization” (or “training”) of the “memory controller” (as opposed to the “memory module”). I also understand that the ITC issued an Initial Determination concluding that “Clock-to-CA training,” which is a training of the memory controller, satisfies the “initialization” (or “training”) limitation. Ex. 1025 (337-TA-1023, Initial Determination) at 90-91.

93. I do not agree that such a construction is necessarily the most reasonable construction, given that the purpose of the alleged invention in the 623 Patent (and the 837 Patent) is to inform the memory controller when the memory module’s training/initialization is complete (*i.e.*, because the memory controller does not know the status of the training/initialization). *See, e.g.*, Ex. 1001 (623 Patent) at 4:1-56. However, I understand that Netlist’s proposed construction is within the broadest reasonable construction, given Netlist’s positions taken at the ITC.

94. Accordingly, I believe a person of ordinary skill in the art would conclude that the broadest reasonable construction “training” would not be limited to “training of a memory module” but would also include “training of a memory system, including, among others, a memory module and/or a memory controller.”

As discussed below, Hazelzet in view of Talbot or Buchman discloses “*training*” under both the most reasonable construction and the broadest reasonable construction. On the other hand, Hazelzet in view of Amidi discloses “*training*” under the broadest reasonable construction.

4. “indicating ... status”

95. I believe a person of ordinary skill in the art would conclude that the broadest reasonable construction of “indicating ... status” would not be limited to “providing the status that has already been determined,” but would also include “providing information that can be used to determine the status later when combined with other information.”

96. Initially, I do not agree that such a construction is necessarily the most reasonable construction of the limitation “*indicating ... status*,” given that the “status” of the training sequence would not be considered to have been provided to the memory controller if the memory controller has to determine it later using other information. Moreover, with respect to the 837 Patent, which shares the same specification as the 623 Patent, I understand that the ITC issued an Initial Determination concluding that the feedback signals from the memory module do not indicate a status of the training because “only the memory controller provides status information regarding the initialization sequence. ... Put differently, these feedback signals passing through the memory module do not amount to a ‘status’

until the memory controller utilizes them to make a status determination.” Ex. 1025 (337-TA-1023, Initial Determination) at 95, 96.

97. However, with respect to the 837 Patent, I understand that Netlist argued in the ITC that a looped back training data (*i.e.*, “ALERT n signal”) from the memory module can indicate to the memory controller the “status” of the initialization sequence (*i.e.*, Clock-to-CA training sequence), because the memory controller can determine whether training is complete using, among others, the looped back training data. Ex. 1025 (337-TA-1023, Initial Determination) at 94 (“Complainant also asserts that the ‘ALERT n signal’ provides notification to the memory controller regarding the status of the ‘Clock-to-CA’ training mode and that both the memory controller and memory module remain in the initialization mode until signaled otherwise by the ‘ALERT n signal.’ ... Complainant supports this contention by pointing to the fact that ‘ALERT n signal’ of the accused products ‘indicates to the memory controller (1) when the ‘eye opening’ has started, (2) when Clock-to-CA training is seeking the ending boundary of the ‘eye opening,’ and (3) when the ending boundary of the ‘eye opening’ has been found.”). I understand that Netlist’s proposed construction is within the broadest reasonable construction, given Netlist’s positions taken at the ITC.

98. Accordingly, I believe a person of ordinary skill in the art would conclude that the broadest reasonable construction of “*indicating ... status*” would

not be limited to “providing the status that has already been determined,” but would also include “providing information that can be used to determine the status later when combined with other information.” As discussed below, the prior art Hazelzet, Buchman, and Amidi references each disclose “*indicating ... status*” under both the most reasonable construction and the broadest reasonable construction. On the other hand, Talbot discloses “*indicating ... status*” under the broadest reasonable construction.

IV. OVERVIEW OF THE PRIOR ART

A. U.S. Patent Publication No. 2008/0098277 to Hazelzet (Ex. 1014)

99. U.S. Patent Publication No. 2008/0098277 to Hazelzet (“Hazelzet”) (Ex. 1014) was filed on October 23, 2006 and published on April 24, 2008. Therefore, I understand that Hazelzet is prior art to the 623 Patent under 35 U.S.C. §§102(a), (b), and (e). Hazelzet is entitled “High Density High Reliability Memory Module With Power Gating and a Fault Tolerant Address and Command Bus” and discloses such functionality. Ex. 1014 (Hazelzet) at Title.

100. Hazelzet discloses memory modules that can operate in an “ECC mode” or a “parity mode.” Ex. 1014 (Hazelzet) at FIG. 8; *see also id.* at ¶¶[0020], [0022], [0049], [0059]-[0062], [0064], [0069]-[0070], [0072], [0075], [0081], [0101]-[0103], [0106]-[0107], [0109].



FIG. 8

101. Hazelzet explains that the “ECC mode” is a mode that allows the host system to detect and/or correct “correctable errors” or “uncorrectable errors” (*e.g.*, “double bit error”). Ex. 1014 (Hazelzet) at ¶[0069] (“This ECC logic will ... correct all single bit errors and detect all double bit errors present on the twenty-two chip select gated data inputs.”) In the “ECC mode,” the detected correctable errors would be output through an “/Error (CE)” output pin and any uncorrectable errors (*e.g.*, double bit errors) would be reported through an “/Error (UE)” output pin. Ex. 1014 (Hazelzet) at ¶[0069]. Hazelzet explains that these two outputs (*i.e.*, CE and UE) are “open-drain outputs.” Ex. 1014 (Hazelzet) at ¶¶[0072], [0109].

102. Hazelzet further explains that the “parity operating mode” is a mode that allows the host system to “interrogate the device to determine the error

condition” of the memory module. Ex. 1014 (Hazelzet) at ¶[0064] (“A parity operating mode is also provided, in conjunction with error reporting circuitry to permit the system to interrogate the device to determine the error condition.”).

Hazelzet further explains that such “error condition” in the “parity mode” is also reported to the host “via the Uncorrectable Error (UE) line” when the memory module is in the “parity mode,” Ex. 1014 (Hazelzet) at ¶[0070], which is the open-drain output that is used to report the uncorrectable errors in the “ECC mode.” Ex. 1014 (Hazelzet) at ¶¶[0072], [0109].

103. Hazelzet further explains that its memory module is in the “ECC mode” when the “/ECC Mode” signal becomes “low,” Ex. 1014 (Hazelzet) at ¶[0069], and that the memory module switches to the “parity mode” when the “/ECC Mode” signal becomes “high.” Ex. 1014 (Hazelzet) at ¶[0070].

B. U.S. Patent Publication No. 2008/0155378 to Amidi (Ex. 1015)

104. U.S. Patent Publication No. 2008/0155378 to Amidi (“Amidi”) (Ex. 1015) was filed on December 21, 2006 and published on June 26, 2008.

Therefore, I understand that Amidi is prior art to the 623 Patent under 35 U.S.C. §§102(a) and (e). Amidi is entitled “Memory Modules With Error Detection and Correction” and discloses such functionality. Ex. 1015 (Amidi) at Title.

105. Amidi discloses memory systems that can enter a training mode, which is coincident with its error correction, during which the memory controller

trains itself using the error signals from the memory module by learning whether the error was “random” or whether the error was due to “a bad memory location” so that it could remember not to use the bad memory location in the future. Ex. 1015 (Amidi) at ¶¶[0024]-[0026]. In particular, Amidi explains how the memory system enters into a training mode (*i.e.*, error correction mode) when the memory controller “halt(s)” the normal operation of the memory system upon receiving an “error signal” from the memory module. *Id.*

106. The purpose of Amidi’s training is to find out whether there is a bad memory location, and if so, where. Ex. 1015 (Amidi) at ¶[0026]. As such, Amidi’s process of determining whether or not there is another error during retransmission is training regardless of whether there happens to be another error or not during retransmission. Ex. 1015 (Amidi) at ¶[0026].

C. U.S. Patent Publication No. 2008/0147897 to Talbot (Ex. 1016)

107. U.S. Patent Publication No. 2008/0147897 to Talbot (“Talbot”) (Ex. 1016) was filed on October 31, 2006 and published on June 19, 2008. Therefore, I understand that Talbot is prior art to the 623 Patent under 35 U.S.C. §§102(a) and (e). Talbot is entitled “Memory Controller Including A Dual-Mode Memory Interconnect” and discloses such functionality. Ex. 1016 (Talbot) at Title.

108. Talbot discloses a “training mode,” during which the memory controller monitors the looped back training pattern data and shifts the training

pattern data in increments until the looped back training pattern data is “byte-locked” (or “bit-locked”). Ex. 1016 (Talbot) at ¶[0052] (“Memory controller 100 obtains bit-lock and byte-lock receiving known data patterns on CRC path (block 430). In addition, memory controller 100 adjusts the phase of the BCMD clock signal by adjusting variable phase unit 294, so that buffer unit 170 may obtain bit-lock (i.e., bit alignment) and byte-lock (i.e., byte alignment) on the BCMD signal path (block 435). More particularly, memory controller 1000 may vary (shift) the pattern being sent by one bit time (UI) to ensure that buffer unit 170 is capturing each bit correctly and shifting in the serial bits and capturing entire eight-bit bytes on the correct byte boundary.”).

109. Once there is “byte-lock” (or “bit-lock”), the memory controller determines that the training has ended, and takes the memory system out of the “training mode.” Ex. 1016 (Talbot) at ¶[0052] (“Memory controller may then send a buffer command to take buffer unit 170 out of training mode (block 440).”), ¶[0089] (“Once the periodic training is complete, buffer unit 870 is placed back into normal mode as describe above in block 945.”).

110. Talbot explains that the training patterns from the “buffer units” that take the memory system out of the training mode are output to the memory controller using the “CRC signal path.” Ex. 1016 (Talbot) at ¶[0052] (“In the training mode, the BCMD signal path is looped back to the CRC signal path during

even MCLK cycles (block 415), and a training pattern (*e.g.*, 10101010 ...) is output on the CRC path during odd MCLK cycles (block 420).”); *see also* Ex. 1016 (Talbot) at ¶[0032] (“In addition, memory controller 100 may control the configuration, training and testing of the buffer units 170 by sending a variety of loopback commands, CRC control commands, and CRC training pattern commands, for example.”), ¶[0052] (“Memory controller may then send a buffer command to take buffer unit 170 out of training mode (block 440).”), ¶¶ [0072], [0089], FIG. 4 (explaining that the “Buffer [is] In Training Mode.”).

111. Talbot explains that “CRC” stands for “cyclic redundancy code.” Ex. 1016 (Talbot) at ¶¶[0011], [0026]. Talbot further explains that a “CRC code” may be used for error detection. Ex. 1016 (Talbot) at ¶[0073] (“[A] CRC code may be used to provide such multiple bit error detection.”).

D. U.S. Patent No. 8,139,430 to Buchmann (Ex. 1017)

112. U.S. Patent No. 8,139,430 to Buchmann *et al.* (“Buchmann”) was filed on July 1, 2008 and issued on March 20, 2012. Therefore, I understand that Buchmann is prior art to the 623 Patent under 35 U.S.C. §102(e). Buchmann is entitled “Power-On Initialization and Test for a Cascade Interconnect Memory System” and discloses such functionality. Ex. 1017 (Buchmann) at Title.

113. Buchmann discloses two modes of operation (*i.e.*, “SBC mode” and “high-speed mode”), Ex. 1017 (Buchmann) at Abstract, 1:39-44. The “high-speed

mode” is a mode of operation “at a high nominal speed.” Ex. 1017 (Buchmann) at 14:67-15:2. On the other hand, the “SBC mode” is a mode of operation “at a reduced rate” of communication. Ex. 1017 (Buchmann) at 14:65-67. According to Buchmann, before the high-speed bus between the memory controller and the memory module (*e.g.*, downstream bus 110 and upstream bus 112 in FIG. 1) can be used reliably to communicate with the memory modules, the memory module operates in the SBC mode to train the high-speed bus. Ex. 1017 (Buchmann) at 14:5-8, 3:52-54, 3:64-67; *see also id.* at FIG. 1, 4:11-14.

114. In particular, Buchmann discloses executing training and/or initialization sequences in its “SBC mode,” which is also referred to as an “error correcting code protected quasi-static bit communication” mode. Ex. 1017 (Buchmann) at 14:1-15:15; *see also id.* at 3:52-54, 5:6-22, 5:40-12:59. For example, the training and/or initialization sequences taught by Buchmann include “TS0-clock detection, test and repair; TS1-static configuration and calibration; TS2-downstream lane training and repair; TS3-upstream lane training and repair; TS4 packet training; TS5-initiate valid packets; TS6-frame lock; and TS7-read data latency calculation and inter-channel de-skew,” Ex. 1017 (Buchmann) at 5:40-46, FIG. 3, and “SDRAM Interface Initialization” including “memory delay line calibration; DDR3 SD RAM and RCD reset and initialization procedure; DDR3

read data gate delay optimization; read data de-skew; DDR3 read data strobe centering; and write leveling.” Ex. 1017 (Buchmann) at 11:48-53.

E. U.S. Patent No. 7,525,860 to Hokenmaier (Ex. 1018)

115. U.S. Patent No. 7,525,860 to Hokenmaier (“Hokenmaier”) was filed on April 30, 2007 and issued on April 28, 2009. Therefore, I understand that Hokenmaier is prior art to the 623 Patent under 35 U.S.C. §102(a) and §102(e). Hokenmaier is entitled “System and Method for Monitoring Temperature in a Multiple Die Package” and discloses such functionality. Ex. 1018 (Hokenmaier) at Title.

116. Hokenmaier discloses how one skilled in the art would design open-drain (or open-collector) outputs to drive notification signals. Ex. 1018 (Hokenmaier) at 3:52-54, FIGS. 4-7. Initially, Hokenmaier explains that the distinction between an open-drain and an open-collector design is merely a difference between what kind of transistors are used, *i.e.*, bipolar junction transistors (BJT) or a metal-oxide semiconductor (MOS) transistor. Ex. 1018 (Hokenmaier) at 6:27-8:7; FIGS. 4 and 6 (using MOS transistors for open-drain outputs); FIGS. 5 and 7 (using BJTs for open-collector outputs).

117. Hokenmaier further explains that there are two ways to design open-drain outputs. Ex. 1018 (Hokenmaier) at FIGS. 4 and 6. The first way is shown in FIG. 4 of Hokenmaier, *i.e.*, using an “open drain pull-down transistor,” and the

second way is shown in FIG. 6 of Hokenmaier, *i.e.*, using an “open drain pull-up transistor.” Ex. 1018 (Hokenmaier) at FIGS. 4 and 6; *id.* at 10:66-11:3. Likewise, there are two corresponding ways to design an open-collector output, *i.e.*, using an “open collector pull-down transistor” or an “open collector pull-up transistor.”

Ex. 1018 (Hokenmaier) at FIGS. 5 and 7; *id.* at 10:66-11:3.

F. U.S. Patent No. 8,359,521 to Kim (Ex. 1019)

118. United States Patent No. 8,359,521 to Kim et al. (“Kim”) was filed on January 22, 2008, the underlying application was published July 23, 2009, and the patent issued January 22, 2013. I therefore understand that Kim is prior art to the 623 Patent pursuant to 35 U.S.C. § 102 (e). Kim is entitled “Providing a Memory Device Having a Shared Error Feedback Pin” and discloses such functionality. Ex. 1019 (Kim) at Title.

119. Kim discloses a “system and method for providing a memory device having a shared error feedback pin.” Ex. 1019 (Kim) at Abstract. Kim discloses that sharing functionality with an error pin is desirable in order maximize use of available bandwidth, simplify the memory controller, Ex. 1019 (Kim) at 4:40-49, “mak[e] use of unused circuits within the subsystem,” Ex. 1019 (Kim) at 11:31-34, and enable compatibility with standard memory modules, Ex. 1019 (Kim) at 5:49-57. Kim teaches that this shared error feedback pin can be used to provide information such as status and error conditions and information related to

initialization. Ex. 1019 (Kim) at 12:18-26. Kim further teaches how this shared error pin can be implemented using an open drain configuration and/or an OR-gate.

Ex. 1019 (Kim) at 5:49-57; *see also id.* at 4:40-48; 6:1-16; FIG. 4, FIG. 5.

V. COMPARISON OF THE PRIOR ART TO THE CLAIMS OF THE 623 PATENT

A. Claims 1-29 Are Unpatentable

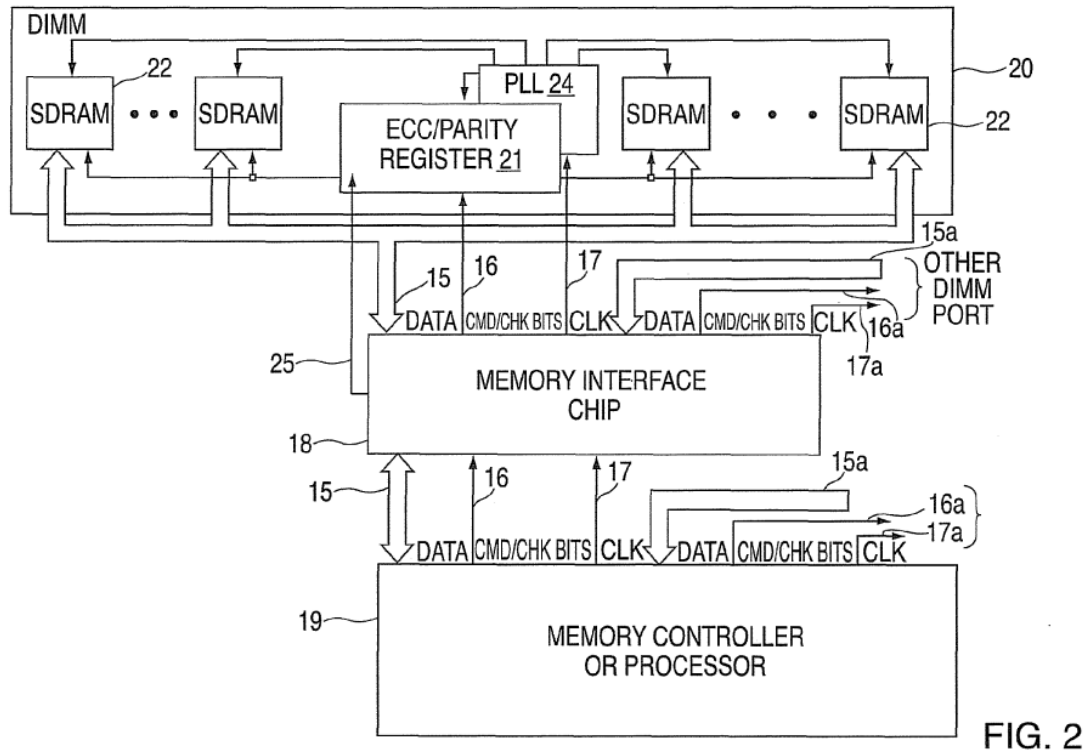
120. My analysis is based on the broadest reasonable interpretation of the claim terms in the 623 Patent. See § III.F.

1. Claim 1 is Unpatentable

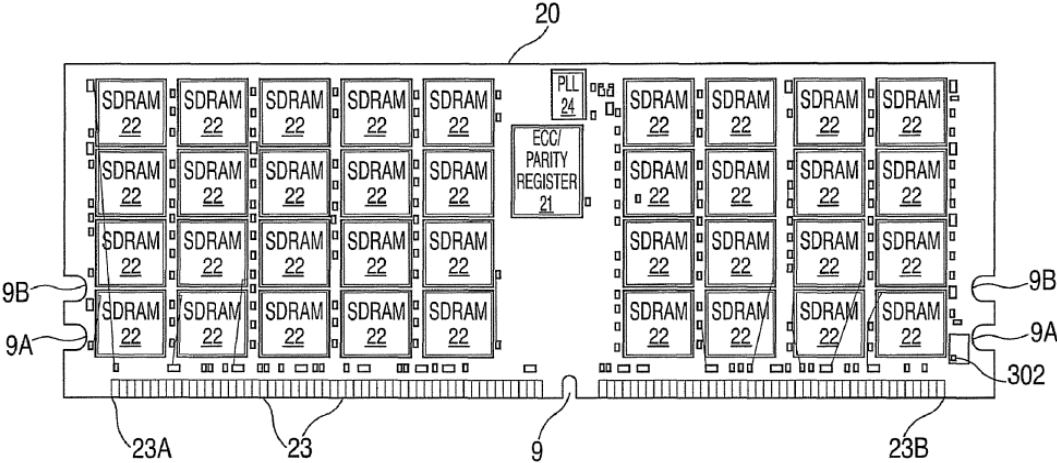
a) Preamble

121. The preamble of claim 1 requires “[a] memory module configured to fit into a corresponding slot of a host system to operate with a memory controller of the host system, the memory module comprising”:

122. Hazelzet discloses dual inline memory modules 20 (*i.e.*, “DIMMs”) that are configured to operate with the “memory controller” 19 of the host system. Ex. 1014 (Hazelzet) at ¶¶[0037], [0038], FIG. 2; *see also* Ex. 1014 (Hazelzet) at ¶[0036] (referring to DIMMs as “dual inline memory modules”).

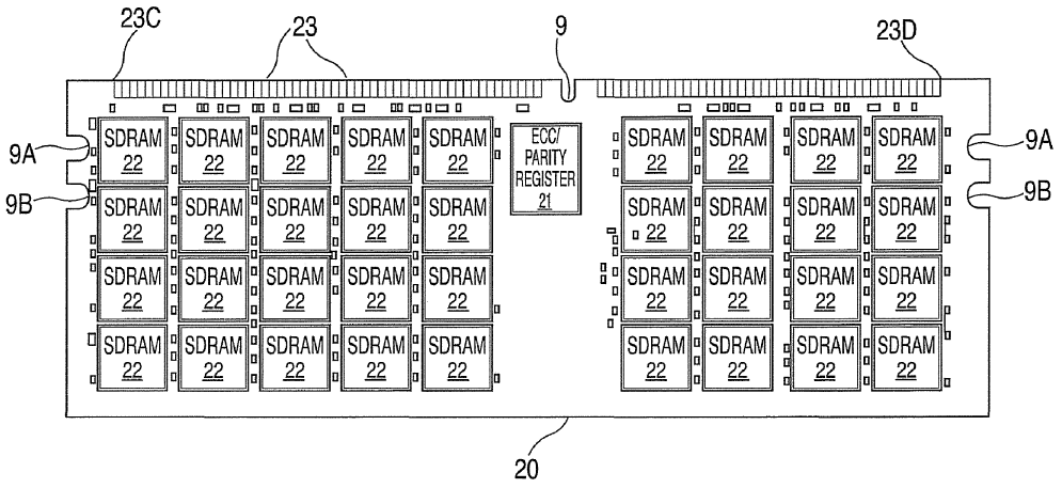


123. Hazelzet explains that the “DIMM of the present invention is comprised of a printed circuit board having a front side and a back side and a plurality of double data rate (DDR) DRAMs or synchronous dynamic random access memories (SDRAMs) affixed to both the front surface and the back surface,” Ex. 1014 (Hazelzet) at ¶[0015], examples of which are shown in FIGS. 2, 3A, 3B, 3C, and 3D. See Ex. 1014 (Hazelzet) at FIGS. 2, 3A-3D; see also *id.* at ¶[0037] (“Turning now to FIGS. 2, 3A, 3B, 3C, 3D, 4A, 4B, 5, 8, 9, 10 and 11, the enhanced server memory arrangement of the present invention will be described.”), ¶[0039].



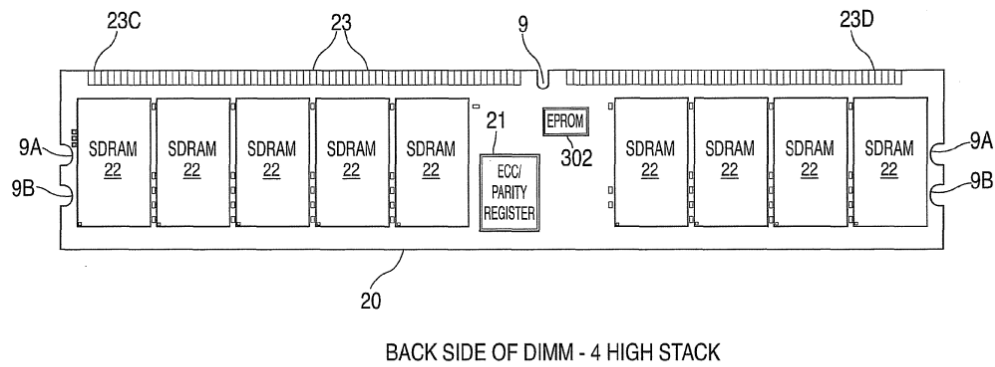
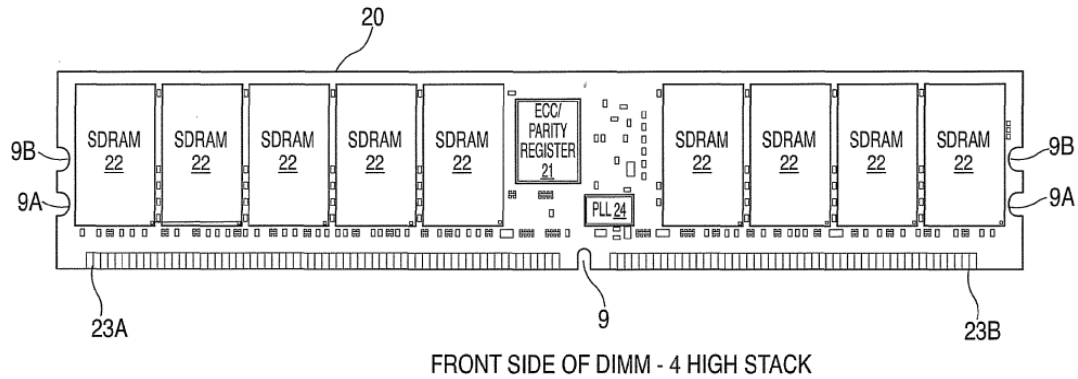
FRONT SIDE OF DIMM - 72 PLANAR

FIG. 3A



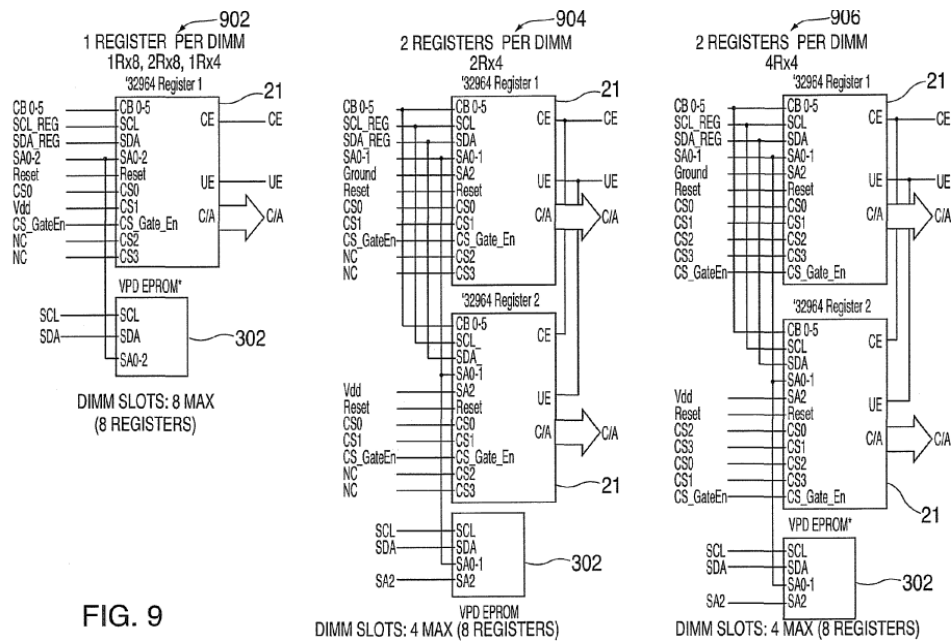
BACK SIDE OF DIMM - 72 PLANAR

FIG. 3B



124. Hazelzet further explains that its memory modules include “connectors 23 along the edge of both the back and front sides” of the printed circuit board, Ex. 1014 (Hazelzet) at ¶[0039] (“Generally speaking DIMMs are printed circuit cards designed to carry a plurality of DRAMs 22 thereon and the DRAM output pins (not shown) are connected via the printed circuit to selected connectors 23 along the edge of both the back and front sides of the card and are often provided with a single indexing key or notch 9 on the connector edge.”), which are provided to form connections to the memory controller of the host system when the memory module is inserted into a corresponding “slot” of the host

system. Ex. 1014 (Hazelzet) at ¶[0097] (“Referring to FIG. 11, column 1102 is the slot in the memory system being occupied by the memory module. ... For memory modules having one buffer chip 21, the IIC address correlates to the slot being occupied by the memory module as shown in column 1106”); *see also id.* at ¶[0032], FIG. 11, FIG. 9 (showing the maximum number of “DIMM slots” for “1, 2, and 4 rank DIMMs.”)



125. Accordingly, Hazelzet discloses “[a] memory module configured to fit into a corresponding slot of a host system to operate with a memory controller of the host system,” as claimed.

b) Module Controller Generating Parity Error Signal In The First Mode

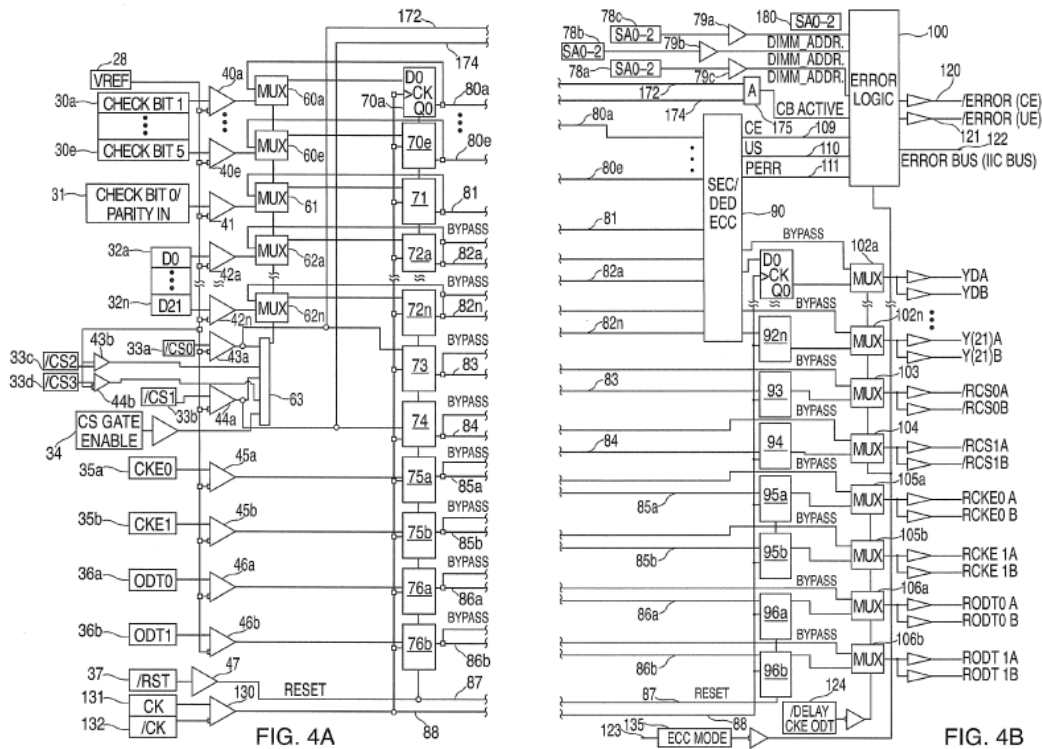
126. Claim 1 requires “a module controller having an open drain output, the module controller generating a parity error signal and driving the parity error

signal to the memory controller of the host system via the open drain output while the memory module operates in a first mode, the parity error signal indicating a parity error having occurred in the memory module while the memory module operates in the first mode.”

127. Hazelzet discloses an “ECC/Parity register,” which is the claimed “*module controller*,”¹ having an output for uncorrectable errors (shown in FIG. 4B as “/ERROR (UE) 121”), which is the claimed “*open drain output*.” Ex. 1014 (Hazelzet) at FIGS. 3A-3D, 4A, 4B, 5. FIGS. 4A and 4B of Hazelzet show the “ECC/Parity register” of FIGS. 3A-3D in greater detail, and FIG. 5 of Hazelzet

¹ Hazelzet explains that its “ECC/Parity register” (i.e., “*module controller*,” as claimed) is also referred to as a “buffer device” (or “ECC/Parity Buffer chip 21”). Ex. 1014 (Hazelzet) at ¶ [0038] (“In this FIG. 2 the server comprises a novel DIMM 20 provided with a novel ECC/Parity Buffer chip 21 (also referred to as a ‘buffer device’) coupled to the memory interface chip 18 which is in turn coupled to the memory controller or processor 19. ... The memory interface chip 18 then sends and receives data, via line 15, to the memory devices or DRAMs 22 and sends address and command information to the register chip 21 via add/cmd line 16 and check bits for error correction purposes to the ECC/ Parity register chip 21 via line 25.”).

shows the “single error correction / double error detection error correction code (SEC/DED ECC)” circuit of FIG. 4B. Ex. 1014 (Hazelzet) at ¶¶[0027], [0028].



128. As shown in FIG. 4B, the “SEC/DED ECC” circuit is coupled to an “ERROR LOGIC” 100, both of which being part of the “ECC/Parity register” (*i.e.*,

“*module controller*,” as claimed). The “ERROR LOGIC” of the “ECC/Parity register” (*i.e.*, “*module controller*,” as claimed) is coupled to an “error reporting circuitry” with two “open drain outputs,” one for correctable errors “CE” (120 of FIG. 4B) and another for uncorrectable errors “UE” (121 of FIG. 4B). Ex. 1014 (Hazelzet) at FIG. 4B, ¶[0072] (“Error reporting circuitry is included to permit external monitoring of DIMM operations. Two open-drain outputs are available to permit multiple modules to share a common signal line for reporting an error that occurred during a valid command (/CS=low) cycle (consistent with the re-driven signals). These two outputs are driven low for two clocks to allow the memory controller time to sense the error. /Error (CE) indicates that a correctable error occurred and was corrected by the ECC logic, /Error (UE) indicates that an uncorrectable error occurred and depending on the mode selected is an uncorrectable ECC error or a parity error. Note that the timing of /Error (UE) is different in parity mode vs. ECC mode.”), ¶[0109]. The output for uncorrectable errors (shown in FIG. 4B as “/ERROR (UE) 121”) is the “*open drain output*,” as claimed.

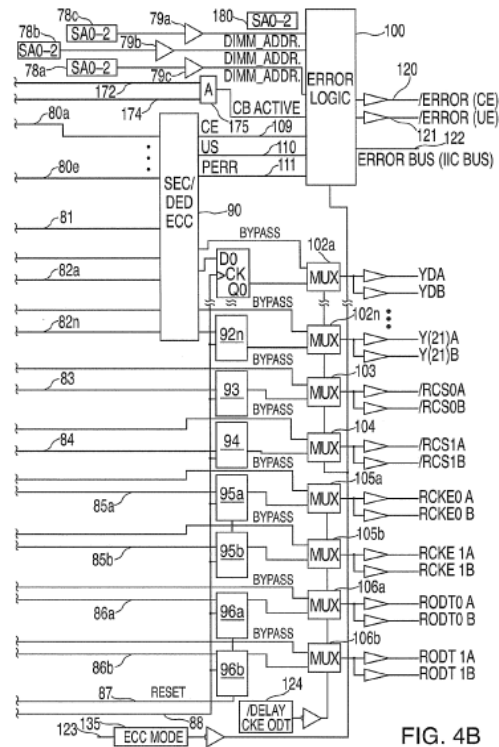


FIG. 4B

129. Accordingly, Hazelzet discloses “a module controller having an open drain output,” as claimed.

130. Hazelzet explains that its memory modules operate in (1) a “parity mode,” which is the claimed “*first mode* (or state),” and (2) an “ECC mode,” which is the claimed “*second mode* (or state).” Ex. 1014 (Hazelzet) at FIG. 8; see also *id.* at ¶¶[0020], [0022], [0049], [0059]-[0062], [0064], [0069]-[0070], [0072], [0075], [0081], [0101]-[0103], [0106]-[0107], [0109]. Hazelzet’s “parity mode” and “ECC mode” are “modes” under (1) the plain and ordinary meaning of that term, (2) Patent Owner’s construction (“a *distinct behavioral state that a system may be switched to*”), (3) or even “state.” Hazelzet uses the term “mode” in accordance with the plain and ordinary meaning, Ex. 1014 at FIG. 8, ¶¶[0044],

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

[0049], [0059]-[0065], [0069]-[0072], [0075], [0095], [0102], [0103], [0107], [0109], and therefore its “parity mode” and “ECC mode” satisfy the plain and ordinary meaning of the term “mode.” Further, Hazelzet’s “parity mode” and “ECC mode” are in a “distinct behavioral state that a system may be switched to,” because Hazelzet’s system cannot be in both “modes,” or behavioral states, at the same time, Hazelzet’s two modes comprise different behaviors, EX1014, FIG. 8, ¶¶[0069], [0070], and also because Hazelzet’s system switches between each “mode,” or behavioral state, in response to the “/ECC Mode” pin. Ex. 1014 at ¶[0069] (explaining that Hazelzet’s system is in an “ECC Mode” when “/ECC Mode” is switched to “low”), ¶[0070] (explaining that Hazelzet’s system is in a “parity mode” when the “/ECC Mode” is switched to “high”), ¶[0022].

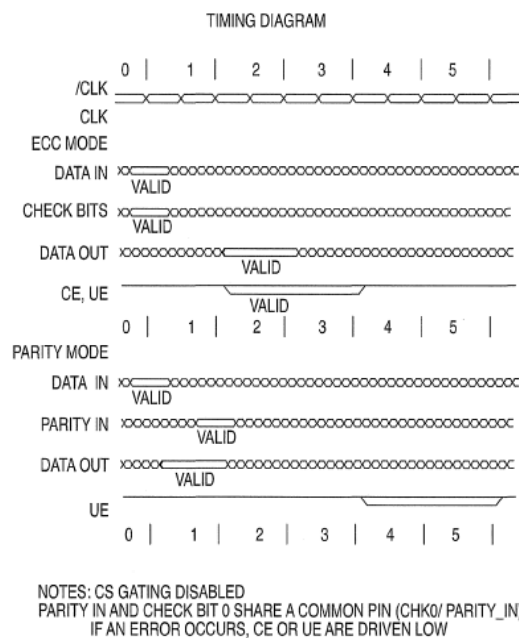


FIG. 8

131. The “ECC/Parity register” (*i.e.*, “*module controller*,” as claimed) includes an “SEC/DED ECC” 90. Ex. 1014 (Hazelzet) at FIG. 4B. FIG. 5 of Hazelzet, which is a block diagram of the “SEC/DED ECC” 90 in FIG. 4B, includes a “parity generator/checker circuit 231.” Ex. 1014 (Hazelzet) at ¶¶[0028], [0076], FIG. 5.

132. When the memory module is in the “parity mode” (*i.e.*, “*first mode*,” as claimed), the “parity generator/checker circuit 231” generates and sends the “parity error signal (PERR)” to the “error logic circuit” 100. Ex. 1014 (Hazelzet) at ¶[0076] (“FIG. 5 is a block diagram of the SEC/DED ECC circuit of FIG. 4B. ... The parity generator/checker circuit 231 further has a second input coupled to the parity in signal source 31 via primary latch 71 and output line 81 and depending on the state of the parity input signal on input 31 sends a parity error signal (PERR) on output line 111 to the error logic circuit 100.”), FIG. 5; *see also* Ex. 1014 (Hazelzet) at ¶[0070] (“In addition to the above ECC mode, the same twenty two chip select gated data signals can be operated in ‘parity mode (/ECC Mode high)’, whereby the signal received on CHK0/Parity in line is received as parity to the register one clock pulse later than the chip select gated data inputs. The received parity bit is then compared to the parity calculated across these same inputs by the register parity logic to verify that the information has not been corrupted. The twenty two chip select gated data signals will be latched and re-driven on the first

clock pulse and any error will be reported two clock pulses later via the Uncorrectable Error (UE) line (driven low for two clock pulses) and in the Error Bus Registers. No correction of errors will be completed in this mode. The convention of parity, in this application, is odd parity (odd numbers of 1s [sic] across data and parity inputs equals valid parity).”), FIG. 4B.

133. In the “parity mode” (i.e., “*first mode*,” as claimed), the “parity error signal (PERR)” 111 that is fed into the “error logic circuit” 100 is output to the memory controller through the output line 121 (i.e., output for uncorrectable error “UE”). Ex. 1014 (Hazelzet) at ¶[0059] (“This SEC/DED ECC circuit 90 provides three outputs 109, 110, and 111 to the error logic circuit 100. These outputs are: a correctable error (CE) line 109, an uncorrectable error (UE) line 110 and a parity error bit line 111 fed to the error logic circuit 100 which provides outputs regarding correctable and uncorrectable errors on output lines 120 and 121. When either [correctable] error line (CE) 109 or uncorrectable error line (UE) 110 is low this indicates that an error was identified as being associated with the address and/or command inputs (either correctable or uncorrectable). The error lines 120,121 will be active, i.e., low, for two clock cycles simultaneous with the re-driven address/command data when operating in ECC mode or delayed by two clock cycles when operating in parity mode.”). As explained above, the output line 121 (“UE”) is an open-drain output, Ex. 1014 (Hazelzet) at ¶¶[0072], [0109],

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

which is coupled to the memory controller of the host system using pin number 142. Ex. 1014 (Hazelzet) at FIG. 7A (showing that the “Server PIN Number 142” corresponds to “DIMM PIN Name UE”), FIG. 2 (showing how the “DIMM 20” is coupled to the “Memory Controller or Processor 19”); *see also id.* at ¶[0016] (“More particularly the server of the present invention comprises a novel DIMM provided with a new and unique ECC/ Parity Register that is operable with 1 to 4 memory ranks, coupled to a memory interface chip which is in turn coupled to a memory controller or processor such that the memory controller sends address and command information to the buffer (or register) via address/command lines together with check bits for error correction purposes to the ECC/ Parity register.”), ¶[0038] (“In this FIG. 2 the server comprises a novel DIMM 20 provided with a novel ECC/Parity Buffer chip 21 (also referred to as a ‘buffer device’) coupled to the memory interface chip 18 which is in turn coupled to the memory controller or processor 19. ... In this configuration, the check bits and/or parity bits associated with the ECC/Parity register chip 21 are developed in the memory interface chip 18, although in other embodiments these bits could be developed in the memory controller or processor 19, be communicated to the memory interface chip as a subset of the information on line 16 and the memory interface would then re-drive this information to the memory module on line 25”).

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

SERVER DIMM PIN NUMBER	PIN NAME	SERVER DIMM PIN NUMBER	PIN NAME
1	Vref	139	Vref
2	SCL	140	SCL
3	SDA	141	SDA
4	UE	142	UE
5	CE	143	CE
6	VDD	144	VDD
7	ECCMODE	145	ECCMODE
8	DLY CKE/ODT	146	DLY CKE/ODT
9	CS GATE EN	147	CS GATE EN
10	GND	148	GND
11	DQ0	149	DQ4
12	DQ1	150	DQ5
13	GND	151	GND
14	DQS0*	152	DQS9
15	DQS0	153	DQS9*
16	GND	154	GND
17	DQ2	155	DQ6
18	DQ3	156	DQ7
19	GND	157	GND
20	VDD	158	VDD
21	GND	159	GND
22	DQ8	160	DQ12
23	DQ9	161	DQ13
24	GND	162	GND
25	DQS1*	163	DQS10
26	DQS1	164	DQS10*
27	GND	165	GND
28	DQ10	166	DQ14
29	DQ11	167	DQ15
30	GND	168	GND
31	DQ16	169	DQ20
32	DQ17	170	DQ21
33	GND	171	GND
34	DQS2*	172	DQS11
35	DQS2	173	DQS11*
36	GND	174	GND
37	DQ18	175	DQ22
38	DQ19	176	DQ23
39	GND	177	GND
40	RESET	178	RESET
41	VDD	179	VDD
42	GND	180	GND
43	DQ24	181	DQ28
44	DQ25	182	DQ29
45	GND	183	GND
46	DQS3*	184	DQS12
47	DQS3	185	DQS12*
48	GND	186	GND
49	DQ26	187	DQ30
50	DQ27	188	DQ31
51	GND	189	GND
52	CB0	190	CB4

FIG. 7A

134. Hazelzet explains that the generated parity error signal is driven to the memory controller of the host system. Ex. 1014 (Hazelzet) at ¶[0044] (“FIG. 4B shows the output portion and error correction code circuit segment 21b. As depicted in FIGS. 4A and 4B, the buffer chip (also referred to herein as the buffer device) includes a plurality of buffer circuits (e.g., receivers, multiplexers, latches, etc.) for driving the address and command data.”), FIG. 2 (showing that the signals generated by the “DIMM 20” is driven to the “Memory Controller or Processor 19” through data paths 15 and 15a), ¶[0036] (“Each DRAM 11 on the DIMM 10 has a plurality of input/output pins that are coupled, via the printed circuitry on the

DIMM 10 to the contacts on the DIMM 10 and these contacts are further coupled, via a data line 15, to a memory interface chip 18 and to a memory controller or processor 19.”); *see also id.* at ¶[0072] (“Two open-drain outputs are available to permit multiple modules to share a common signal line for reporting an error that occurred during a valid command (/CS=low) cycle (consistent with the re-driven signals). These two outputs are driven low for two clocks to allow the memory controller time to sense the error. /Error (CE) indicates that a correctable error occurred and was corrected by the ECC logic, /Error (UE) indicates that an uncorrectable error occurred and depending on the mode selected is an uncorrectable ECC error or a parity error. Note that the timing of /Error (UE) is different in parity mode vs. ECC mode.”).

135. Accordingly, Hazelzet discloses “*the module controller generating a parity error signal and driving the parity error signal to the memory controller of the host system via the open drain output while the memory module operates in a first mode,*” as claimed.

136. Hazelzet explains that the “parity operating mode” is a mode that allows the host system to “determine the error condition” of the memory module. Ex. 1014 (Hazelzet) at ¶[0064] (“A parity operating mode is also provided, in conjunction with error reporting circuitry to permit the system to interrogate the device to determine the error condition.”). Hazelzet further explains that such

“error condition” is reported to the host “via the Uncorrectable Error (UE) line” when the memory module is in the “parity mode.” Ex. 1014 (Hazelzet) at ¶[0070] (“In addition to the above ECC mode, the same twenty two chip select gated data signals can be operated in 'parity mode (/ECC Mode high), whereby the signal received on CHK0/Parity in line is received as parity to the register one clock pulse later than the chip select gated data inputs. The received parity bit is then compared to the parity calculated across these same inputs by the register parity logic to verify that the information has not been corrupted. The twenty two chip select gated data signals will be latched and re-driven on the first clock pulse and any error will be reported two clock pulses later via the Uncorrectable Error (UE) line (driven low for two clock pulses) and in the Error Bus Registers. No correction of errors will be completed in this mode. The convention of parity, in this application, is odd parity (odd numbers of 1s [*sic*] across data and parity inputs equals valid parity).”), ¶[0107] (“The 22 'CS-gated signals will be latched and re-driven on the first clock and any error will be reported two clocks later via the /Error (UE) line (driven low for two clocks) and in the Error Bus Registers.”). Hazelzet further explains that the error line is driven “low, i.e., ‘0’,” when “an error is detected.” Ex. 1014 (Hazelzet) at ¶[0099] (“Parity error reporting in the present invention is realized by delivering the parity signal one cycle after the address and command to which it applies, and driving the error line low, i.e., ‘0’ two clock cycles after the

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

address and command bits are driven to the DRAMs from the memory interface chip if an error is detected. After holding the error line low for only 2 clock cycles, the driver will be disabled and the output permitted to return to an un-driven state (high impedance) allowing this line to be shared by multiple modules.”), ¶[0018] (“Still another object of the invention is to provide for Parity error reporting in which the parity signal is delivered one cycle after the address and command to which it applies, and the error line be driven low two clocks after the address and command bits are driven to the DRAMs from the register on the DIMM.”), FIG. 8 (explaining that UE is driven low “if an error occurs” in the “parity mode”). As such, Hazelzet explains that the parity error signal output from the “ECC/Parity register” (*i.e.*, “*module controller*,” as claimed) indicates a parity error having occurred in the memory module while the memory module operates in the “parity mode” (*i.e.*, “*first mode*,” as claimed).

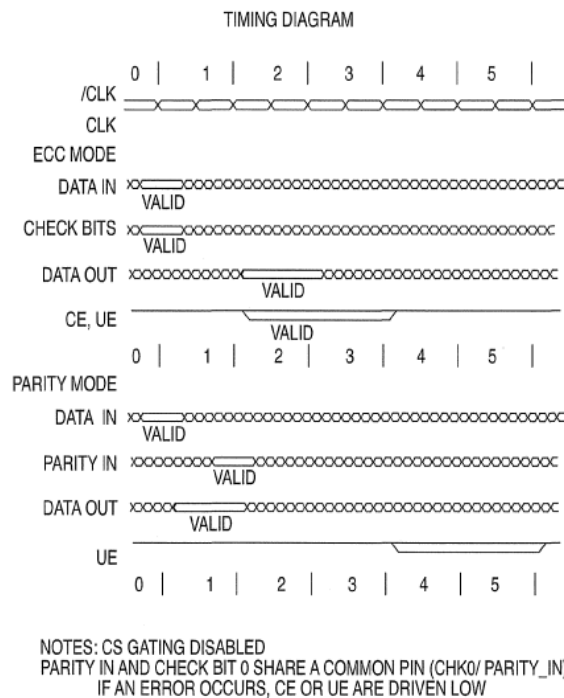


FIG. 8

137. Accordingly, Hazelzet discloses “*the parity error signal indicating a parity error having occurred in the memory module while the memory module operates in the first mode,*” as claimed.

138. As such, Hazelzet discloses “*a module controller having an open drain output, the module controller generating a parity error signal and driving the parity error signal to the memory controller of the host system via the open drain output while the memory module operates in a first mode, the parity error signal indicating a parity error having occurred in the memory module while the memory module operates in the first mode,*” as claimed.

c) Module Controller Generating Notification Signal

In The Second Mode

139. Claim 1 requires “*wherein the module controller is configured to cause the memory module to enter a second mode in response to a command from the memory controller of the host system, the module controller generating a notification signal indicating at least one status of one or more training sequences while the memory module is in the second mode and outputting the notification signal to the memory controller of the host system via the open drain output while the memory module is in the second mode.*”

140. As explained in claim element [1.b.], Hazelzet explains that its memory modules operate in (1) a “parity mode,” which is the claimed “*first mode*,” and (2) an “ECC mode,” which is the claimed “*second mode*.” Ex. 1014 (Hazelzet) at FIG. 8; see also *id.* at ¶¶[0020], [0022], [0049], [0059]-[0062], [0064], [0069]-[0070], [0072], [0075], [0081], [0101]-[0103], [0106]-[0107], [0109]. EX1003, ¶130.

141. Hazelzet explains that its “CS gating” circuitry within its “ECC/Parity register” (“*module controller*,” as claimed) can “cause” the memory module to be in an inactive state when the “chip select” commands² from the memory controller of the host system is not active (*i.e.*, “active low”). Ex. 1014 (Hazelzet) at ¶[0104]

² See Ex. 1014 (Hazelzet) at ¶ [0072] (referring to “/CS” as “commands”).

(“The present invention also provides CS gating of key inputs through the use of CS0, CS1, CS2, CS3 and CS Gate Enable. As a means to reduce device power, the buffer’s internal latches will only be updated when one or more of the CS inputs are active low (and CS Gate Enable at a high level) at the rising edge of clock. The 22 'CS-gated signals associated with this function include addresses (addr 0:15, BA 0:2). RAS, CAS, WE-with the remaining signals (CS, CKE, ODT) continuously re-driven at the rising edge of every clock as the re-drive of these inputs by the buffer does not require that one or more of the CS inputs to be at a low level. The CS gating function can be disabled by tying CS Gate Enable low enabling all internal latches to be updated on every rising edge of clock. This novel use of CS gating reduces the power dissipation of the buffer device and attached memory devices when any of the gated inputs (addr 0:15, BA 0:2) transition, as the transitions will not propagate through the buffer(s) to the memory device(s), thereby causing circuitry within the buffer and memory devices to switch, unless one or more of the CS inputs to the buffer(s) are at a low level (selecting the buffer and associated memory device(s).”); see also *id.* at ¶[0066] (“CS gating of key inputs, *e.g.*, /CS0, /CS1, /CS2 and /CS3, is provided as a means of reducing device power for the internal latches which will only be updated and related internal circuitry utilized when one, two, three or all of the chip select CS) inputs are active low (and chip select gate enable tied high) at the rising edge of the system clock.

The twenty-two chip select-gated signals associated with this function include addresses continuously re-driven at the rising edge of every clock depending on the state of chip select.”), claim 1 (reciting a “power savings means for causing one or more of the buffer circuits to be in an inactive mode when the chip select input lines are not active”). Such a “CS gating” circuitry is shown in FIG. 4A, which is part of the “ECC/Parity register” (“*module controller*,” as claimed). Ex. 1014 (Hazelzet) at ¶[0027], FIG. 4A. Hazelzet further explains that the chip select (“CS”) commands are “*from the memory controller of the host system.*” Ex. 1014 (Hazelzet) at ¶[0074] (explaining that “the /CS3 signal is applied via contact or pin 89 on the front side of the DIMM and also via contact or pin 225 on the back side of the DIMM.”), FIG. 7B (showing that the CS0, CS1, CS2, CS3 pins of the memory module is coupled to the memory controller of the host system using pin number 86, 91, 93, and 89, respectively, or 224, 229, 231, and 225, respectively). Because the memory module can be in an “ECC mode” only when the “ECC/Parity register” is in an “active” state, Hazelzet’s “ECC/Parity register” (“*module controller*,” as claimed) is configured to cause the memory module to enter an “ECC mode” (*i.e.*, “*second mode*,” as claimed) when at least one of the “chip select” commands from the memory controller of the host system is active (*i.e.*, “active low”). Ex. 1014 (Hazelzet) at ¶¶ [0027], [0066], [0074], [0104], FIG. 4A, claim 1. As such, under the broadest reasonable construction, Hazelzet

discloses that the “*module controller is configured to cause the memory module to enter a second mode in response to a command from the memory controller of the host system.*” *Id.*

142. To the extent the Board concludes that Hazelzet’s “chip select” is not a “*command*,” as claimed, e.g., because it does not command the “*memory module to enter a second mode*,” one skilled in the art would nevertheless have found this feature as obvious in view of Hazelzet for the following reasons.

143. In particular, Hazelzet explains that its memory module is in the “ECC mode” when the “/ECC Mode” signal becomes “low,” Ex. 1014 (Hazelzet) at ¶[0069], and that the memory module is in the “parity mode” when the “/ECC Mode” signal becomes “high.” Ex. 1014 (Hazelzet) at ¶[0070]. Because the ECC Mode signals are output from the “/ECC mode signal source 135,” Ex. 1014 (Hazelzet) at FIG. 4B, the “/ECC mode signal source 135” of the “ECC/Parity register” (i.e., “*module controller*,” as claimed) instructs the memory module to enter into the “ECC mode” (i.e., “*second mode*”) by generating and outputting a low “/ECC Mode” signal, which is separate from and in addition to the CS commands from the memory controller. Ex. 1014 (Hazelzet) at ¶[0062] (“The /ECC mode signal source 135 is coupled to a selection third input of secondary multiplexers 102a through 102n, 103 and 104 and to the error logic circuit 100.”); Ex. 1014 (Hazelzet) at ¶[0069] (“ECC Mode (/ECC Mode low): For all inputs

gated by CS, on-chip SEC/DED ECC logic is enabled and the signal received on CHKO/Parity in is received as check bit 0 when /ECC Mode input is low.”).

144. Further, Hazelzet explains that its ECC Mode input is “a programming pin.” Ex. 1014 (Hazelzet) at ¶[0106] (“Also for all inputs gated by CS, the on-chip SEC/DED ECC logic is enabled and the signal received on CHK0/Parity In is received as check bit 0 via a programming pin (/ECC Mode low).”). Hazelzet does not exclude any type of “programming” that can be used. *Id.* As such, one skilled in the art would have recognized that there are two ways to implement such programming, *i.e.*, either (1) internally within the memory module, or alternatively (2) externally by the user through the memory controller of the host system, and would have found it obvious to try both options. *Id.*

145. Accordingly, one skilled in the art would have found it obvious to implement the “ECC/Parity register,” such that its “/ECC mode signal source 135,” is configured to instruct and cause the memory module to enter into the “ECC mode” (*i.e.*, “*second mode*,” as claimed) in response to a command from the memory controller of the host system. Ex. 1014 (Hazelzet) at ¶¶[0027], [0060], [0062], [0066] [0069], [0070], [0074], [0104], [0106], FIG. 7A, FIG. 4A, FIG. 4B, FIG. 2. As such, Hazelzet discloses (or at least renders obvious) “*the module controller is configured to cause the memory module to enter a second mode in*

response to a command from the memory controller of the host system,” as claimed.

146. If an uncorrectable error is detected while the memory module is in the “ECC mode” (i.e., “*second mode*,” as claimed), the “ECC/Parity register” (i.e., “*module controller*,” as claimed) generates a notification signal that is “low, for two clocks,” and outputs the notification signal to the memory controller of the host system through the “/Error (UE)” 121 pin. Ex. 1014 (Hazelzet) at ¶[0059] (“When either [correctable] error line (CE) 109 or uncorrectable error line (UE) 110 is low this indicates that an error was identified as being associated with the address and/or command inputs (either correctable or uncorrectable). The error lines 120,121 will be active, i.e., low, for two clock cycles simultaneous with the re-driven address/command data when operating in ECC mode or delayed by two clock cycles when operating in parity mode.”), ¶[0072] (“Two open-drain outputs are available to permit multiple modules to share a common signal line for reporting an error that occurred during a valid command (/CS=low) cycle (consistent with the re-driven signals). These two outputs are driven low for two clocks to allow the memory controller time to sense the error. /Error (CE) indicates that a correctable error occurred and was corrected by the ECC logic, /Error (UE) indicates that an uncorrectable error occurred and depending on the mode selected is an uncorrectable ECC error or a parity error. Note that the timing of /Error (UE)

is different in parity mode vs. ECC mode.”), FIG. 2, FIG. 4B, FIG. 7A (showing that the UE pin of the memory module is coupled to the memory controller of the host system using pin number 142).

147. Before the information indicating whether a correctable error or an uncorrectable error occurred is driven through the “error lines 120, 121,” Ex. 1014 (Hazelzet) at ¶¶[0059], [0072], Hazelzet explains that such information is stored in the “status register” of the “error logic block 100.” Ex. 1014 (Hazelzet) at ¶[0079] (“The Error Logic block 100 consists of 3 major elements (not shown) which are an error counter, a status register block that contains a plurality of status registers, and an IIC logic block all of which are interconnected through common logic circuits.”), ¶[0085] (“The Byte 0: Status Register is the general status bit register that can be read to determine the type of error. ...

Byte 0: Status Register							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Extended	DIMM	DIMM	DIMM	Mode	Parity	ECC error	ECCerror
Status	Address	Address	Address	1=ECC	Error	1=UE	1=CE
Register	SA2	SA1	SA0	0=Pty	1=PERR		
(If 1, see Byte 9)							

.”); *see also id.* at ¶¶[0081]-[0090], [0122].

148. As such, Hazelzet explains that the notification signal from the “ECC/Parity register” (*i.e.*, “buffer chip”) indicates the status of the error correction sequence when the memory module is in the “ECC mode,” by indicating (1) whether “an error was identified” and (2) whether the error from the memory

module is an “uncorrectable error.” Ex. 1014 (Hazelzet) at ¶¶[0059] (“When either [correctable] error line (CE) 109 or uncorrectable error line (UE) 110 is low this indicates that an error was identified as being associated with the address and/or command inputs (either correctable or uncorrectable). The error lines 120,121 will be active, i.e., low, for two clock cycles simultaneous with the re-driven address/command data when operating in ECC mode or delayed by two clock cycles when operating in parity mode.”), [0069], [0072], FIG. 2, FIG. 4B, FIG. 7A; *see also id.*, at ¶[0092] (“As shown in configuration 902, outputs from the buffer chip 21 include an indication of whether the error is a correctable error (CE) or an uncorrectable error (UE) as well as the corrected (if error is correctable) command and address information, which connect to memory devices 22.”). Hazelzet’s notification signal is not a polled signal. In particular, the “address and/or command inputs” from the memory controller cannot be a request (*i.e.*, polling) for a status of the error correction sequence, because Hazelzet’s notification is sent to the memory controller only when “an error was identified” as opposed to whenever the memory controller sends the “address and/or command inputs” to the memory module. Further, the “status” of the error correction sequence, e.g., (1) whether “an error was identified” and/or (2) whether the error from the memory module is an “uncorrectable error,” Ex. 1014 (Hazelzet) at ¶[0059], is determined by Hazelzet’s memory module before it is indicated to the memory controller via the

notification signal. Ex. 1014 (Hazelzet) at FIG. 8 (explaining that “if an error occurs, CE or UE [of the “ECC/Parity register” within the memory module] are driven low”), ¶[0069] (“If a correctable error is detected, /Error (CE) will be driven low for two clocks and errors will be counted and latched in the error bus registers for the 28 inputs if this is the first error since a reset is issued. Any double bit error will also be detected (as well as many other errors that are not correctable), and will be reported on the /Error (UE) error line (driven low for two clocks) and in the error bus registers if this error is the first since a Reset is issued.”). Accordingly, Hazelzet discloses “*notification signal*” and “*indicating ... status*” even under the narrower construction of these terms.

149. As explained above in claim element [1.b.], “/Error (UE)” 121 is the “*open drain output*,” as claimed. Ex. 1014 (Hazelzet) at FIG. 4B, ¶¶ [0072], [0109]. The open-drain “/Error (UE)” 121 output is coupled to the memory controller of the host system using pin number 142. Ex. 1014 (Hazelzet) at FIG. 7A.

150. Accordingly, Hazelzet discloses or at least renders obvious “*the module controller generating a notification signal indicating at least one status of one or more [error correction] sequences while the memory module is in the second mode and outputting the notification signal to the memory controller of the host system via the open drain output while the memory module is in the second*

mode,” as claimed. Ex. 1014 (Hazelzet) at ¶¶[0059], [0069], [0072], [0092], [0109], FIG. 2, FIG. 4B, FIG. 7A.

151. To the extent the Board finds that Hazelzet does not explicitly disclose that its notification signal indicates the status of a “training” sequence, one skilled in the art would have nevertheless found it obvious in view of Hazelzet. In particular, one skilled in the art would have understood that the purpose of “training” was to avoid errors during normal operation. *See, e.g.*, U.S. Patent No. 7,647,467 to Hutsell et al. (“Hutsell”) (Ex. 1020) at Abstract (“In the training mode, the memory controller tunes one or more parameters (voltage references, timing skews, etc.) used in an upcoming operational mode.”). Accordingly, one skilled in the art would have found it obvious to use Hazelzet’s circuitry, which is capable of error detection/correction, for generating a notification signal that indicates the status of a training sequence along with error information. *See, e.g.*, U.S. Patent Application Pub. 2005/0071580 to LeClerc *et al.* (“LeClerc”)(Ex. 1027) at ¶[0091] (“If the process has completed, the buffer module returns a ‘Completed’ message to the processor, in block 608. In one embodiment, the buffer module also returns a ‘success’ or ‘failure’ indicator, and error information, if any is indicated in the error registers. The success or failure indicator and the error information can be sent as part of the Completed message, or as one or more separate messages, in various embodiments.”). Further, one skilled in the art

would have understood that it was generally known to use error signals, as disclosed in Hazelzet, as part of a “training” sequence. See, e.g., Ex. 1020 (Hutsell) at 6:37-8:6 (explaining that one skilled in the art understood error detection/correction to be a form of “training”), 7:21-52 (explaining that a “training mode” involves processing “errors.”); U.S. Patent No. 7,221,699 to Lindskog (“Lindskog”) (Ex. 1021) at Abstract (showing that one skilled in the art referred to the reference signal that is used for error correction as the “training signal.”). In addition, one skilled in the art would have found this claim to be obvious over Hazelzet because it was generally known to generate and output error signals, as disclosed in Hazelzet, as part of a “training” sequence. *Id.*; Ex. 1027 at ¶[0091]. Accordingly, Hazelzet least renders obvious “*the module controller generating a notification signal indicating at least one status of one or more training sequences while the memory module is in the second mode and outputting the notification signal to the memory controller of the host system via the open drain output while the memory module is in the second mode,*” as claimed. See *id.*, Ex. 1014 (Hazelzet) at ¶¶[0059], [0069], [0072], [0092], [0109], FIG. 2, FIG. 4B, FIG. 7A.

152. Alternatively, to the extent the Board finds that Hazelzet does not disclose that its notification signal indicates the status of a “training” sequence nor render this claim obvious, one skilled in the art would have nevertheless found this

claim to be obvious over Hazelzet in view of U.S. Patent Publication No. 2008/0155378 to Amidi (“Amidi”), under the broadest reasonable construction of the term “*training*.”

153. Similar to Hazelzet, Amidi teaches a “*second mode*,” i.e., error correction mode where the “memory controller” trains itself using the “error” signals from the “memory module” by learning whether the error was “random” or whether the error was due to “a bad memory location” so that it could remember not to use the bad memory location in the future. Ex. 1015 (Amidi) at ¶[0024] (“The error detection module 158 calculates a register parity bit based on the received address/command signals 142 and compares the calculated register parity bit to the received system parity bit. If the register parity bit does not match the system parity bit, then the error detection module 158 declares an error, and the buffer device 152 transmits an error signal to the memory controller 110 via the terminal error-out 168.”), ¶[0025] (“After receiving the error signal, the memory controller 110 can halt operation of the memory module 150 and retrieve stored address/command signals 142 from the transmission memory 156.”), ¶[0026] (“In one embodiment, the memory controller 110 can retransmit the correct address to the memory module 150 via the second bus 114. After receiving the retransmitted address/command signals 142, the buffer device 152 can perform the error checking again as described above. If no error is found during retransmission, the

memory controller 110 can conclude that the error was random and resume normal operation. If an error is again declared during retransmission, the memory controller 110 can, for example, mark the transmitted address location as a bad memory location, and reprocess the instructions from the processor 102 to derive and transmit a new set of address/command signals 142 corresponding to a different memory location to the memory module 150.”). In other words, Amidi explains that the memory controller “halt(s)” the normal operation of the memory system upon receiving an “error signal” from the memory module. *Id.* Once the normal operation of the memory system in Amidi halts, one skilled in the art would understand that memory system would then enter into a “training” mode from that point on, because Amidi makes it clear that the “memory controller” is trained (*e.g.*, through “retransmission”) once the memory controller halts the operation of the memory module. Ex. 1015 (Amidi) at ¶¶[0025], [0026]. Amidi’s process of determining whether or not there is another error during retransmission is training regardless of whether there happens to be another error or not during retransmission. Ex. 1015 (Amidi) at ¶[0026]. In other words, the purpose of Amidi’s training is to find out whether there is a bad memory location, and if so, where. Ex. 1015 (Amidi) at ¶[0026]. In particular, Amidi explains that the “memory controller” is trained to learn whether the error was “random” or whether the error was due to “a bad memory location” so that it could remember not to use

the bad memory location in the future. *Id.* Such training is done by an iterative process (*e.g.*, “retransmission”), whereby the memory controller 110 retransmits the correct address to the memory module 150 via the second bus 114. Ex. 1015 (Amidi) at ¶[0026]. After the retransmission, the memory module performs the error checking again, and sends a notification signal indicating the status of the training (*i.e.*, (1) whether an error was declared during retransmission and/or (2) whether the initial error was “random” or whether the initial error was due to “a bad memory location” so that it could remember not to use the bad memory location in the future) to the memory controller. Ex. 1015 (Amidi) at ¶[0026]. Amidi’s notification signal is not a polled signal (which was disparaged by the 623 Patent, Ex. 1001 at 4:33-56). In particular, the “address/command signals” from the memory controller cannot be a request (*i.e.*, polling) for a status of the training sequence, because Amidi’s notification is sent to the memory controller only when “the error detection module 158 [of the memory module] declares an error” as opposed to whenever the memory controller sends the “address/command signals” to the memory module. Ex. 1015 (Amidi) at ¶[0026] (“[T]he memory controller 110 can retransmit the correct address to the memory module 150 via the second bus 114. After receiving the retransmitted address/command signals 142, the buffer device 152 can perform the error checking again as described above.”), ¶[0024] (“The error detection module 158 calculates a register parity bit based on the

received address/command signals 142 and compares the calculated register parity bit to the received system parity bit. If the register parity bit does not match the system parity bit, then the error detection module 158 declares an error, and the buffer device 152 transmits an error signal to the memory controller 110 via the terminal error-out 168. If the register parity bit matches the system parity bit, no error is declared.”). Accordingly, Amidi discloses “*notification signal*” even under the narrower construction of this term. *Id.* Further, the “status”³ of Amidi’s

³ Amidi explains that “the memory controller 110 can conclude that the error was random and resume normal operation ... or mark the transmitted address location as a bad memory location.” Ex. 1015 (Amidi) at ¶ [0026]. To the extent the Board concludes that the determination of whether the error was “random” or whether the error was due to “a bad memory location” is made later by the “memory controller” as opposed to by the “memory module” before the notification signal is sent to the memory controller, Amidi nevertheless discloses a memory module sending a notification signal indicating the status of the training under the broadest reasonable construction, because the broadest reasonable construction of “indicating ... status” would include “providing information that can be used to determine the status later when combined with other information ” in addition to “providing the status that has already been determined.”

training sequence, e.g., whether an error was declared during retransmission, Ex. 1015 (Amidi) at ¶¶[0026], [0024], is determined by Amidi's "error detection module 158" (within the memory module) before it is indicated to the memory controller via the notification signal. *Id.* Accordingly, Amidi's notification signal provides a status that has already been determined to the memory controller. "[I]f no error is found during retransmission, the memory controller 110 can conclude that the error was random" and exit the second mode to "resume normal operation." Ex. 1015 (Amidi) at ¶[0026]. However, "if an error is again declared during retransmission, the memory controller 110 can, for example, mark the transmitted address location as a bad memory location." Ex. 1015 (Amidi) at ¶[0026]. Here, the "memory controller" is trained not to use the "bad memory location" in the future (by "mark[ing]") if the error was due to "bad memory" in that location. Ex. 1015 (Amidi) at ¶[0026]. As such, one skilled in the art would have found it obvious to modify Hazelzet's memory system such that the "memory controller" trains itself using the "error" signals from the "memory module," during the ECC mode, by learning whether the error was "random" or whether the error was due to "a bad memory location" so that it could remember not to use the bad memory location in the future, as taught by Amidi. For example, one skilled in the art would have found it obvious to modify Hazelzet's memory system such that, during the "ECC mode," the memory controller trains itself not to use the bad

memory location in the future upon receiving a notification signal (from the “ECC/Parity register”) that indicates the error from the specific memory location was an “uncorrectable error (UE).” Ex. 1014 (Hazelzet) at ¶¶[0059], [0069], [0072], [0092], FIG. 2, FIG. 4B, FIG. 7A; *see also* Ex. 1015 (Amidi) at ¶¶[0024]-[0026]. This would have been an arrangement of old elements (Amidi’s memory system, Hazelzet’s memory system) with each performing the same function it had been known to perform (*e.g.*, error checking/correcting) and yielding no more than what one would expect from such an arrangement (*e.g.*, improved overall system performance and/or reliability). Such a modification would have therefore been obvious.

154. Hazelzet and Amidi are analogous art to the 623 Patent because each is directed to the field of error detection and correction in memory modules, Ex. 1001 (623 Patent) at Abstract, 2:35-43, 9:6-14; Ex. 1014 (Hazelzet) at ¶¶[0020], [0022], [0038], [0049], [0059]-[0062], [0069]-[0070], [0072], [0075], [0081], [0101]-[0102], [0106]-[0107], [0109]; Ex. 1015 (Amidi) at ¶¶[0003], [0013], [0017]-[0026], [0029], [0030], [0036]-[0037], and also because Hazelzet and Amidi are reasonably pertinent to the particular problem the inventor of the 623 Patent was trying to solve (increasing the memory system capacity, performance, and reliability by applying training sequences with errors reported using a shared output, Ex. 1001 (623 Patent) at 9:6-14), *see* Ex. 1015 (Amidi) at ¶¶ [0003], [0018]

(“An output device 120 functioning as a logic OR ties the output ends of the error detection module 158 and the transmission memory 156 together such that the error detection module 158 and the transmission memory 156 share the output terminal error-out 168 to transmit signals to the memory controller 110. The output device 120 can be implemented as an OR gate, a wired-OR gate, an open collector, or other device functioning as a logic OR.”); Ex. 1014 (Hazelzet) at ¶¶[0003], [0018] (“After holding the error line low for only 2 clock cycles, the driver can be disabled and the output permitted to return to an un-driven state (high impedance) thus allowing this line to be shared by multiple modules.”), ¶[0072] (“Two open-drain outputs are available to permit multiple modules to share a common signal line for reporting an error that occurred during a valid command (/CS=low) cycle (consistent with the re-driven signals).”), ¶[0099] (“After holding the error line low for only 2 clock cycles, the driver will be disabled and the output permitted to return to an un-driven state (high impedance) allowing this line to be shared by multiple modules.”), ¶[0109] (“Two open-drain outputs are available to permit multiple modules to share a common signal pin for reporting an error that occurred during a valid command (any /CS=low) cycle (consistent with the re-driven signals).”).

155. A person of ordinary skill in the art would have been motivated to combine Hazelzet and Amidi in this manner for various reasons. For example,

Hazelzet emphasizes the need for “improved overall system reliability” and proposes “error correction” as a solution. Ex. 1014 (Hazelzet) at ¶[0003] (“Thus the emphasis and need for improved overall system reliability is increasing dramatically and requires a comprehensive system solution that includes both a high degree of fault tolerance and overall reliability.”); *see also* Ex. 1014 (Hazelzet) at ¶[0042] (“The ECC/Parity buffer chips 21 on DIMM 20 include unique error correction code (ECC) circuitry that is coupled to the memory interface chip 18 via line 25 to provide even greater significant reliability enhancement to such servers. The inclusion of this new, improved error correction code (ECC) circuitry results in a significant reduction in interconnect failure.”), ¶[0044] (“The novel ECC/ Parity buffer 21 on DIMM 20 thus provides leading-edge performance and reliability and key operational features different from and unavailable from the prior art while retaining timing requirements generally consistent with devices such as the JEDEC 14 bit 1:2 DDR II register.”), Abstract (“A high density high reliability memory module with power gating and a fault tolerant address and command bus.”), ¶[0001] (“This invention relates generally to a high-density, high-reliability memory module with a fault tolerant address and command bus for use as a main memory that will achieve the degree of fault-tolerance and self-healing necessary for autonomic computing systems.”), ¶[0004] (“The present invention provides such a comprehensive system solution that

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

includes the capability of high memory density and a high degree of fault tolerance and the overall differentiated system reliability long desired in the server market.”),

¶[0007] (“The present invention is directed to a high density high reliability memory controller/interface module, provided with a high degree of compatibility with industry standard solutions, capable of meeting the desired density, performance and reliability requirements and interfacing with the presently available memory modules, as well as with existing or enhanced support devices.

The present invention accomplishes all these ends, resulting in a high density and enhanced reliability memory solution at low cost.”). One skilled in the art would

have found it obvious to combine Hazelzet and Amidi because the error correction/training schemes of Amidi enhances “overall system performance” and “accuracy and operability for transmitting data.” Ex. 1015 (Amidi) at ¶[0003]

(“Accordingly, there is a need for detecting and correcting transmission errors between the processor and the memory modules to achieve increased overall system performance.”), ¶[0029] (“One expected advantage of the error detection and correction scheme is that the computing device 100 can have improved accuracy and operability for transmitting data between the processor 102 and the memory module 150. The computing device 100 not only can verify the integrity of transmitted signals but also can correct certain transmission errors. The error correction capability reduces the likelihood that the computer device 100 must be

shut down when an error occurs, and thus improves the operability of the computing device 100.”); *see also* Ex. 1014 (Hazelzet) at ¶[0044] (explaining that its error correction relates to the “address and command data.”). For example, as explained in Amidi, once the memory controller goes through the error correction/training process, the memory controller would know the location of the “bad memory” that it should avoid in the future, thereby enhancing overall system reliability and performance. Ex. 1015 (Amidi) at ¶¶[0024]-[0026]. One skilled in the art would have recognized the error correction scheme of Amidi as merely another obvious variation of Hazelzet, especially given that the same “open drain” (or “open collector”) output configurations are used in both Amidi and Hazelzet to achieve the same purpose (*i.e.*, enhancing accuracy through parity checking and/or error correction). Ex. 1014 (Hazelzet) at ¶[0072]; Ex. 1015 (Amidi) at ¶[0018].

156. Further, given Hazelzet’s emphasis on the need for “improved overall system reliability,” one skilled in the art would have been motivated to combine Hazelzet and Amidi because normal operation could be made more reliable with error correction/training, as described in Amidi. Ex. 1015 (Amidi) at ¶¶[0024]-[0026]; Ex. 1014 (Hazelzet) at Abstract, ¶¶[0001], [0003], [0004], [0007], [0042], [0044].

157. A person of ordinary skill in the art would have been motivated to combine Hazelzet and Amidi also because such a combination would have been

merely a use of prior art techniques for prior art purposes with only expected results. For example, it was generally known to perform error correction, as disclosed in Hazelzet, as part of a “training” sequence, as disclosed in Amidi. See, e.g., Ex. 1015 (Amidi) at ¶¶[0024]-[0026] (explaining that the “memory controller” trains itself using the “error” signals from the “memory module” by learning whether the error was “random” or whether the error was due to “a bad memory location” so that it could remember not to use the bad memory location in the future); see also Ex. 1016 (Talbot) at ¶[0088] (explaining that the “training pattern” is used to “calculate an error sign.”); Ex. 1020 (Hutsell) at Abstract, 6:37-8:6 (explaining that one skilled in the art understood error detection/correction to be a form of “training”), 7:21-52 (explaining that a “training mode” involves processing “errors.”); Ex. 1021 (Lindskog) at Abstract (showing that one skilled in the art referred to the reference signal that is used for error correction as the “training signal.”); Ex. 1027 at ¶[0091]; see also Ex. 1003 at ¶151. Moreover, the combination of Hazelzet and Amidi would have yielded the same result as each of the references. Ex. 1014 (Hazelzet) at ¶¶[0059], [0069]-[0072], FIG. 2, FIGS. 4A-4B, FIG. 8; see also Ex. 1015 (Amidi) at ¶¶[0024]-[0026].

158. Alternatively, to the extent the Board finds that Hazelzet does not discloses that its notification signal indicates the status of a “training” sequence nor render this claim obvious, one skilled in the art would have nevertheless found this

claim to be obvious over Hazelzet in view of U.S. Patent Publication No. 2008/0147897 to Talbot (“Talbot”).

159. In particular, Talbot discloses a “training mode,” in which a “*notification signal indicating ... [the] status of one or more training sequences*” is sent to the memory controller, under the broadest reasonable construction of the terms “notification signal”⁴ and “indicating ... status.”⁵ Ex. 1016 (Talbot) at

⁴ Talbot discloses that “[i]n the training mode, the BCMD signal path is looped back to the CRC signal path.” Ex. 1016 (Talbot) at ¶ [0052]. To the extent the Board concludes that Talbot’s notification signal is a polled signal because it is “looped back,” Talbot nevertheless discloses a “*notification signal*,” as claimed, because under the broadest reasonable construction, the “*notification signal*” can be “in response to polling.”

⁵ Talbot discloses that the memory controller shifts the training pattern data in increments until the looped back training pattern data is “byte-locked” (or “bit-locked”). Ex. 1016 (Talbot) at ¶ [0053]. To the extent the Board concludes that the determination of the status of the training sequence is made later by the “memory controller” (as opposed to by the “buffer unit” before the notification signal is sent to the memory controller), Talbot nevertheless discloses a buffer unit sending a notification signal indicating the status of the training sequence under the broadest

¶[0052] (“In the training mode, the BCMD signal path is looped back to the CRC signal path during even MCLK cycles (block 415), and a training pattern (e.g., 10101010 ...) is output on the CRC path during odd MCLK cycles (block 420). Memory controller 100 drives a training pattern on the BCMD signal path, which is output on the CRC path during the even MCLK cycles (block 425).”), FIGS. 1-11. Talbot further discloses that its “buffer unit 170” outputs that notification signal to the memory controller using the same output (*i.e.*, “CRC path”) that is used to send an error signal (*i.e.*, “CRC signal”) during normal operation (*i.e.*, “read and write operations”). *Id.*; *see also* Ex. 1016 (Talbot) at ¶[0049] (“As described above, the CRC is generated and sent to memory controller 170 [sic] during read and write operations between memory controller 100 and buffer units 170.”). During its “training mode,” Talbot explains that the memory controller monitors the looped back training pattern data (*i.e.*, also referred to as the “CRC information”) and shifts the training pattern data in increments until the looped back training pattern data is “byte-locked” (or “bit-locked”). Ex. 1016 (Talbot) at

reasonable construction, because the broadest reasonable construction of “indicating ... status” would include “providing information that can be used to determine the status later when combined with other information ” in addition to “providing the status that has already been determined.”

¶[0053] (“While monitoring the CRC information, memory controller 100 may shift the training pattern data in one UI increments. If the CRC information is correct, byte-lock is established (block 460).”), ¶[0052] (“Memory controller 100 obtains bit-lock and byte-lock receiving known data patterns on CRC path (block 430). In addition, memory controller 100 adjusts the phase of the BCMD clock signal by adjusting variable phase unit 294, so that buffer unit 170 may obtain bit-lock (i.e., bit alignment) and byte-lock (i.e., byte alignment) on the BCMD signal path (block 435). More particularly, memory controller 1000 may vary (shift) the pattern being sent by one bit time (UI) to ensure that buffer unit 170 is capturing each bit correctly and shifting in the serial bits and capturing entire eight-bit bytes on the correct byte boundary.”). Once there is “byte-lock” (or “bit-lock”), the memory controller determines that the training has ended, and takes the memory system out of the “training mode.” Ex. 1016 (Talbot) at ¶[0052] (“Memory controller may then send a buffer command to take buffer unit 170 out of training mode (block 440).”), ¶[0089] (“Once the periodic training is complete, buffer unit 870 is placed back into normal mode as describe above in block 945.”). As such, one skilled in the art would have found it obvious to modify Hazelzet’s “buffer device” (i.e., “ECC/Parity register,” as explained in Ex. 1014 (Hazelzet) at ¶[0038]) such that it uses a shared notification output for both training and error correction, as taught by Talbot (e.g., “buffer unit 170” including “control logic

255,” as shown in FIG. 2 of Talbot). *See, e.g.*, Ex. 1016 (Talbot) at ¶¶[0052], [0053], [0088], [0089]. Talbot explains that its training involves “training” of the “buffer units 170.” Ex. 1016 (Talbot) at ¶[0032] (“In addition, memory controller 100 may control the configuration, training and testing of the buffer units 170 by sending a variety of loopback commands, CRC control commands, and CRC training pattern commands, for example.”), ¶[0052] (“Memory controller may then send a buffer command to take buffer unit 170 out of training mode (block 440).”), ¶¶[0072], [0089], FIG. 4 (explaining that the “Buffer [is] In Training Mode.”). Accordingly, Talbot discloses “training” even under the narrower construction that excludes “training of a memory controller.” Further, modifying Hazelzet’s “buffer device” in view of Talbot’s “buffer unit 170” would have rendered obvious the claim even under the narrower construction of “training” that is limited to “training of a memory module.” Indeed, Hazelzet explains that Talbot’s CRC is an example of its “error detection and/or correction methods.” Ex. 1014 (Hazelzet) at ¶[0127] (“Any or all of the various elements may include error detection and/or correction methods such as CRC (Cyclic Redundancy Code), EDC (Error Detection and Correction), parity or other encoding/decoding methods suited for this purpose.”). This would have been an arrangement of old elements (Talbot’s memory system, Hazelzet’s memory system) with each performing the same function it had been known to perform (*e.g.*, error checking/correcting) and yielding no more than what

one would expect from such an arrangement (*e.g.*, improved overall system reliability using shared output pins). Such a modification would have therefore been obvious.

160. Hazelzet and Talbot are analogous art to the 623 Patent because each is directed to the field of error detection and correction in memory modules, Ex. 1001 (623 Patent) at Abstract, 2:35-43, 9:6-14; Ex. 1014 (Hazelzet) at ¶¶[0020], [0022], [0038], [0049], [0059]-[0062], [0069]-[0070], [0072], [0075], [0081], [0101]-[0102], [0106]-[0107], [0109], [0127] (explaining that its error detection and/or correction methods include “CRC (Cyclic Redundancy Code)” for multi-bit errors); Ex. 1016 (Talbot) at ¶[0033] (“In one embodiment, a CRC code may be used to provide such multiple bit error detection.”), [0035], [0073], and also because Hazelzet and Talbot are reasonably pertinent to the particular problem the inventor of the 623 Patent was trying to solve (increasing the memory system capacity, performance, and reliability by applying training sequences with errors reported using a shared output, Ex. 1001 (623 Patent) at 9:6-14), *see* Ex. 1016 (Talbot) at ¶¶[0005], [0033] (“When an error is detected on the link in either direction, memory controller 100 may correct the error by retrying the operation.”), [0035], [0052], [0053], [0073]; Ex. 1014 (Hazelzet) at ¶¶[0003], [0018] (“After holding the error line low for only 2 clock cycles, the driver can be disabled and the output permitted to return to an un-driven state (high impedance) thus allowing

this line to be shared by multiple modules.”), ¶[0072] (“Two open-drain outputs are available to permit multiple modules to share a common signal line for reporting an error that occurred during a valid command (/CS=low) cycle (consistent with the re-driven signals).”), ¶[0099] (“After holding the error line low for only 2 clock cycles, the driver will be disabled and the output permitted to return to an un-driven state (high impedance) allowing this line to be shared by multiple modules.”), ¶[0109] (“Two open-drain outputs are available to permit multiple modules to share a common signal pin for reporting an error that occurred during a valid command (any /CS=low) cycle (consistent with the re-driven signals).”).

161. A person of ordinary skill in the art would have been motivated to combine Hazelzet and Talbot in this manner for various reasons. For example, Hazelzet emphasizes the need for “improved overall system reliability” and proposes multiple-bit error detection and “error correction” as a solution. Ex. 1014 (Hazelzet) at ¶[0003] (“Thus the emphasis and need for improved overall system reliability is increasing dramatically and requires a comprehensive system solution that includes both a high degree of fault tolerance and overall reliability.”); see also Ex. 1014 (Hazelzet) at ¶[0042] (“The ECC/Parity buffer chips 21 on DIMM 20 include unique error correction code (ECC) circuitry that is coupled to the memory interface chip 18 via line 25 to provide even greater significant reliability

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

enhancement to such servers. The inclusion of this new, improved error correction code (ECC) circuitry results in a significant reduction in interconnect failure.”), ¶[0044] (“The novel ECC/ Parity buffer 21 on DIMM 20 thus provides leading-edge performance and reliability and key operational features different from and unavailable from the prior art while retaining timing requirements generally consistent with devices such as the JEDEC 14 bit 1:2 DDR II register.”), Abstract (“A high density high reliability memory module with power gating and a fault tolerant address and command bus.”), ¶[0001] (“This invention relates generally to a high-density, high-reliability memory module with a fault tolerant address and command bus for use as a main memory that will achieve the degree of fault-tolerance and self-healing necessary for autonomic computing systems.”), ¶[0004] (“The present invention provides such a comprehensive system solution that includes the capability of high memory density and a high degree of fault tolerance and the overall differentiated system reliability long desired in the server market.”), ¶[0007] (“The present invention is directed to a high density high reliability memory controller/interface module, provided with a high degree of compatibility with industry standard solutions, capable of meeting the desired density, performance and reliability requirements and interfacing with the presently available memory modules, as well as with existing or enhanced support devices. The present invention accomplishes all these ends, resulting in a high density and

enhanced reliability memory solution at low cost.”), ¶[0063] (“The use of this SEC/DED code ensures that all single errors associated with the address and command bits are detected and corrected and that all double bit errors are detected.”). Similarly, Talbot notes the need for error correction as the data rates increases. *See, e.g.*, Ex. 1016 (Talbot) at ¶[0033] (“At high data rates the probability of buffer unit 170 or memory controller 100 receiving a bit error is significant. Accordingly, it may be necessary to protect transfers between memory controller 100 and buffer units 170 with an error detection code that: will robustly detect multiple bit errors within a protected block. In one embodiment, a CRC code may be used to provide such multiple bit error detection.”). One skilled in the art would have found it obvious to combine Hazelzet and Talbot because the error correction/training schemes of Talbot allows the memory system to transmit data accurately and reliably even at a high rate. *Id.*

162. Further, given Hazelzet’s emphasis on the need for “improved overall system reliability,” one skilled in the art would have been motivated to combine Hazelzet and Talbot because normal operation could be made more reliable with error correction/training, as described in Talbot. Ex. 1016 (Talbot) at ¶¶[0033], [0049], [0050], [0073]; Ex. 1014 (Hazelzet) at Abstract, ¶¶[0001], [0003], [0004], [0007], [0042], [0044].

163. A person of ordinary skill in the art would have been motivated to combine Hazelzet and Talbot also because such a combination would have been merely a use of prior art techniques for prior art purposes with only expected results. For example, it was generally known to perform error correction, as disclosed in Hazelzet, as part of a “training” sequence, as disclosed in Talbot. See, e.g., Ex. 1016 (Talbot) at ¶[0088] (explaining that the “training pattern” is used to “calculate an error sign.”); see also Ex. 1015 (Amidi) at ¶¶[0024]-[0026] (explaining that the “memory controller” trains itself using the “error” signals from the “memory module” by learning whether the error was “random” or whether the error was due to “a bad memory location” so that it could remember not to use the bad memory location in the future); Ex. 1020 (Hutsell) at Abstract, 6:37-8:6 (explaining that one skilled in the art understood error detection/correction to be a form of “training”), 7:21-52 (explaining that a “training mode” involves processing “errors.”); Ex. 1021 (Lindskog) at Abstract (showing that one skilled in the art referred to the reference signal that is used for error correction as the “training signal.”); Ex. 1027 at ¶[0091]; see also Ex. 1003 at ¶151.

164. Alternatively, to the extent the Board finds that Hazelzet does not disclose that its “notification signal” indicates the status of a “training” sequence nor render this claim obvious, one skilled in the art would have nevertheless found

this claim to be obvious over Hazelzet in view of U.S. Patent No. 8,139,430 to Buchmann *et al.* (“Buchmann”).

165. Buchmann discloses two modes of operation (*i.e.*, “SBC mode” and “high-speed mode”), Ex. 1017 (Buchmann) at Abstract, 1:39-44, and in particular, discloses executing training and/or initialization sequences in its “SBC mode,” which is also referred to as an “error correcting code protected quasi-static bit communication” mode. Ex. 1017 (Buchmann) at 14:1-15:15; *see also id.* at 3:52-54, 5:6-22, 5:40-12:59. One skilled in the art would have found it obvious to implement the notification techniques of Hazelzet (*i.e.*, using the “ECC/Parity register” to send notification signals indicating the status of the error correction sequence through an open drain output when the memory module is in the “ECC mode”) in the “training” schemes (*e.g.*, “TS3: upstream lane training and repair”) of Buchmann that make use of the “SBC mode” (*i.e.*, error correcting mode). *See, e.g.*, Ex. 1017 (Buchmann) at 8:24-9:17. Indeed, Buchmann explains that its “SBC transmission logic 1116 generates an SBC command,” which not only includes the “ECC check bits” but also “can be a ‘start training state’ command to perform calibration or lane repair” (*i.e.*, “*training*,” as claimed). Ex. 1017 (Buchmann) at 20:20-44; *see also id.* at 15:43-50 (“The SBC FSM and command encoding logic 1118 can determine a particular command to send based upon a sequence of training states used to configure characteristics of devices coupled to the high-

speed link 1106. In the example depicted in FIG. 11, six bits of SBC commands are output to the ECC encoder 1120. The ECC encoder 1120 generates ECC check bits to detect and correct for error conditions.”).

166. In particular, the training and/or initialization sequences taught by Buchmann include “TS0-clock detection, test and repair; TS1-static configuration and calibration; TS2-downstream lane training and repair; TS3-upstream lane training and repair; TS4 packet training; TS5-initiate valid packets; TS6-frame lock; and TS7-read data latency calculation and inter-channel de-skew,” Ex. 1017 (Buchmann) at 5:40-46, FIG. 3, and “SDRAM Interface Initialization” including “memory delay line calibration; DDR3 SD RAM and RCD reset and initialization procedure; DDR3 read data gate delay optimization; read data de-skew; DDR3 read data strobe centering; and write leveling.” Ex. 1017 (Buchmann) at 11:48-53. As such, Buchmann explains that its training involves training of components within the memory module. *Id.*; see also Ex. 1017 (Buchmann) at 5:40-13:67. Accordingly, Buchmann discloses “training” even under the narrower construction of “training” that is limited to “training of a memory module.”

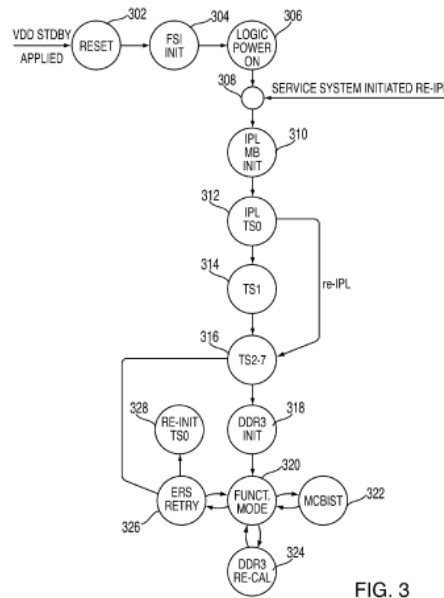


FIG. 3

167. Buchmann discloses that its notifications signals can be both (1) without polling, e.g., Ex. 1017 (Buchmann) at 8:24-9:18, Table 5, Table 6, FIG. 6, or (2) in response to polling, Ex. 1017 (Buchmann) at 12:57-59 (“When a positive write leveling status is polled by system control software, initialization is complete.”). As an example of notification signals without polling, Buchmann explains that during the “upstream lane training” TS3, the memory module generates and outputs various “notification signals” indicating the status of the TS3 training sequence to the memory controller of the host system. Ex. 1017 (Buchmann) at 8:24-9:18, Table 5, Table 6, FIG. 6. As explained in Table 5 of Buchmann, the notification signal “TS3_ack” (e.g., TS3b_ack in Table 6 and FIG. 6) is not in response to polling, because it is notified only when there was an “acknowledgement” as opposed to whenever the TS3_annnc was sent. Buchmann’s

notification signal “TS3_ack” (*e.g.*, TS3b_ack in Table 6 and FIG. 6) also indicates the status of the training sequence by indicating “that all downstream MBs have returned the TS3a_ack SBC.” Ex. 1017 (Buchmann) at Table 5, Table 6, FIG. 6. Further, as explained in Table 6 of Buchmann, the notification signal TS3b_ack, which is a “decoded, corrected” version of TS3a_ack, is “forwarded back to the host.” Ex. 1017 (Buchmann) at Table 5, Table 6, FIG. 6. Accordingly, combination of Hazelzet and Buchmann would have rendered obvious the features related to the “*notification signal*” even under a construction that excludes “polled signal” from the scope of “*notification signal*.” Further, combination of Hazelzet and Buchmann would have rendered obvious the features related to the “*indicating status*” even under a narrower construction of that term, *i.e.*, “providing the status that has already been determined.”

168. Buchmann discloses that its training (*e.g.*, training state zero (TS0) to training state seven (TS7)), Ex. 1017 (Buchmann) at 5:6-22, 5:40-11:44, can be performed in an “SBC mode,” Ex. 1017 (Buchmann) at 3:52-54 (“An exemplary embodiment provides improvements in the overall training sequencing making use of a static bit communication (SBC) mode”). The “SBC mode” is a mode of operation where “error correcting code” is provided, Ex. 1017 (Buchmann) at 14:25-32, to establish “reliable communication” on the bus. Ex. 1017 (Buchmann) at 14:28-35 (“Quasi-static bit communication, also referred to as ‘static bit

communication' or 'SBC', can be used to control the sequence of link initialization and training phases to establish reliable communication on the high-speed bus.

When error correction of an SBC sample is performed prior to checking for a static pattern, the error correction may be able to repair errors that would otherwise prevent the pattern from being static over a period of time.”); *see also id.* at 14:5-8, 3:52-54, 3:64-67.

169. As an example, TS3 involves “upstream lane training” and is “used to perform upstream data lane training,” Ex. 1017 (Buchmann) at 8:24-26, in the SBC mode, Ex. 1017 (Buchmann) at 8:27-47 (showing the SBC commands used during TS3 in Table 5). In particular, the “upstream lane training” TS3 of Buchmann (*i.e.*, “*training sequence*” in the “*second mode*”) is caused by the memory buffer MB (*i.e.*, “*module controller*,” as claimed) by correction/forwarding of the SBC command “TS3_annnc” (*i.e.*, “TS3a announcement SBC”) from the host. Ex. 1017 (Buchmann) at 8:24-27 (“TS3 ... immediately follows TS2 with the host responsible for sending the TS3a announcement SBC.”), Table 5 (“TS3_annnc: This command announces the beginning of TS3a. This SBC is corrected and forwarded by MBs in the memory channel.”). Further, during the “upstream lane training” TS3, the memory module generates and outputs various notification signals indicating the status of the TS3 training sequence to the memory controller of the host system. Ex. 1017 (Buchmann) at 8:24-9:18, Table 5, Table 6, FIG. 6.

170. Buchmann's notification signal indicates the status of the various trainings. For example, as explained in Table 5 of Buchmann, the notification signal "TS3_ack" (e.g., TS3b_ack in Table 6 and FIG. 6) indicates the status of the training sequence by indicating "that all downstream MBs have returned the TS3a_ack SBC." Ex. 1017 (Buchmann) at Table 5, Table 6, FIG. 6. Further, as explained in Table 6 of Buchmann, the notification signal TS3b_ack, which is a "decoded, corrected" version of TS3a_ack, is "forwarded back to the host." Ex. 1017 (Buchmann) at Table 5, Table 6, FIG. 6. Accordingly, combination of Hazelzet and Buchmann would have rendered obvious the features related to the "status ... indicating" even under a narrower construction that excludes "providing information that can be used to determine the status when combined with other information" from the scope of "indicating ... status."

171. Because the "SBC mode" is a mode of operation where "error correcting code" is provided, Ex. 1017 (Buchmann) at 14:25-28, one skilled in the art would have found it obvious to combine the "training" schemes (e.g., "TS3: upstream lane training and repair") described in Buchmann with the notification scheme of Hazelzet in the "ECC mode" (i.e., "second mode," as claimed). Indeed, Buchmann explains that "multi-drop bus(es) or other shared or parallel bus(es)" can be used for such notification, Ex. 1017 (Buchmann) at 28:23-31, and therefore one would have found it obvious to implement such notification using the open

drain outputs in Hazelzet. Ex. 1014 (Hazelzet) at ¶[0072]. As such, one skilled in the art would have found the combination of Hazelzet and Buchmann to be merely an arrangement of old elements with each performing the same function it had been known to perform and yielding no more than what one would expect from such an arrangement.

172. Hazelzet and Buchmann are analogous art to the 623 Patent because each is directed to the field of error detection and correction in memory modules, Ex. 1001 (623 Patent) at Abstract, 2:35-43, 9:6-14; Ex. 1014 (Hazelzet) at ¶¶[0020], [0022], [0038], [0049], [0059]-[0062], [0069]-[0070], [0072], [0075], [0081], [0101]-[0102], [0106]-[0107], [0109], [0127] (explaining that its error detection and/or correction methods include “CRC (Cyclic Redundancy Code)” for multi-bit errors and “parity” for single-bit errors); Ex. 1017 (Buchmann) at 14:22-28, 15:33-35, 15:48-18:24, 28:23-31, 35:8-12 (explaining that its error detection and/or correction methods include “CRC (Cyclic Redundancy Code)” for multi-bit errors and “parity” for single-bit errors), and also because Hazelzet and Buchmann are reasonably pertinent to the particular problem the inventor of the 623 Patent was trying to solve (increasing the memory system capacity, performance, and reliability by applying training sequences with errors reported using a shared output, Ex. 1001 (623 Patent) at 9:6-14), see Ex. 1017 (Buchmann) at 16:46-47 (discussing “ECC protection of the SBC protocol [that] allows error detection and

correction”), 28:23-65 (discussing “multi-drop bus(es) or other shared or parallel bus(es)”); Ex. 1014 (Hazelzet) at ¶¶[0003], [0018] (“After holding the error line low for only 2 clock cycles, the driver can be disabled and the output permitted to return to an un-driven state (high impedance) thus allowing this line to be shared by multiple modules.”), ¶[0072] (“Two open-drain outputs are available to permit multiple modules to share a common signal line for reporting an error that occurred during a valid command (/CS=low) cycle (consistent with the re-driven signals).”), ¶[0099] (“After holding the error line low for only 2 clock cycles, the driver will be disabled and the output permitted to return to an un-driven state (high impedance) allowing this line to be shared by multiple modules.”), ¶[0109] (“Two open-drain outputs are available to permit multiple modules to share a common signal pin for reporting an error that occurred during a valid command (any /CS=low) cycle (consistent with the re-driven signals).”).

173. A person of ordinary skill in the art would have been motivated to combine Hazelzet and Buchmann in this manner for various reasons. For example, Hazelzet emphasizes the need for “improved overall system reliability.” Ex. 1014 (Hazelzet) at ¶[0003] (“Thus the emphasis and need for improved overall system reliability is increasing dramatically and requires a comprehensive system solution that includes both a high degree of fault tolerance and overall reliability.”); see also Ex. 1014 (Hazelzet) at ¶[0042] (“The ECC/Parity buffer chips 21 on DIMM

20 include unique error correction code (ECC) circuitry that is coupled to the memory interface chip 18 via line 25 to provide even greater significant reliability enhancement to such servers. The inclusion of this new, improved error correction code (ECC) circuitry results in a significant reduction in interconnect failure.”), ¶[0044] (“The novel ECC/ Parity buffer 21 on DIMM 20 thus provides leading-edge performance and reliability and key operational features different from and unavailable from the prior art while retaining timing requirements generally consistent with devices such as the JEDEC 14 bit 1:2 DDR II register.”), Abstract (“A high density high reliability memory module with power gating and a fault tolerant address and command bus.”), ¶[0001] (“This invention relates generally to a high-density, high-reliability memory module with a fault tolerant address and command bus for use as a main memory that will achieve the degree of fault-tolerance and self-healing necessary for autonomic computing systems.”), ¶[0004] (“The present invention provides such a comprehensive system solution that includes the capability of high memory density and a high degree of fault tolerance and the overall differentiated system reliability long desired in the server market.”), ¶[0007] (“The present invention is directed to a high density high reliability memory controller/interface module, provided with a high degree of compatibility with industry standard solutions, capable of meeting the desired density, performance and reliability requirements and interfacing with the presently

available memory modules, as well as with existing or enhanced support devices. The present invention accomplishes all these ends, resulting in a high density and enhanced reliability memory solution at low cost.”). Given Hazelzet’s emphasis on the need for “improved overall system reliability,” one skilled in the art would have been motivated to combine Hazelzet and Buchmann because normal operation would be less reliable without the training, as described in Buchmann. Ex. 1017 (Buchmann) at 3:52-54, 14:22-35; Ex. 1014 (Hazelzet) at Abstract, ¶¶[0001], [0003], [0004], [0007], [0042], [0044]. In other words, one skilled in the art would have found it obvious to combine Hazelzet and Buchmann because the training schemes of Buchmann enhances reliability of the overall system. *Id.*

174. A person of ordinary skill in the art would have been motivated to combine Hazelzet and Buchmann also because one skilled in the art would have known that memory systems are initialized before normal operation. Buchmann’s invention is about “initialization training sequence” once the memory system is powered on. Ex. 1017 (Buchmann) at Abstract (explaining that its “memory buffer includes logic for executing a power-on and initialization training sequence initiated by the memory controller.”). Because the memory system in Hazelzet is initialized upon power-on, *see, e.g.*, Ex. 1014 (Hazelzet) at ¶[0123] (“Initialization of the memory subsystem may be completed via one or more methods, based on the available interface busses, the desired initialization speed, available space,

cost/complexity objectives, subsystem interconnect structures, the use of alternate processors (such as a service processor) which may be used for this and other purposes, etc. In one embodiment, the high speed bus may be used to complete the initialization of the memory subsystem(s).”), one skilled in the art would have been motivated to combine Hazelzet and Buchmann. Moreover, Hazelzet expressly suggests using a “high speed bus” to “complete the initialization of the memory subsystem(s).” Ex. 1014 (Hazelzet) at ¶[0123]. Accordingly, one skilled in the art would have been motivated to initialize the memory system of Hazelzet using the initialization/error correction schemes of Buchmann, which are specifically for “high-speed channels.” Ex. 1017 (Buchmann) at 1:63-67 (“The memory system also includes a high-speed channel in communication with the memory controller, the high-speed channel including a plurality of links that are initialized by a plurality of training states.”), 3:30-34 (“FIG. 13 depicts a memory system that includes fully buffered DIMMs communicating via high-speed upstream and downstream busses using error correcting code protected quasi-static bit communication that may be implemented by exemplary embodiments.”), 5:13-16 (“Next, training state one (TS1) is performed at state 314. This includes starting the MB core logic clocks, configuring the MB control and mode registers, and performing DC calibration of the high-speed memory channel.”), 11:48-53 (“SDRAM Interface Initialization. This step follows the memory channel high-

speed link training states and includes the following procedures: memory delay line calibration; DDR3 SDRAM and RCD reset and initialization procedure; DDR3 read data gate delay optimization; read data de-skew; DDR3 read data strobe centering; and write leveling.”).

175. A person of ordinary skill in the art would have been motivated to combine Hazelzet and Buchmann also because such a combination would have been merely a use of prior art techniques for prior art purposes with only expected results. For example, it was generally known to perform error correction, as disclosed in Hazelzet, as part of a “training” sequence, as disclosed in Buchmann. See, e.g., Ex. 1016 (Talbot) at ¶[0088] (explaining that the “training pattern” is used to “calculate an error sign.”); Ex. 1015 (Amidi) at ¶¶[0024]-[0026] (explaining that the “memory controller” trains itself using the “error” signals from the “memory module” by learning whether the error was “random” or whether the error was due to “a bad memory location” so that it could remember not to use the bad memory location in the future); Ex. 1020 (Hutsell) at Abstract, 6:37-8:6 (explaining that one skilled in the art understood error detection/correction to be a form of “training”), 7:21-52 (explaining that a “training mode” involves processing “errors.”); Ex. 1021 (Lindskog) at Abstract (showing that one skilled in the art referred to the reference signal that is used for error correction as the “training signal.”); Ex. 1027 at ¶[0091]; see also Ex. 1003 at ¶151. Further, using the same

output to notify error signals (e.g., CRC signals) and training status was known to one skilled in the art. *See, e.g.,* Ex. 1016 (Talbot) at ¶¶[0052], [0053], FIGS. 1-11; *see also* Ex. 1016 (Talbot) at ¶[0049].

176. Accordingly, for reasons set forth above, Hazelzet discloses this claim element or at least renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann.

d) *Printed Circuit Board Communicating With Memory Controller*

177. Claim 1 requires “*a printed circuit board having a first set of edge connections for communicating address and control signals from the memory controller of the host system, a second set of edge connections for communicating data signals between the memory module and the memory controller of the host system while the memory module operates in the first mode, and an error edge connection coupled to the open drain output of the module controller.*”

178. Hazelzet explains that the “DIMM of the present invention is comprised of a printed circuit board having a front side and a back side and a plurality of double data rate (DDR) DRAMs or synchronous dynamic random access memories (SDRAMs) affixed to both the front surface and the back surface.” Ex. 1014 (Hazelzet) at ¶[0015]; *see also id.*, at FIGS. 3A-3D, ¶¶[0037], [0039] (“Generally speaking DIMMs are printed circuit cards designed to carry a plurality of DRAMs 22 thereon and the DRAM output pins (not shown) are

connected via the printed circuit to selected connectors 23 along the edge of both the back and front sides of the card and are often provided with a single indexing key or notch 9 on the connector edge.”), Abstract (“The memory module includes a rectangular printed circuit board having a first side and a second side, a length of between 149 and 153 millimeters and first and second ends having a width smaller than said length.”).

179. Hazelzet further explains that the printed circuit board has a first set of edge connection for communicating address and control signals from the memory controller of the host system. Ex. 1014 (Hazelzet) at FIGs 7A-7C (showing address edge connections including pins 67, 70, 72-75, 205-207, 210-213, 220 and command edge connections including pins 86, 89, 91, 93), ¶[0072] (referring to “/CS” as commands), ¶[0068] (listing various control signals including “chip select (CS), address (Addr), row address strobe (RAS), column address strobe (CAS) and write enable (WE)” signals); *see also* Ex. 1014 (Hazelzet) at ¶[0038] (“The memory interface chip 18 then sends and receives data, via line 15, to the memory devices or DRAMs 22 and sends address and command information to the register chip 21 via add/cmd line 16 and check bits for error correction purposes to the ECC/ Parity register chip 21 via line 25.”), FIG. 2 (showing address and command line 16).

180. Hazelzet further explains that the printed circuit board has a second set of edge connection for communicating data signals between the memory module and the memory controller of the host system, Ex. 1014 (Hazelzet) at FIGs 7A-7C (showing data edge connections including pins 11, 12, 17, 18, 22, 23, 31, 32, 37, 38), while the memory module is in either the “parity mode” or the “ECC mode,” *id.* at FIG. 8 (showing “Data In” and “Data Out” in both the “parity mode” and the “ECC mode.”).

181. Hazelzet further explains that the printed circuit board has an error edge connection coupled to the open drain output of the module controller. Ex. 1014 (Hazelzet) at FIGs 7A (showing an uncorrectable error pin UE (142) of the memory module that is coupled to the open drain output “/Error (UE)” 121); *see also id.* at FIG. 4B, ¶¶[0072], [0109].

182. Accordingly, Hazelzet discloses “a printed circuit board having a first set of edge connections for communicating address and control signals from the memory controller of the host system, a second set of edge connections for communicating data signals between the memory module and the memory controller of the host system while the memory module operates in the first mode, and an error edge connection coupled to the open drain output of the module controller,” as claimed.

e) Memory Module’s Communication With Memory

Controller In The First and Second Modes

183. Claim 1 requires “*the memory module communicating to the memory controller of the host system via the error edge connection the parity error signal while the memory module operates in the first mode and the notification signal while the memory module is in the second mode.*”

184. As explained in claim element [1.b.], in the “parity mode” (i.e., “first mode,” as claimed), the “parity error signal (PERR)” 111 that is fed into the “error logic circuit” 100 is output to the memory controller through the open-drain output line 121 (i.e., output for uncorrectable error “UE”). Ex. 1014 (Hazelzet) at ¶¶[0059], [0070], [0072], [0076], [0109], FIG. 2, FIG. 4B; *see also* claim element [1.b.]. The open-drain output line 121 is coupled to the memory controller of the host system using error edge connection number 142. Ex. 1014 (Hazelzet) at FIG. 7A.

185. As explained in claim element [1.c.], in the “ECC mode” (i.e., “second mode,” as claimed), a notification signal is output to the memory controller through the open-drain output line and 121 (i.e., output for uncorrectable error “UE”). Ex. 1014 (Hazelzet) at ¶¶[0059], [0069], [0072], [0092], [0109], FIG. 2, FIG. 4B; *see also* claim element [1.c.]. The open-drain output line 121 is coupled to the memory controller of the host system using error edge connection number 142. Ex. 1014 (Hazelzet) at FIG. 7A.

186. Accordingly, Hazelzet discloses “*the memory module communicating to the memory controller of the host system via the error edge connection the parity error signal while the memory module operates in the first mode and the notification signal while the memory module is in the second mode,*” as claimed.

187. Accordingly, Hazelzet renders claim 1 obvious either singly or in combination with Talbot, Amidi, and/or Buchmann.

2. Claim 2 is Unpatentable

188. Claim 2 requires “[t]he memory module of claim 1, further comprising: a plurality of synchronous dynamic random access memory elements, wherein, while the memory module is in the first mode, the module controller receives and processes address and control signals corresponding to read or write commands from the memory controller of the host system and transmits processed address and control signals to the plurality of synchronous dynamic random access memory elements.”

189. As explained in claim element [1.a.], Hazelzet explains that its memory module (DIMM) includes a plurality of “synchronous dynamic random access memories (SDRAMs),” Ex. 1014 (Hazelzet) at ¶[0015], examples of which are shown in FIGS. 2, 3A, 3B, 3C, and 3D. See Ex. 1014 (Hazelzet) at FIGS. 2, 3A-3D; see also *id.* at ¶ [0037], ¶ [0039]. Accordingly, Hazelzet discloses that its

memory module comprises “*a plurality of synchronous dynamic random access memory elements*,” as claimed.

190. As explained in claim element [1.d.], Hazelzet explains that its memory module has connections for communicating address and control signals from the memory controller of the host system. Ex. 1014 (Hazelzet) at FIGs 7A-7C (showing address edge connections including pins 67, 70, 72-75, 205-207, 210-213, 220 and command edge connections including pins 86, 89, 91, 93), ¶ [0072] (referring to “/CS” as commands), ¶ [0068] (listing various control signals including “chip select (CS), address (Addr), row address strobe (RAS), column address strobe (CAS) and write enable (WE)” signals). One skilled in the art would have understood that the “write enable (WE)” is a control signal “*corresponding to read or write commands from the memory controller of the host system*,” as claimed. Hazelzet further discloses driving “the address and command bits ... to the DRAMs from the memory interface chip” in the “parity mode” (i.e., “*first mode*,” as claimed). Ex. 1014 (Hazelzet) at ¶ [0099] (“Parity error reporting in the present invention is realized by delivering the parity signal one cycle after the address and command to which it applies, and driving the error line low, i.e., ‘0’ two clock cycles after the address and command bits are driven to the DRAMs from the memory interface chip if an error is detected. After holding the error line low for only 2 clock cycles, the driver will be disabled and the output permitted to

return to an un-driven state (high impedance) allowing this line to be shared by multiple modules.”). FIG. 2 of Hazelzet discloses that such address and command bits are driven to the DRAMs from the memory interface chip through the “ECC/Parity register” (i.e., “*module controller*,” as claimed) using the address and command line (i.e., “add/cmd line 16”). Ex. 1014 (Hazelzet) at ¶ [0038] (“The memory interface chip 18 then sends and receives data, via line 15, to the memory devices or DRAMs 22 and sends address and command information to the register chip 21 via add/cmd line 16 and check bits for error correction purposes to the ECC/ Parity register chip 21 via line 25.”), FIG. 2. Accordingly, Hazelzet discloses that “*while the memory module is in the first mode, the module controller receives and processes address and control signals corresponding to read or write commands from the memory controller of the host system and transmits processed address and control signals to the plurality of synchronous dynamic random access memory elements,*” as claimed.

191. Accordingly, for reasons set forth above and in claim 1, Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann.

3. Claim 3 is Unpatentable

192. Claim 3 requires “[t]he memory module of claim 1, wherein, while the memory module is in the second mode, the module controller drives the

notification signal to a logic low level via the open drain output to indicate a status of the one or more training sequences.”

193. Hazelzet explains that, while its memory module is in the “ECC mode” (i.e., “second mode,” as claimed), the “ECC/Parity register” (i.e., “module controller,” as claimed) drives the notification signal to a logic low level via the open drain output (i.e., “/ERROR (UE) 121”) to indicate that a “double bit error” (i.e., “uncorrectable error”) has been detected. Ex. 1014 (Hazelzet) at ¶ [0069] (“ECC Mode (/ECC Mode low): ... If a correctable error is detected, /Error (CE) will be driven low for two clocks and errors will be counted and latched in the error bus registers for the 28 inputs if this is the first error since a reset is issued. Any double bit error will also be detected (as well as many other errors that are not correctable), and will be reported on the /Error (UE) error line (driven low for two clocks) and in the error bus registers if this error is the first since a Reset is issued.”), Ex. 1014 (Hazelzet) at ¶ [0059] (“When either [correctable] error line (CE) 109 or uncorrectable error line (UE) 110 is low this indicates that an error was identified as being associated with the address and/or command inputs (either correctable or uncorrectable). The error lines 120, 121 will be active, i.e., low, for two clock cycles simultaneous with the re-driven address/command data when operating in ECC mode or delayed by two clock cycles when operating in parity

mode.”), FIG. 4B; *see also* Ex. 1014 (Hazelzet) at ¶ [0072] (explaining that the error output lines are “open-drain outputs”).

194. Accordingly, Hazelzet discloses “while the memory module is in the second mode, the module controller drives the notification signal to a logic low level via the open drain output to indicate a status of the one or more [error correction] sequences,” as claimed.

195. Accordingly, for reasons set forth above and in claim 1, Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann.

4. Claim 4 is Unpatentable

196. Claim 4 requires “[t]he memory module of claim 1, wherein the first mode is an operational mode of the memory module, and the second mode is a training mode of the memory module.”

197. Hazelzet explains that its “parity mode” (i.e., “first mode,” as claimed) is an “operating mode” with “no correction of errors.” Ex. 1014 (Hazelzet) at ¶ [0064] (“A parity operating mode is also provided, in conjunction with error reporting circuitry to permit the system to interrogate the device to determine the error condition.”), ¶ [0103] (“A parity operating mode is also provided, in conjunction with error reporting circuitry to permit the system to interrogate the device to determine the error condition.”); *see also id.* at ¶ [0020] (“Still further the

present invention permits operation of the memory module in parity mode, such that unused ECC check bit inputs are held at a low level thus ensuring that these inputs are at a known and quiescent state.”); ¶ [0072] (“Error reporting circuitry is included to permit external monitoring of DIMM operation.”); ¶ [0070] (“In addition to the above ECC mode, the same twenty two chip select gated data signals can be operated in parity mode (/ECC Mode high), whereby the signal received on CHK0/Parity in line is received as parity to the register one clock pulse later than the chip select gated data inputs. The received parity bit is then compared to the parity calculated across these same inputs by the register parity logic to verify that the information has not been corrupted. ... No correction of errors will be completed in this mode.”). Accordingly, Hazelzet discloses that its “parity mode” (i.e., “*first mode*,” as claimed) is “*an operational mode of the memory module*” under the broadest reasonable construction of the term “*operational mode*.”

198. In the Institution Decision of the 837 Patent IPR, the Board “determined that it is not necessary to provide an express interpretation of any term of the claims” (including the term “operational mode”). IPR2017-00548, Paper No. 7, at 4. Even if the Board later concludes that the broadest reasonable construction of the term “*operational mode*” requires it to be a “normal mode of operation,” Hazelzet nevertheless discloses or at least renders this claim obvious.

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

In particular, the 623 Patent explains that “memory read/write” is an example of standard operations that are performed in the “normal mode” of operation. Ex. 1001 (623 Patent) at 7:29-40. One skilled in the art would have known that memory systems routinely go through “parity” checking during such “normal operation” involving memory read/writes. *See, e.g.*, U.S. Patent No. 5,606,662 to Wisor (“Wisor”) (Ex. 1022) at 3:60-63 (“During normal operation when the system error parity checking is enabled, upon a write cycle to one of the DRAM banks 106, the write data is buffered through data buffer 112A of memory control unit 102.”); *see also* U.S. Patent No. 7,568,130 to Smith *et al.* (“Smith”) (Ex. 1023) at 4:53-58 (“During these ‘normal’ bus operations, each module still checks the parity of registers within each respective module that the address on the bus pertains to (each module can potentially use the same address sub-space; however, a module acknowledges system requests only if the module’s select line is asserted); Ex. 1015 (Amidi) at ¶¶ [0019]-[0021] (explaining how “the computing device 100 is configured to detect and correct transmission errors of the address/command signals 142 using parity checking” while reading/writing data from the memory module). Accordingly, one skilled in the art would have considered Hazelzet’s “parity mode” (*i.e.*, “*first mode*,” as claimed) to be a mode for “normal operation,” or alternatively, would have found it obvious to try implementing Hazelzet’s “parity mode” in a “normal mode of operation.” *Id.*

199. On the other hand, Hazelzet explains that its “ECC mode” (*i.e.*, “*second mode*,” as claimed) is a mode for “correcting single bit errors while permitting continuous memory operation independent of the existence of these errors.” Ex. 1014 (Hazelzet) at ¶ [0064]. As explained in claim element [1.c.], one skilled in the art would have found it obvious to use Hazelzet’s “error correction” as part of a “training” sequence. *See, e.g.*, EX1027, ¶ [0091]; EX1020, Abstract, 6:37-8:6, 7:21-52; EX1021, Abstract; EX1003, ¶151; *see also* claim element [1.c.]. In addition, one skilled in the art would have found it obvious to implement the notification techniques of Hazelzet in the “training” schemes of Buchmann that make use of the “SBC mode” (*i.e.*, error correcting mode). *See* claim element [1.c.].

200. Accordingly, for reasons set forth above and in claim 1, Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann.

5. Claim 5 is Unpatentable

201. Claim 5 requires “[t]he memory module of claim 1, wherein the one or more training sequences are executed by the module controller in response to a command or a signal received from the memory controller of the host system.”

202. As explained in claim element [1.c.], Hazelzet discloses that the memory module enters into the “ECC mode” (*i.e.*, “*second mode*,” as claimed) in

response to a command from the memory controller of the host system, through the “ECC mode input 123.” Ex. 1014 (Hazelzet) at ¶¶ [0060], [0062], [0069], [0070], FIG. 7A, FIG. 4B, FIG. 2; *see also* claim element [1.c.].

203. Hazelzet further discloses that error correction sequences are executed by the “ECC/Parity register” (*i.e.*, “*module controller*,” as claimed) in the “ECC mode” (*i.e.*, “*second mode*,” as claimed). In particular, Hazelzet teaches that the “error correction code (ECC) logic” (within the “ECC/Parity register”) executes the error correction sequences by identifying and correcting single bit errors on the command or address bus. Ex. 1014 (Hazelzet) at ¶ [0008] (“The register of the present invention has added thereto additional command inputs, error correction code (ECC) logic to identify and correct single bit errors on the command or address bus, and permit continuous memory operation independent of the existence of these errors.”), ¶ [0044] (“The error correction code circuit ECC segment 21b, shown in FIG. 4B, corrects for single bit errors and thus permits continuous memory operation independent of the existence of these errors.”), ¶ [0069] (“ECC Mode (/ECC Mode low): ... This ECC logic will operate across 28 inputs (22 'CS-gated inputs and the 6 check bits) and will correct all single bit errors and detect all double bit errors present on the twenty-two chip select gated data inputs. ... If a correctable error is detected, /Error (CE) will be driven low for two clocks and errors will be counted and latched in the error bus registers for the 28 inputs if this

is the first error since a reset is issued. Any double bit error will also be detected (as well as many other errors that are not correctable), and will be reported on the /Error (UE) error line (driven low for two clocks) and in the error bus registers if this error is the first since a Reset is issued.”).

204. Accordingly, Hazelzet discloses “*the one or more [error correction] sequences are executed by the module controller in response to a command or a signal received from the memory controller of the host system,*” as claimed. Ex. 1014 (Hazelzet) at ¶¶ [0008], [0044], [0060], [0062], [0069], [0070], FIG. 7A, FIG. 4B, FIG. 2.

205. As explained in claim element [1.c.], one skilled in the art would have found it obvious to use Hazelzet’s “error correction” as part of a “training” sequence. *See, e.g.*, Ex. 1027 at ¶ [0091]; Ex. 1020 (Hutsell) at Abstract, 6:37-8:6 (explaining that one skilled in the art understood error detection/correction to be a form of “training”), 7:21-52 (explaining that a “training mode” involves processing “errors.”); Ex. 1021 (Lindskog) at Abstract (showing that one skilled in the art referred to the reference signal that is used for error correction as the “training signal.”); *see also* Ex. 1003 at ¶ 151.

206. Further, as explained in claim element [1.c.], one skilled in the art would have found it obvious to modify Hazelzet’s memory system such that the “memory controller” trains itself using the “error” signals from the “memory

module,” during the ECC mode, by learning whether the error was “random” or whether the error was due to “a bad memory location” so that it could remember not to use the bad memory location in the future, as taught by Amidi. Ex. 1014 (Hazelzet) at ¶¶ [0059], [0069], [0072], [0092], FIG. 2, FIG. 4B, FIG. 7A; *see also* Ex. 1015 (Amidi) at ¶¶ [0024]-[0026]. *See* claim element [1.c.].

207. Further, as explained in claim element [1.c.], one skilled in the art would have found it obvious to modify Hazelzet’s “buffer device” (*i.e.*, “ECC/Parity register,” as explained in Ex. 1014 (Hazelzet) at ¶ [0038]) such that both training and error correction sequences are executed and output using a shared notification output, as taught by Talbot (*e.g.*, “buffer unit 170” including “control logic 255,” as shown in FIG. 2 of Talbot). *See, e.g.*, Ex. 1016 (Talbot) at ¶¶ [0052], [0053], [0088], [0089]. *See* claim element [1.c.].

208. As explained in claim element [1.c.], one skilled in the art would have found it obvious to combine Hazelzet and Buchmann, such that the training sequences of Buchmann are executed in the “ECC mode” (*i.e.*, “*second mode*,” as claimed) of Hazelzet. *See* claim element [1.c.].

209. Accordingly, for reasons set forth above and in claim 1, Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann.

6. Claim 6 is Unpatentable

210. Claim 6 requires “[t]he memory module of claim 1, wherein the module controller comprises one or more integrated circuit.”

211. As explained in claim element [1.b.], Hazelzet’s “ECC/Parity register” (i.e., “module controller,” as claimed) is also referred to as a “buffer device” or a “buffer chip.” Ex. 1014 (Hazelzet) at ¶ [0038]. Hazelzet further explains that the module support devices, including “buffers” (i.e., “module controller,” as claimed), “registers,” and “PLL” may be comprised of “multiple separate chips” or may be “combined onto a single package or even integrated onto a single device.” Ex. 1014 (Hazelzet) at ¶ [0115]; *see also id.* at FIG. 3A-3D, ¶ [0039] (“It should be understood that the PLL chip 24 can be eliminated if its circuitry is provided on the buffer chip 21 or within the same package as the buffer chip 21.”), ¶ [0042]. Accordingly, Hazelzet discloses that “*the module controller comprises one or more integrated circuit,*” as claimed.

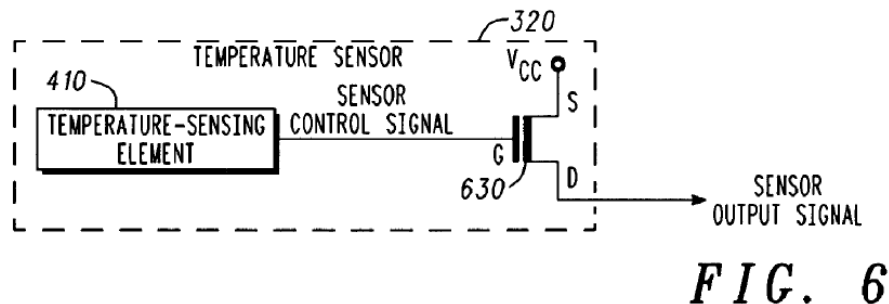
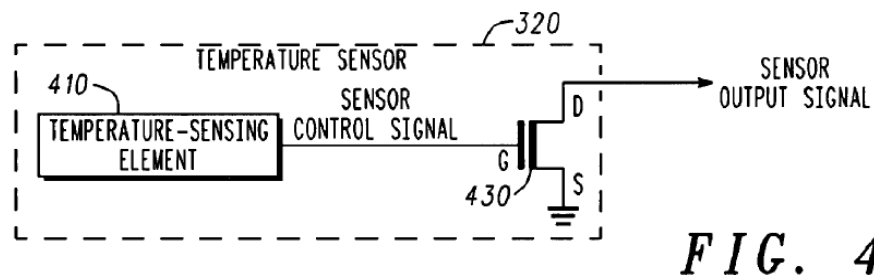
212. Accordingly, for reasons set forth above and in claim 1, Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann.

7. Claim 7 is Unpatentable

213. Claim 7 requires “[t]he memory module of claim 1, wherein the module controller includes a notification circuit comprising one or more transistors each having an open drain coupled to the open drain output.”

214. Hazelzet explains that the “ECC/Parity register” (i.e., “module controller,” as claimed) has an “error reporting circuitry” (i.e., “notification circuit,” as claimed). Ex. 1014 (Hazelzet) at ¶ [0072] (“Error reporting circuitry is included to permit external monitoring of DIMM operation.”), ¶ [0044] (explaining that the “error reporting circuitry” is within the “ECC segment” 21b of the “ECC/Parity buffer”). Hazelzet further explains that such “error reporting circuitry” includes “open-drain outputs.” Ex. 1014 (Hazelzet) at ¶ [0072] (“Two open-drain outputs are available to permit multiple modules to share a common signal line for reporting an error that occurred during a valid command (/CS=low) cycle (consistent with the re-driven signals). These two outputs are driven low for two clocks to allow the memory controller time to sense the error. /Error (CE) indicates that a correctable error occurred and was corrected by the ECC logic, /Error (UE) indicates that an uncorrectable error occurred and depending on the mode selected is an uncorrectable ECC error or a parity error. Note that the timing of /Error (UE) is different in parity mode vs. ECC mode.”).

215. One skilled in the art would have understood that a field effect transistor necessarily has three terminals, *i.e.*, gate, source, and drain. *See generally* U.S. 3,414,781 to Dill (“Dill”) (Ex. 1024), entitled “Field Effect Transistor Having Interdigitated Source And Drain And Overlying, Insulated Gate.” Accordingly, one skilled in the art would understand that a circuit with an “open-drain output,” which is a merely a circuit component with the “drain” terminal “open,” includes “*one or more transistors,*” as claimed. *See, e.g.*, Ex. 1018 (Hokenmaier) at FIG. 4 and FIG. 6.



216. Accordingly, Hazelzet discloses that “the module controller includes a notification circuit comprising one or more transistors each having an open drain coupled to the open drain output,” as claimed.

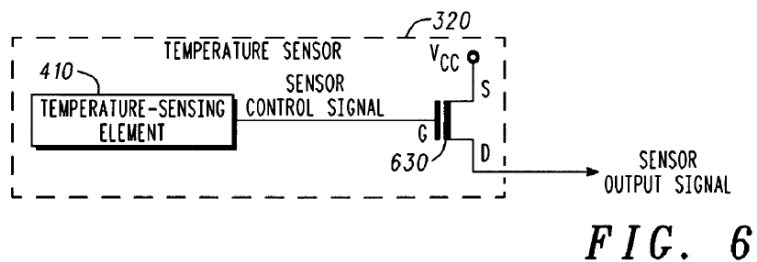
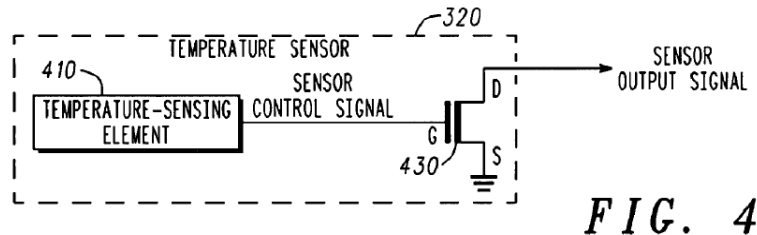
217. As such, for reasons set forth above and in claim 1, Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann.

8. Claim 8 is Unpatentable

218. Claim 8 requires “[t]he memory module of claim 7, wherein the notification circuit generates the notification signal by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.”

219. To the extent the Board finds that Hazelzet does not explicitly disclose driving the notification signal “by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground,” one skilled in the art would have nevertheless found this claim to be obvious in view of Hazelzet (alone or in combination with Talbot, Amidi, and/or Buchmann, for reasons set forth in claim 1). In particular, one skilled in the art would have known that there are only two ways to drive a notification signal using open-drain outputs. The first way, which is shown in FIG. 4 of Hokenmaier, is “by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground” (i.e., using an “open drain pull-down transistor”) and the second way, which is shown in FIG. 6 of

Hokenmaier, is “by driving a gate of each of the one or more transistors to a logic low level such that the open drain of each of the one or more transistors provides a low impedance path to Vcc” (i.e., using an “open drain pull-up transistor”).



220. Having known that there are only two options for driving a signal through an open drain output, one skilled in the art would have found it obvious to drive the notification signal of Hazelzet “by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.” Indeed, using a “pull-down transistor” or a “pull-up transistor” would have been an obvious design choice because there are only two options. See, e.g., Ex. 1018 (Hokenmaier) at 3:52-60, 6:27-8:7, FIGs 4 and 6, 10:66-11:3 (listing “open drain pull-down transistor” and “open drain pull-up transistor” as obvious design choices). One skilled in the art would have found it obvious to try either technique. The use of

either technique with the system of Hazelzet would have been only the use of known technique for its known purpose to achieve predictable results, *i.e.*, either driving a signal low or high. A skilled artisan would have been motivated to use such techniques with Hazelzet because Hazelzet already discloses using open-drain outputs to drive its signals, and it would have been common sense to drive a signal to either a low value or a high value to convey information in a conventional digital circuit.

221. To the extent this claim is not found to be obvious over Hazelzet (alone or in combination with Talbot, Amidi, and/or Buchmann, for reasons set forth in claim 1), one skilled in the art would have found it to be obvious over Hazelzet (alone or in combination with Talbot, Amidi, and/or Buchmann, for reasons set forth in claim 1) in view of Hokenmaier. In particular, Hokenmaier discloses an open-drain output that is configured to generate a signal “*by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground,*” as claimed. See Ex. 1018 (Hokenmaier) at FIG. 4; 6:38-47, 6:50-57 (“Thus, when the sensor control signal indicates that the sensed temperature is above the overheating threshold, the MOS transistor 430 will be turned on and the sensor output signal will be set to ground. But if the sensor control signal indicates that the sensed temperature is not above the over heating threshold, the MOS

transistor 430 will be turned off and the sensor output signal will be allowed to float (*i.e.*, the drain D will be set to have a high impedance).”).

222. One skilled in the art would have been motivated to combine Hazelzet and Hokenmaier, because Hazelzet discloses using “open-drain outputs,” Ex. 1014 (Hazelzet) at ¶ [0072], and Hokenmaier provides a comprehensive explanation as to how “open-drain outputs” can be configured. Ex. 1018 (Hokenmaier) at 3:52-60, 6:27-8:7, FIGs 4 and 6, 10:66-11:3. Further, one skilled in the art would have been motivated to combine Hazelzet and Hokenmaier, because both prior art references provide notification schemes that address the need for connecting multiple notification outputs into a single output line using open-drain outputs, such that a status notification signal from any of the multiple lines would notify the host regarding the overall status of the various components. *See, e.g.*, Ex. 1014 (Hazelzet) at ¶ [0072] (“Two open-drain outputs are available to permit multiple modules to share a common signal line for reporting an error that occurred during a valid command (/CS=low) cycle (consistent with the re-driven signals).”); Ex. 1018 (Hokenmaier) at 1:7-14 (“In particular it relates to a system and method for connecting multiple temperature-sensing signals in an MCP to a single output line such that when any of the multiple temperature sensing signals indicate overheating, the combined sensor output signal will indicate an overheating condition.”).

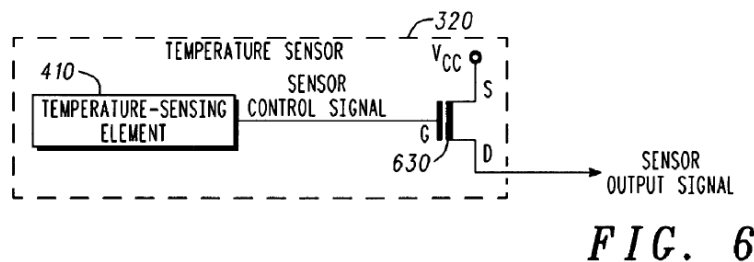
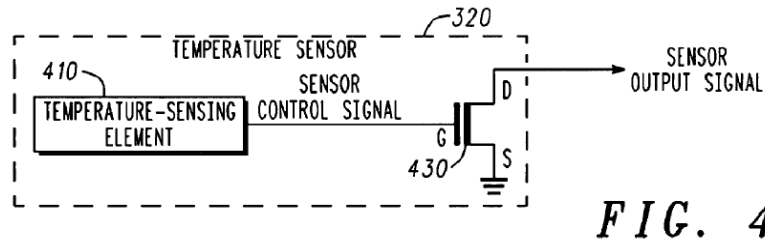
223. Accordingly, for reasons set forth above and in claim 1, Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, Buchmann, and/or Hokenmaier.

9. Claim 9 is Unpatentable

224. Claim 9 requires “[t]he memory module of claim 7, wherein the notification circuit generates the parity error signal by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.”

225. To the extent the Board finds that Hazelzet does not explicitly disclose driving the parity error signal “by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground,” one skilled in the art would have nevertheless found this claim to be obvious in view of Hazelzet (alone or in combination with Talbot, Amidi, and/or Buchmann, for reasons set forth in claim 1). In particular, one skilled in the art would have known that there are only two ways to drive a parity error signal using open-drain outputs. The first way, which is shown in FIG. 4 of Hokenmaier, is “by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground” (i.e., using an “open drain pull-down transistor”) and the second way, which is shown in FIG. 6 of

Hokenmaier, is “by driving a gate of each of the one or more transistors to a logic low level such that the open drain of each of the one or more transistors provides a low impedance path to Vcc” (i.e., using an “open drain pull-up transistor”).



226. Having known that there are only two options for driving a signal through an open drain output, one skilled in the art would have found it obvious to drive the parity error signal of Hazelzet “by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.” Indeed, using a “pull-down transistor” or a “pull-up transistor” would have been an obvious design choice because there are only two options. See, e.g., Ex. 1018 (Hokenmaier) at 3:52-60, 6:27-8:7, FIGs 4 and 6, 10:66-11:3 (listing “open drain pull-down transistor” and “open drain pull-up transistor” as obvious design choices). One skilled in the art would have found it obvious to try either technique. The use of

either technique with the system of Hazelzet would have been only the use of known technique for its known purpose to achieve predictable results, *i.e.*, either driving a signal low or high. A skilled artisan would have been motivated to use such techniques with Hazelzet because Hazelzet already discloses using open-drain outputs to drive its signals, and it would have been common sense to drive a signal to either a low value or a high value to convey information in a conventional digital circuit.

227. To the extent this claim is not found to be obvious over Hazelzet (alone or in combination with Talbot, Amidi, and/or Buchmann, for reasons set forth in claim 1), one skilled in the art would have found it to be obvious over Hazelzet (alone or in combination with Talbot, Amidi, and/or Buchmann, for reasons set forth in claim 1) in view of Hokenmaier. In particular, Hokenmaier discloses an open-drain output that is configured to generate a signal “*by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground,*” as claimed. See Ex. 1018 (Hokenmaier) at FIG. 4; 6:38-47, 6:50-57 (“Thus, when the sensor control signal indicates that the sensed temperature is above the overheating threshold, the MOS transistor 430 will be turned on and the sensor output signal will be set to ground. But if the sensor control signal indicates that the sensed temperature is not above the over heating threshold, the MOS

transistor 430 will be turned off and the sensor output signal will be allowed to float (*i.e.*, the drain D will be set to have a high impedance).”).

228. One skilled in the art would have been motivated to combine Hazelzet and Hokenmaier, because Hazelzet discloses using “open-drain outputs,” Ex. 1014 (Hazelzet) at ¶ [0072], and Hokenmaier provides a comprehensive explanation as to how “open-drain outputs” can be configured. Ex. 1018 (Hokenmaier) at 3:52-60, 6:27-8:7, FIGs 4 and 6, 10:66-11:3. Further, one skilled in the art would have been motivated to combine Hazelzet and Hokenmaier, because both prior art references provide schemes that address the need for connecting multiple outputs into a single output line using open-drain outputs, such that an output signal from any of the multiple lines would inform the host regarding various information (e.g., error). *See, e.g.*, Ex. 1014 (Hazelzet) at ¶ [0072] (“Two open-drain outputs are available to permit multiple modules to share a common signal line for reporting an error that occurred during a valid command (/CS=low) cycle (consistent with the re-driven signals).”); Ex. 1018 (Hokenmaier) at 1:7-14 (“In particular it relates to a system and method for connecting multiple temperature-sensing signals in an MCP to a single output line such that when any of the multiple temperature sensing signals indicate overheating, the combined sensor output signal will indicate an overheating condition.”).

229. Accordingly, for reasons set forth above and in claim 1, Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, Buchmann, and/or Hokenmaier.

10. Claim 10 is Unpatentable

230. Claim 10 requires “[t]he memory module of claim 7, wherein the notification circuit generates the notification signal using one or more OR logic elements.”

231. Claim 10 requires an “open drain” circuit by virtue of its dependency from claim 7. As admitted by Patent Owner’s expert in the IPR for the 837 Patent, which shares the same specification as the 623 Patent, one skilled in the art would have understood that “open drain” signaling required a circuitry with one or more OR logic elements (*e.g.*, “wired-OR”) to send information. Ex. 1026 (Murphy Deposition Transcript in IPR2017-00548) at 75:19-76:1 (“Q If I used the phrase ‘open drain signaling,’ do you have an understanding of what I mean? A I do. Q What do you think it means? A It means the type of circuitry that allows one to create a wired or to send information across a single wire.”). Accordingly, one skilled in the art would have found it obvious to use “one or more OR logic elements” to generate and output a notification signal through an “open drain” circuitry.

232. To the extent the Board finds that Hazelzet does not explicitly disclose that “*the notification circuit generates the notification signal using one or more OR logic elements,*” as claimed, one skilled in the art would have nevertheless found this claim to be obvious in view of Hazelzet, because Hazelzet discloses using “open-drain outputs” as the output of the “*notification circuit [that] generates the notification signal[s]*” for the memory module. Ex. 1014 (Hazelzet) at ¶ [0072]. In particular, Hazelzet explains that its “open-drain output” is “available to permit multiple modules to share a common signal line for reporting an error.” Ex. 1014 (Hazelzet) at ¶ [0072]. A skilled artisan would have understood that the coupling of several memory elements to a single out pin using an open drain connection effectively creates an “OR-gate” such that the pin would be asserted (*i.e.*, active) so long as any of the memory elements so connected to the pin had asserted an active signal on that line. See United States Patent No. 8,359,521 to Kim *et al.* (“Kim”)(Ex. 1019) at FIG. 4, 4:5-18, 4:40-48, 4:50-57, 5:5-11, 5:49-57, 5:65-6:18, 15:7-10; *see also* Ex. 1015 (Amidi) at FIG. 2, FIG. 4, ¶ [0018].

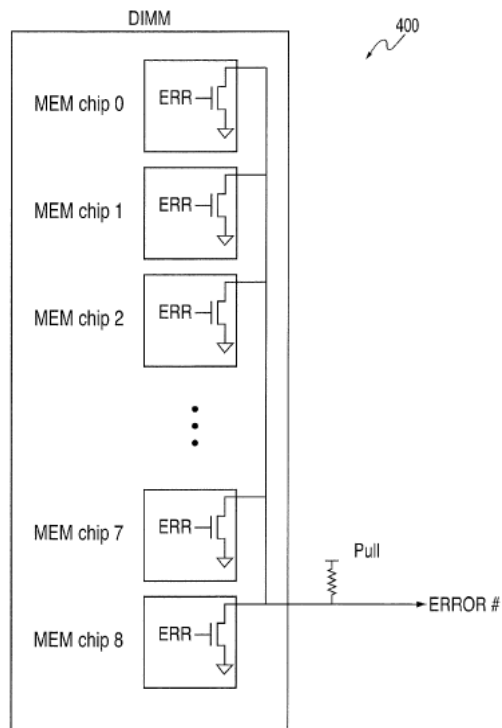


FIG. 4

[FIG. 4 of Kim]

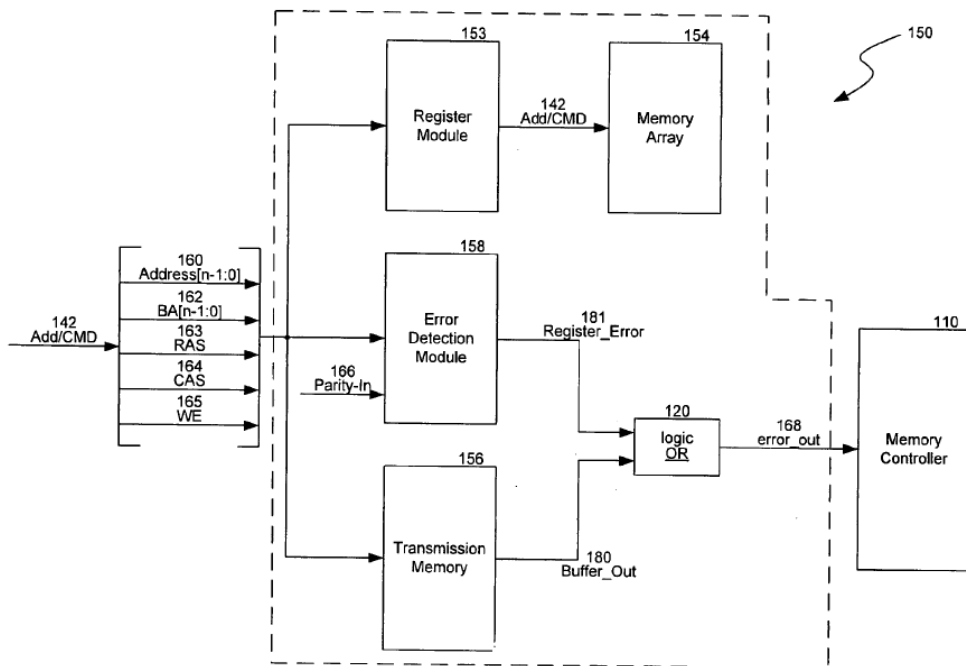


FIG. 2

[FIG. 2 of Amidi]

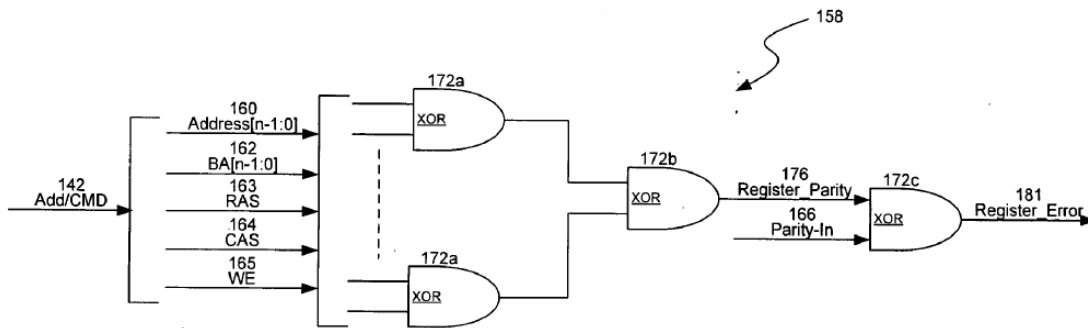


FIG. 4

[FIG. 4 of Amidi]

233. Accordingly, this claim is obvious over Hazelzet (alone or in combination with Talbot and/or Buchmann, for reasons set forth in claim 1).

234. To the extent this claim is not found to be obvious over Hazelzet (alone or in combination with Talbot and/or Buchmann, for reasons set forth in claim 1), this claim would have been obvious over Hazelzet (alone or in combination with Talbot and/or Buchmann, for reasons set forth in claim 1) in view of Amidi. In particular, Amidi explains that an “OR gate” or “logic OR” element can be used interchangeably with an “open collector” (i.e., “open drain”) output. See Ex. 1015 (Amidi) at ¶ [0018] (“An output device 120 functioning as a logic OR ties the output ends of the error detection module 158 and the transmission memory 156 together such that the error detection module 158 and the transmission memory 156 share the output terminal error-out 168 to transmit signals to the memory controller 110. The output device 120 can be implemented

as an OR gate, a wired-OR gate, an open collector, or other device functioning as a logic OR.”); *see also id.* at FIG. 2 (showing that the “error out” signal is output from a “logic OR” element), FIG. 4, claim 17 (“The buffer device of claim 16 wherein the output terminals from the error detection module and the transmission memory are tied using a device functional as a logic OR before being applied to the output terminal for sending the error signal.”). One skilled in the art would have found it obvious to modify Hazelzet’s open-drain output with the teachings of Amidi and would have been motivated to do so for reasons set forth in claim 1.c. *See* claim [1.c.]. Accordingly, this claim is obvious over Hazelzet (alone or in combination with Talbot and/or Buchmann, for reasons set forth in claim 1) in view of Amidi.

235. Alternatively, to the extent this claim is not found to be obvious over Hazelzet (alone or in combination with Talbot, Amidi, and/or Buchmann, for reasons set forth in claim 1), this claim would have been obvious over Hazelzet (alone or in combination with Talbot, Amidi, and/or Buchmann, for reasons set forth above, and in claim 1) in view Kim. As explained above, Kim teaches coupling several memory devices to a single out pin using an open drain connection to effectively create an “OR-gate” such that the pin would be asserted (*i.e.*, active) so long as any of the memory devices so connected to the pin had asserted an active signal on that line. *See* Ex. 1019 (Kim) at FIG. 4.

236. Alternatively, FIG. 5 of Kim shows that the signal from the “open drain output” is generated using an “OR-gate” receiving “parity error” and “CRC error” as input. Ex. 1019 (Kim) at FIG. 5; 6:13-16 (“FIG. 5 is a block diagram of a memory device 500 that shares an error feedback pin between data CRC and address parity in accordance with an exemplary embodiment of the present invention.”).

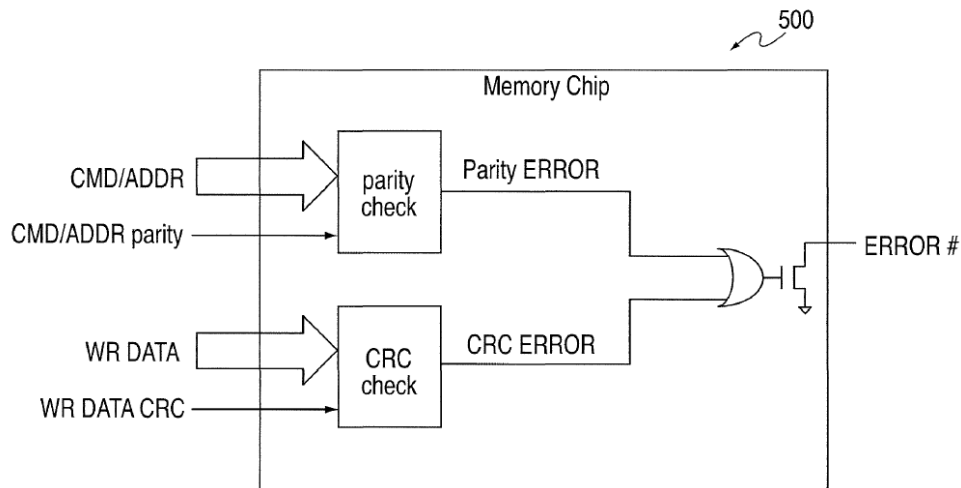


FIG. 5

237. Similar to Hazelzet, Kim teaches the use of a single pin to report errors. Ex. 1014 (Hazelzet) at ¶ [0072]; Ex. 1019 (Kim) at 4:5-18, 4:50-57, 5:5-11, 5:49-57, 5:65-6:18, 15:7-10. Moreover, similar to Hazelzet, Kim also proposes connecting his various memory elements to the output pin using an open drain approach. Ex. 1014 (Hazelzet) at ¶ [0072]; Ex. 1019 (Kim) at 5:49-57; *see also id.* at 4:40-48; 6:1-12; FIG. 4. To use the output-pin scheme of Kim in the system of Hazelzet would therefore have been only the use of known techniques for their

known purposes to achieve predictable results, *i.e.*, using a single line with an “OR logic element” to provide error or status information from multiple components.

238. Kim and Hazelzet are analogous art to the 623 Patent because each is directed to the field of memory systems and error detection/correction therein, Ex. 1001 (623 Patent) at Abstract, 1:43-60, 2:20-34, 9:6-62; Ex. 1014 (Hazelzet) at FIG. 8; *see also id.* at ¶¶ [0020], [0022], [0049], [0059]-[0062], [0064], [0069]-[0070], [0072], [0075], [0081], [0101]-[0103], [0106]-[0107], [0109]; Ex. 1019 (Kim) at Abstract, 1:6-8, 2:7-3:2, 4:50-6:12, FIG. 4, and also because Kim and Hazelzet are reasonably pertinent to the particular problem the inventor of the 623 Patent was trying to solve (making error detection/correction of memory modules more efficient, by using shared lines, Ex. 1001 (623 Patent) at 1:43-3:14, 9:6-26), *see* Ex. 1014 (Hazelzet) at ¶ [0072]; Ex. 1019 (Kim) at Abstract, 5:65-6:3, 6:13-16, 6:19-21, 8:33-38, 12:36-13:38, FIG. 4, FIG. 5, FIG. 6.

239. Moreover, a person of ordinary skill in the art would have been motivated to use the open-drain output pin reporting technique of Kim in the scheme of Hazelzet because it represented a well-known and simply technique to provide error and/or status information from a number of memory components. *See, e.g.*, Ex. 1014 (Hazelzet) at ¶ [0072], ¶ [0122]; Ex. 1019 (Kim) at FIG. 4, 12:18-23. The open-drain interrupt reporting technique of Kim permitted a single signal line with an “OR logic element” to indicate error or status information, thus

simplifying the design of the system and saving pins on the various integrated circuits connected to the bus, which necessarily can have only a finite number of pins. Thus, a person of ordinary skill in the art could readily have modified Hazelzet's open-drain output with the teachings of Kim and would have been motivated to do so.

240. Accordingly, for reasons set forth above and in claim 1, Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, Buchmann, and/or Kim.

11. Claim 11 is Unpatentable

241. Claim 11 requires “[t]he memory module of claim 1, wherein the module controller implements the one or more training sequences in response to a signal or a command received from the memory controller of the host system.”

242. As explained in claim element [1.c.] and claim 5, Hazelzet discloses that the memory module enters into the “ECC mode” (*i.e.*, “*second mode*,” as claimed) in response to a command from the memory controller of the host system, through the “ECC mode input 123.” Ex. 1014 (Hazelzet) at ¶¶ [0060], [0062], [0069], [0070], FIG. 7A, FIG. 4B, FIG. 2; *see also* claim element [1.c.].

243. Hazelzet further discloses that the “ECC/Parity register” (*i.e.*, “*module controller*,” as claimed) implements the error correction sequences in the “ECC mode” (*i.e.*, “*second mode*,” as claimed). In particular, Hazelzet teaches that the

“error correction code (ECC) logic” (within the “ECC/Parity register”) implements the error correction sequences by identifying and correcting single bit errors on the command or address bus. Ex. 1014 (Hazelzet) at ¶ [0008], ¶ [0044], ¶ [0069].

244. Accordingly, Hazelzet discloses that “*the module controller implements the one or more [error correction] sequences in response to a signal or a command received from the memory controller of the host system,*” as claimed. Ex. 1014 (Hazelzet) at ¶¶ [0008], [0044], [0060], [0062], [0069], [0070], FIG. 7A, FIG. 4B, FIG. 2.

245. As explained in claim element [1.c.] and claim 5, one skilled in the art would have found it obvious to use Hazelzet’s “error correction” as part of a “training” sequence. See, e.g., Ex. 1027 at ¶ [0091]; Ex. 1020 (Hutsell) at Abstract, 6:37-8:6 (explaining that one skilled in the art understood error detection/correction to be a form of “training”), 7:21-52 (explaining that a “training mode” involves processing “errors.”); Ex. 1021 (Lindskog) at Abstract (showing that one skilled in the art referred to the reference signal that is used for error correction as the “training signal.”); see also Ex. 1003 at ¶ 151; claim element [1.c.] and claim 5.

246. Further, as explained in claim element [1.c.] and claim 5, one skilled in the art would have found it obvious to modify Hazelzet’s memory system such that the “memory controller” trains itself using the “error” signals from the

“memory module,” during the ECC mode, by learning whether the error was “random” or whether the error was due to “a bad memory location” so that it could remember not to use the bad memory location in the future, as taught by Amidi. Ex. 1014 (Hazelzet) at ¶¶ [0059], [0069], [0072], [0092], FIG. 2, FIG. 4B, FIG. 7A; *see also* Ex. 1015 (Amidi) at ¶¶ [0024]-[0026]. *See* claim element [1.c.] and claim 5.

247. Further, as explained in claim element [1.c.] and claim 5, one skilled in the art would have found it obvious to modify Hazelzet’s “buffer device” (*i.e.*, “ECC/Parity register,” as explained in Ex. 1014 (Hazelzet) at ¶ [0038]) such that both training and error correction sequences are executed and output using a shared notification output, as taught by Talbot (*e.g.*, “buffer unit 170” including “control logic 255,” as shown in FIG. 2 of Talbot). *See, e.g.*, Ex. 1016 (Talbot) at ¶¶ [0052], [0053], [0088], [0089]. *See* claim element [1.c.].

248. As explained in claim element [1.c.], one skilled in the art would have found it obvious to combine Hazelzet and Buchmann, such that the training sequences of Buchmann are implemented in the “ECC mode” (*i.e.*, “*second mode*,” as claimed) of Hazelzet. *See* claim element [1.c.] and claim 5.

249. Accordingly, Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann. *See also* claim 5.

12. Claim 12 is Unpatentable

a) Preamble

250. The preamble of claim 12 requires “[a] memory module to be coupled to a memory controller of a host system, the memory module comprising.”

251. Hazelzet discloses this claim element for the same reasons set forth in claim element [1.a]. See claim element 1.a.

b) SDRAMs

252. Claim 12 requires “a plurality of synchronous dynamic random access memory elements.”

253. Hazelzet discloses this claim element, or at least renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim 2. See claim 2.

c) Printed Circuit Board

254. Claim 12 requires “a printed circuit board (PCB) having a first set of edge connections for communicating address and control signals from the memory controller of the host system, and a second set of edge connections for communicating data signals between the memory module and the memory controller of the host system while the memory module operates in a first mode, the PCB further including an error edge connection in addition to the first set of edge connections and the second set of edge connections.”

255. Hazelzet discloses this claim element, or at least renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim element [1.d]. See claim element 1.d.

d) Module Circuit Controlling SDRAMs In Response To Commands From Memory Controller

256. Claim 12 requires “*a module circuit controlling the plurality of synchronous dynamic random access memory elements in response to read and write commands received from the memory controller of the host system via the first set of edge connections while the memory module operates in the first mode.*”

257. As explained in claim 2, Hazelzet explains that its memory module (DIMM) includes a plurality of “synchronous dynamic random access memories (SDRAMs),” Ex. 1014 (Hazelzet) at ¶ [0015], examples of which are shown in FIGS. 2, 3A, 3B, 3C, and 3D. See Ex. 1014 (Hazelzet) at FIGS. 2, 3A-3D; see also *id.* at ¶ [0037], ¶ [0039].

258. As explained in claim element [1.d.] and claim 2, Hazelzet explains that its memory module has connections for communicating address and control signals “*from the memory controller of the host system*” and that these signals are for “*controlling the plurality of synchronous dynamic random access memory elements.*” Ex. 1014 (Hazelzet) at FIGs 7A-7C (showing address edge connections including pins 67, 70, 72-75, 205-207, 210-213, 220 and command edge connections including pins 86, 89, 91, 93), ¶ [0072] (referring to “/CS” as

commands), ¶ [0068] (listing various control signals including “chip select (CS), address (Addr), row address strobe (RAS), column address strobe (CAS) and write enable (WE)” signals). One skilled in the art would have understood that the “write enable (WE)” is an example of a “*read and write commands received from the memory controller of the host system,*” as claimed. Hazelzet further discloses driving these “address and command bits ... to the DRAMs from the memory interface chip” in the “parity mode” (*i.e.*, “*first mode,*” as claimed). Ex. 1014 (Hazelzet) at ¶ [0099] (“Parity error reporting in the present invention is realized by delivering the parity signal one cycle after the address and command to which it applies, and driving the error line low, *i.e.*, ‘0’ two clock cycles after the address and command bits are driven to the DRAMs from the memory interface chip if an error is detected. After holding the error line low for only 2 clock cycles, the driver will be disabled and the output permitted to return to an un-driven state (high impedance) allowing this line to be shared by multiple modules.”). FIG. 2 of Hazelzet discloses that such address and command signals are driven to the DRAMs from the memory interface chip through the “*module circuit*” (*e.g.*, “SEC/DED ECC 90” and/or “ERROR LOGIC 100” of the “ECC/Parity register”) using the address and command line (*i.e.*, “add/cmd line 16”). Ex. 1014 (Hazelzet) at ¶ [0038] (“The memory interface chip 18 then sends and receives data, via line 15, to the memory devices or DRAMs 22 and sends address and command

information to the register chip 21 via add/cmd line 16 and check bits for error correction purposes to the ECC/ Parity register chip 21 via line 25.”), FIG. 2.

Because there are no address and command signals that are driven to the DRAMs without going through the “*module circuit*,” one skilled in the art would have understood that the “*module circuit*” controls the “*the plurality of synchronous dynamic random access memory elements in response to read and write commands received from the memory controller of the host system*,” as claimed. See Ex. 1014 (Hazelzet) at ¶ [0038], FIG. 2.

259. Accordingly, Hazelzet discloses “*a module circuit controlling the plurality of synchronous dynamic random access memory elements in response to read and write commands received from the memory controller of the host system via the first set of edge connections while the memory module operates in the first mode*,” as claimed. See also claim 2.

e) Module Circuit Generating Parity Error Signal In the First Mode

260. Claim 12 requires “*the module circuit generating a parity error signal indicative of a parity error having occurred in the memory module while the memory module operates in the first mode*.”

261. The “module circuit” in claim 12 is all or part of the “module controller” in claim 1. See dependent claims 16 and 17. Hazelzet discloses this claim element, or at least renders this claim obvious either singly or in combination

with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim element [1.b]. See claim element 1.b.

f) Module Circuit Generating Notification Signal In the Second Mode

262. Claim 12 requires “*the module circuit causing the memory module to enter a second mode in response to a command from the memory controller of the host system, the module circuit generating a notification signal in response to one or more training sequences while the memory module is in the second mode, the notification signal indicating a status of the one or more training sequences.*”

263. The “module circuit” in claim 12 is all or part of the “module controller” in claim 1. See dependent claims 16 and 17. Hazelzet discloses this claim element, or at least renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim element [1.c]. See claim element 1.c.

g) Notification Circuit Of Memory Module Driving Signals To Memory Controller In The First and Second Modes

264. Claim 12 requires “a notification circuit having an open drain output coupled to the error edge connection of the PCB, the notification circuit driving the parity error signal to the memory controller of the host system via the open drain output and the error edge connection while the memory module operates in the first mode, the notification circuit driving the notification signal to the memory

controller of the host system via the open drain output and the error edge connection while the memory module is in the second mode.”

265. The “notification circuit” in claim 12 is part of the “module controller” in claim 1. Hazelzet discloses this claim element, or at least renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim element [1.e] and claim 7. See claim element 1.e. and claim 7.

13. Claim 13 is Unpatentable

266. Claim 13 requires “[t]he memory module of claim 12, wherein the memory module executes the one or more training sequences in response to a signal or a command from the memory controller of the host system.”

267. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claims 5 and 11. See claims 5 and 11; see also claim 12.

14. Claim 14 is Unpatentable

268. Claim 14 requires “[t]he memory module of claim 12, wherein the first mode is an operational mode of the memory module, and the second mode is a training mode of the memory module.”

269. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim 4. See claim 4; see also claim 12.

15. Claim 15 is Unpatentable

270. Claim 15 requires “[t]he memory module of claim 12, wherein the notification circuit generates the notification signal using one or more OR logic elements.”

271. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, Buchmann, and/or Kim, for the same reasons set forth in claim 10. See claim 10; see also claim 12.

16. Claim 16 is Unpatentable

272. Claim 16 requires “[t]he memory module of claim 12, wherein the notification circuit is a physically separate circuit from the module circuit.”

273. Hazelzet explains that the “ECC/Parity register” includes an “error reporting circuitry” (i.e., “notification circuit,” as claimed). Ex. 1014 (Hazelzet) at ¶ [0072] (“Error reporting circuitry is included to permit external monitoring of DIMM operation.”), ¶ [0044] (explaining that the “error reporting circuitry” is within the “ECC segment” 21b of the “ECC/Parity buffer”).

274. One skilled in the art would understand that part of the “ECC/Parity register” (excluding the “error reporting circuitry”) also satisfies the “module

circuit,” as claimed, for reasons set forth in claim 12. Under this mapping, Hazelzet discloses that “the notification circuit is a physically separate circuit from the module circuit,” as claimed. See claim 12; see also Ex. 1014 (Hazelzet) at ¶ [0072], ¶ [0044]. Indeed, Hazelzet explains that the module support devices, including “buffers,” “registers,” and “PLL” may be comprised of “multiple separate chips.” Ex. 1014 (Hazelzet) at ¶ [0115].

275. Accordingly, Hazelzet discloses “*wherein the notification circuit is a physically separate circuit from the module circuit,*” as claimed. As such, for reasons set forth above and in claims 1 and 12, Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann.

17. Claim 17 is Unpatentable

276. Claim 17 requires “[t]he memory module of claim 12, wherein the module circuit includes the notification circuit.”

277. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim 7. See claim 7; see also claim 12.

18. Claim 18 is Unpatentable

278. Claim 18 requires “[t]he memory module of claim 12, wherein the notification circuit includes one or more transistors each having an open drain coupled to the open drain output.”

279. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim 7. See claim 7; see also claim 12.

19. Claim 19 is Unpatentable

280. Claim 19 requires “[t]he memory module of claim 18, wherein the notification circuit generates the notification signal by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.”

281. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, Buchmann, and/or Hokenmaier, for the same reasons set forth in claim 8. See claim 8; see also claim 18.

20. Claim 20 is Unpatentable

282. Claim 20 requires “[t]he memory module of claim 18, wherein the notification circuit generates the parity error signal by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.”

283. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, Buchmann, and/or Hokenmaier, for the same reasons set forth in claim 9. See claim 9; see also claim 18.

21. Claim 21 is Unpatentable

a) Preamble

284. The preamble of claim 21 requires “[a] memory module operable in at least a first mode and a second mode, the memory module comprising.”

285. Hazelzet discloses this claim element, or at least renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim elements [1.a.], [1.b.], and [1.c.]. See claim elements 1.a, 1.b, and 1.c; see also claim 12.

b) SDRAMs

286. Claim 21 requires “a plurality of synchronous dynamic random access memory elements.”

287. Hazelzet discloses this claim element, or at least renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim 2. See claim 2; see also claim 12.

c) Printed Circuit Board Communicating With Memory Controller

288. Claim 21 requires “a printed circuit board (PCB) having a first set of edge connections for communicating address and control signals from the memory controller of the host system, and a second set of edge connections for communicating data signals between the memory module and the memory controller of the host system while the memory module operates in the first mode,

the PCB further including an error edge connection in addition to the first set of edge connections and the second set of edge connections.”

289. Hazelzet discloses this claim element, or at least renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim element [1.d]. See claim element 1.d.; *see also* claim 12.

d) Printed Circuit Board Fits Into Slot Connector Of Host System

290. Claim 21 requires “*wherein the PCB is configured to fit into a corresponding slot connector of the host system.*”

291. Hazelzet discloses this claim element, or at least renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim element [1.a]. See claim element 1.a.; *see also* claim 12.

e) Module Controller Generating/Transmitting Parity Error Signal In The First Mode

292. Claim 21 requires “*a module controller having an open drain output coupled to the error edge connection, the module controller being configured to transmit address and control signals to the plurality of synchronous dynamic random access memory elements in response to read or write commands received from the memory controller of the host system via the first set of edge connections*

while the memory module operates in the first mode, to generate a parity error signal indicating a parity error having occurred in the memory module while the memory module is in the first mode, and to transmit the parity error signal to the memory controller of the host system via the open drain output and the error edge connection while the memory module is in the first mode.”

293. Hazelzet discloses this claim element, or at least renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim elements [1.b], [1.e.], and claim 2. See claim elements 1.b., 1.e., and claim 2; see also claim 12.

f) Module Controller Generating/Transmitting
Notification Signal In The Second Mode

294. Claim 21 requires “*the module controller being further configured to cause the memory module to enter the second mode in response to a command from the memory controller of the host system, to generate a notification signal indicating a status of one or more training sequences while the memory module is in the second mode, and to transmit the notification signal to the memory controller of the host system via the open drain output and the error edge connection while the memory module is in the second mode.*”

295. Hazelzet discloses this claim element, or at least renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for

the same reasons set forth in claim elements [1.c] and [1.e]. See claim elements 1.c. and 1.e.; *see also* claim 12.

22. Claim 22 is Unpatentable

296. Claim 22 requires “[t]he memory module of claim 21, wherein the first mode is an operational mode of the memory module, and the second mode is a training mode of the memory module.”

297. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claims 4 and 14. See claims 4 and 14; *see also* claim 21.

23. Claim 23 is Unpatentable

298. Claim 23 requires “[t]he memory module of claim 21, wherein the memory module executes the one or more training sequences in response to a command or a signal received from the memory controller of the host system.”

299. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claims 5, 11, and 13. See claims 5, 11, and 13; *see also* claim 21.

24. Claim 24 is Unpatentable

300. Claim 24 requires “[t]he memory module of claim 21, wherein the module controller includes a notification circuit comprising one or more transistors, each of the one or more transistors having an open drain coupled to the open drain output.”

301. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claims 7, 17, and 18. See claims 7, 17, and 18; see also claim 21.

25. Claim 25 is Unpatentable

302. Claim 25 requires “[t]he memory module of claim 24, wherein the notification circuit is configured to generate the notification signal by driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.”

303. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, Buchmann, and/or Hokenmaier, for the same reasons set forth in claims 8 and 19. See claims 8 and 19; see also claim 24.

26. Claim 26 is Unpatentable

304. Claim 26 requires “[t]he memory module of claim 24, wherein the notification circuit is configured to generate the parity error signal by [sic] driving a gate of each of the one or more transistors to a logic high level such that the open drain of each of the one or more transistors provides a low impedance path to ground.”

305. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, Buchmann, and/or Hokenmaier, for the same reasons set forth in claims 9 and 20. *See* claims 9 and 20; *see also* claim 24.

27. Claim 27 is Unpatentable

306. Claim 27 requires “[t]he memory module of claim 24, wherein the notification circuit is configured to generate the notification signal using one or more OR logic elements.”

307. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, Buchmann, and/or Kim, for the same reasons set forth in claims 10 and 15. *See* claims 10 and 15; *see also* claim 24.

28. Claim 28 is Unpatentable

308. Claim 28 requires “[t]he memory module of claim 24, wherein the module controller comprises one or more integrated circuits.”

309. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claim 6. *See* claim 6; *see also* claim 24.

29. Claim 29 is Unpatentable

310. Claim 29 requires “[t]he memory module of claim 21, wherein the module controller is further configured to implement the one or more training sequences in response to a signal or a command received from the memory controller of the host system.”

Declaration of Dr. Donald Alpert Regarding U.S. Patent No. 9,535,623

311. Hazelzet renders this claim obvious either singly or in combination with Talbot, Amidi, and/or Buchmann, for the same reasons set forth in claims 5, 11, 13, and 23. *See* claims 5, 11, 13, and 23; *see also* claim 21.

* * *

I, Donald Alpert, do hereby declare and state, that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, under Section 1001 of Title 18 of the United States Code.

Executed on: December 13, 2017


Donald Alpert