

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICRON TECHNOLOGY, INC.,
Petitioner
v.

YANGTZE MEMORY TECHNOLOGIES COMPANY, LTD.,
Patent Owner

Case No. IPR2025-00190
U.S. Patent No. 11,568,941
Issue Date: January 31, 2023

Title: MEMORY INCLUDING A PLURALITY OF PORTIONS AND USED
FOR REDUCING PROGRAM DISTURBANCE AND PROGRAM METHOD
THEREOF

**DECLARATION OF DAVID KUAN-YU LIU, PH.D., IN SUPPORT OF
PETITION FOR INTER PARTES REVIEW OF
U.S. PATENT NO. 11,568,941**

Table of Contents

TABLE OF CONTENTS	II
I. INTRODUCTION	5
II. BACKGROUND AND EXPERT QUALIFICATIONS	6
III. ASSIGNMENT AND MATERIALS CONSIDERED	8
IV. UNDERSTANDING OF THE LAW	10
A. Burden of Proof.....	10
B. Anticipation	11
C. Obviousness.....	11
V. LEVEL OF SKILL IN THE ART	15
VI. TECHNOLOGY BACKGROUND.....	16
VII. THE '941 PATENT	27
A. Effective Filing Date	27
B. Overview	28
C. Prosecution History	35
VIII. CLAIM CONSTRUCTION.....	35
IX. DESCRIPTION OF THE MAIN PRIOR ART REFERENCES.	37
A. The Dong Reference (Ex. 1005)	37
B. The Costa Reference (Ex. 1012)	40
C. The Shirai Reference (Ex. 1018).....	42
X. UNPATENTABILITY ANALYSIS	44
A. Grounds 1 and 2: Claims 1, 3, 5-7, 11, and 13 are Anticipated (Ground 1) and Rendered Obvious (Ground 2) by Dong	45
1. Claim 5	46
2. Claim 6: “The program method of claim 5, further comprising:”	69

3.	Claim 7	71
4.	Claim 1	76
5.	Claim 3 “The memory of claim 1, further comprising”	97
6.	Claim 11	100
7.	Claim 13: “The program method of claim 11, wherein the memory further comprises”	103
B.	Ground 3: Claims 8 and 9 are Rendered Obvious by the Combination of Dong and Costa.....	105
1.	Motivation to Combine Dong and Costa	105
2.	Claim 8: “The program method of claim 7, wherein: [A] the first voltage has a first level if the xth wordline is in the first portion; [B] the first voltage has a second level if the xth wordline is in the second portion; [C] the first voltage has a third level if the xth wordline is in the third portion; and [D] the first level is lower than the second level, and the second level is lower than the third level.”	109
3.	Claim 9: “The method of claim 5, wherein: [A] the first voltage has a first level if the xth wordline is in the first portion; [B] the first voltage has a second level if the xth wordline is in the second portion; and [C] the first level is lower than the second level.”	113
C.	Ground 4: Claims 4, 10, and 14 are Rendered Obvious by the Combination of Dong and Shirai	114
1.	Motivation to Combine Dong and Shirai.....	114
2.	Claim 4: “The memory of claim 1, further comprising: [A] an upper dummy wordline formed below the first portion; [B] a lower dummy wordline formed above the second portion; and [C] a joint oxide layer formed between the upper dummy wordline and the lower dummy wordline; [D] wherein the memory has a two-deck structure.”	119
3.	Claim 10: “The program method of claim 5, wherein the memory further comprises: [A] a lower dummy wordline formed above the first portion; [B] an upper dummy wordline formed below the second portion; [C] a joint oxide layer formed between the lower dummy wordline and the upper dummy wordline; [D] wherein the memory has a two-deck structure.”	122
4.	Claim 14: “[PRE] The program method of claim 11, wherein the memory further comprises: [A] an upper	

dummy wordline formed below the first portion; [B] a lower dummy wordline formed above the second portion; and [C] a joint oxide layer formed between the upper dummy wordline and the lower dummy wordline; [D] wherein the memory has a two-deck structure.” 123

XI. SECONDARY CONSIDERATIONS 124

XII. CONCLUSION 125

I, David Kuan-Yu Liu, Ph.D. declare as follows:

I. INTRODUCTION

1. I have been retained by Micron Technology, Inc. (“Micron”) as an independent expert consultant in this proceeding before the United States Patent and Trademark Office (“PTO”). I am not an employee of Micron or any affiliate or subsidiary of Micron.

2. I have been asked to consider whether certain references disclose or teach the features recited in certain claims of U.S. Patent No. 11,568,941, which I refer to herein as the ’941 patent. In particular, I have been asked to render my opinion regarding the validity of ’941 patent claims 1, 3-11, and 13-14, which I refer to as the “challenged claims.”

3. My opinions and the bases for my opinions are set forth below.

4. I am being compensated at my standard consulting rate for my time. My compensation is based solely on the amount of time that I devote to activity related to this case and is in no way contingent on the nature of my findings, the presentation of my findings in testimony, or the outcome of this or any other proceeding. I have no other financial interest in this proceeding.

II. BACKGROUND AND EXPERT QUALIFICATIONS

5. My *curriculum vitae* (“CV”) is Ex. 1021 and provides an accurate identification of my background, experience, and qualifications as an expert in electronics and NAND memory technology.

6. I received a Bachelor of Science in Electrical Engineering from the University of California, Berkeley, in 1983. I received a Master of Science and Ph.D. degrees in Electrical Engineering from Stanford University in 1985 and 1989, respectively.

7. I worked on a novel Conductivity Modulated Power MOSFET for my Ph.D. thesis. The novel device uses a trench injector to facilitate an optimal tradeoff between the breakdown voltage and ON-Resistance of a Power MOSFET (Vertical DMOS) through the use of a minority carrier injector.

8. From 1989 to 1992, I was a member of technical staff at Texas Instruments, Inc. At Texas Instruments, my job responsibilities included process integration, device modeling, high-voltage CMOS process integration, and investigating a novel source-side injection mechanism for flash EPROM channel hot-electron programming. A source-side injection mechanism is one particular form of a hot carrier injection (HCI) mechanism.

9. From 1992 to 1995, I was a member of the technical staff at Advanced Micro Devices, Inc. (AMD) where I was a key contributor in optimizing flash cell

and periphery devices in AMD's CMOS-based 0.5 μm and 0.35 μm flash EPROM technology. My job responsibilities at AMD also included process integration, device modeling, and development of triple-well process technology to accommodate x-decoder transistors and high voltage transistors for negative gate erase operation. While at AMD, I was awarded a Spotlight Award for developing a method of manufacturing a self-aligned source (SAS) etch for a NOR flash memory.

10. I spent the next five years of my career in managerial and director roles at several California-based semiconductor companies. I was responsible for increasing yield and for leading teams of engineers working to develop next-generation memory devices. This included various different types of memory, including NAND flash memory and embedded flash memory.

11. In 2000, I co-founded Progressant Technologies in Fremont, California. Progressant Technologies developed IP for negative differential resistance transistor technology and was eventually acquired by Synopsys, Inc.

12. From 2000 to 2004, I was a Senior Manager at Xilinx, Incorporated, where I was responsible for developing nonvolatile memory process technology for flash and CPLD product applications, as well as advanced CMOS process technology (specifically 75 nm CMOS technology node, a half node version between 90 nm and 65 nm).

13. From 2004 to 2007, I was a Senior Scientist at Maxim Integrated Products where I was responsible for developing embedded non-volatile memory process technology for power management product applications.

14. Since 2007, I have served as a technical consultant where I have provided expert advice regarding flash memory technology, CMOS process technology, and semiconductor device physics. As part of this, I have considered, assessed, and assisted with the design of various different memory devices, including NAND flash and monolithic 3D memory structures.

III. ASSIGNMENT AND MATERIALS CONSIDERED

15. I have been asked to provide analysis and explain the subject matter of the '941 patent, including the state of the art when the application underlying the '941 patent was filed. I have also been asked to consider, analyze, and explain certain prior art to the '941 patent including how that art relates to the challenged claims of the '941 patent and to provide my opinions regarding whether that art invalidates the claimed subject matter.

16. The opinions expressed in this declaration are not exhaustive of opinions I may offer in the future regarding the unpatentability of the claims of the '941 patent. Therefore, the fact that I do not address a particular point should not be understood to indicate an agreement on my part that any claim complies with the requirements of any applicable rule.

17. I reserve the right to amend and supplement this declaration in light of additional evidence, arguments, or testimony presented during this IPR or related proceedings on the '941 patent.

18. In forming the opinions set forth in this declaration, I have considered and relied upon my education, knowledge of the relevant field, knowledge of scientific and engineering principles, and my experience. I have also reviewed and considered the '941 patent (Ex. 1001), its prosecution history (Exs. 1002 and 1003), and the following additional materials, along with any other materials referenced in the body of my declaration:

Exhibit	Description
1005	U.S. Patent Publication No. 2008/0279008 (“Dong”)
1006	U.S. Patent Publication No. 2008/0159003 (“Dong II”)
1007	U.S. Patent Publication No. 2006/0250850 (“Lee”)
1008	U.S. Patent Publication No. 2018/0190498 (“Lee II”)
1009	Aritome, “NAND Flash Memory Technologies” (2016)
1010	Micheloni et al., “Inside Solid State Drives (SSDs)” (2nd Ed. 2018)
1011	Silvagni, “3D NAND Flash Based on Planar Cells,” Computers 2017
1012	U.S. Patent Publication No. 2018/0012667 (“Costa”)
1013	U.S. Patent Publication No. 2019/0102104 (“Righetti”)
1014	U.S. Patent Publication No. 2012/0003800 (“Lee III”)

Exhibit	Description
1015	U.S. Patent No. 6,859,397 (“Lutze”)
1016	U.S. Patent No. 10,269,570 (“Lee IV”)
1017	U.S. Patent Publication No. 2012/0163096 (“Futatsuyama”)
1018	U.S. Patent Publication No. 2019/0214405 (“Shirai”)
1019	U.S. Patent No. 10,438,671 (“Chen”)
1020	U.S. Patent No. 9,589,978 (“Yip”)
1022	Yeh et al., “Z-Interference and Z-Disturbance in Vertical Gate-Type 3-D NAND,” IEEE Transactions on Electron Devices, Vol. 63, No. 3, March 2016, pp. 1047-1053

IV. UNDERSTANDING OF THE LAW

19. I am not an attorney, but have been instructed in and applied the law as described in this section. I have been informed that each patent claim is considered separately for purposes of patentability and that a dependent claim that depends from another claim includes all of the limitations of the claim it depends from.

A. Burden of Proof

20. I have been informed that the petitioner in an *inter partes* review bears the burden of establishing unpatentability by “a preponderance of the evidence.” I have been further informed and understand that, to prove an assertion by “a preponderance of the evidence,” the party with the burden of proof must demonstrate that it is more likely than not that the assertion is true.

B. Anticipation

21. I have been informed that a patent claim is unpatentable if it is “anticipated” by prior art. I have been informed that a reference anticipates a claim if the reference discloses each limitation of the claim at issue, either expressly or inherently. I have been informed that a prior art reference may anticipate a claim without disclosing a feature of the claimed invention if that missing characteristic is necessarily present, or inherent, in the single anticipating reference as would have been understood by one of ordinary skill in the art.

22. I have been informed that material not explicitly contained in a single prior art document may still be considered for purposes of anticipation if that material is incorporated by reference into the document.

C. Obviousness

23. I have been further instructed and understand that a patent claim is unpatentable and invalid as obvious if the subject matter of the claim as a whole would have been obvious to a POSITA of the claimed subject matter as of the time of the invention at issue. I understand that when assessing the obviousness of claimed subject matter, the following factors are evaluated: (1) the scope and content of the prior art; (2) the difference or differences between each claim of the patent and the prior art; and (3) the level of ordinary skill in the art at the time the patent was filed.

24. I have been informed that a single reference alone can render a patent claim obvious, if any differences between the reference and the claim would have been known or obvious to a POSITA at the time of the invention. That is, if the POSITA could have adapted the reference to meet the claims of the patent by applying known concepts to achieve expected results, then the patent claims are rendered obvious by that reference.

25. I understand that claimed subject matter may be obvious in view of more than one item of prior art. I understand, however, that it is not enough to show simply that all the limitations of the claimed subject matter are spread throughout the prior art. Instead, for claimed subject matter to be obvious over multiple references, there must be some reason or motivation for a POSITA to combine or modify the prior art reference(s) to arrive at the claimed subject matter.

26. I have been informed that, in seeking to determine whether an invention that is a combination of known elements would have been obvious to a POSITA at the time of the invention, one must consider the references in their entirety to ascertain whether the disclosures in those references render the combination obvious to such a person.

27. I have been informed and understand that, while not required, the prior art references themselves may provide a teaching, suggestion, motivation, or reason to combine. The motivation linking two or more prior art references can also come

from other sources, including, for instance, economic or market demand or the common sense of a POSITA at the time of the invention.

28. I understand that a particular combination may be proven obvious by showing that it was obvious to try the combination. I have been informed that, if a technique has been used to improve one device, and a POSITA would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill.

29. I further understand that an obviousness analysis recognizes that market demand, rather than scientific literature, often drives innovation, and that a motivation to combine references also may be supplied by the direction of the marketplace. For example, when there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a POSITA has good reason to pursue the known options within his or her technical grasp because the result is likely the product not of innovation but of ordinary skill and common sense.

30. I have been informed that the combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results. Thus, where all the elements of a claim are used in substantially the same manner, in devices in the same field of endeavor, the claim is likely obvious.

31. Additionally, I understand that a patent is likely to be invalid for obviousness if a POSITA can implement a predictable variation or if there existed at the time of the invention a known problem for which there was an obvious solution encompassed by the patent's claims. Therefore, when a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one.

32. I further understand that combining embodiments related to each other in a single prior art reference would not ordinarily require a leap of inventiveness.

33. I also understand that a POSITA must have had a reasonable expectation of success when combining or modifying references for claimed subject matter to be obvious.

34. I have been informed and I understand that factors referred to as "objective indicia of non-obviousness" or "secondary considerations" are also to be considered when assessing obviousness when such evidence is available. I understand that these factors can include: (1) commercial success; (2) long-felt but unresolved needs; (3) copying of the invention by others in the field; (4) initial expressions of disbelief by experts in the field; (5) failure of others to solve the problem the claimed subject matter solved; and (6) unexpected results.

35. I also understand that evidence of objective indicia of non-obviousness must be commensurate in scope with the claimed subject matter. I further understand

that there must be a relationship, sometimes referred to as a “nexus,” between any such secondary indicia and the claimed invention.

36. Finally, I have been informed that one cannot use hindsight to determine that an invention was obvious.

37. I provide my opinions in this declaration based on the guidelines set forth above.

V. LEVEL OF SKILL IN THE ART

38. I have been informed and understand that the level of ordinary skill in the relevant art at the time of the invention is relevant to inquiries such as the meaning of claim terms, the meaning of disclosures found in the prior art, and the reasons a POSITA may have for combining references.

39. I have been informed and understand that factors that may be considered in determining the level of ordinary skill include: (1) the education of the inventor; (2) the type of problems encountered in the art; (3) prior art solutions to those problems; (4) rapidity with which innovations are made; (5) sophistication of the technology; and (6) education level of active workers in the relevant field. I have been further informed and understand that a person of ordinary skill in the art (POSITA) is also a person of ordinary creativity.

40. In my opinion, a POSITA in the field of the '941 patent would have had a bachelor's degree in electrical engineering or a similar discipline, along with 2-3

years of professional experience working with (e.g., researching, designing, or teaching) NAND flash memory devices (including issues relating to the operation, e.g., programming, of said devices), or an equivalent level of skill, knowledge, and experience (e.g., an advanced degree may replace some of the professional experience).

41. In my opinion, this POSITA would have been aware of and generally knowledgeable about NAND flash memory's structure and fabrication, its component parts, how it operates, and how it is controlled.

42. In view of my educational background and decades of experience working with semiconductor memory—including non-volatile NAND flash memory—I was a person of more than the ordinary level of skill in the art as of October 2019. In forming my opinion, I have drawn on my academic background and professional experiences. My opinions herein, however, were formed taking into account the perspective of an ordinarily skilled artisan.

VI. TECHNOLOGY BACKGROUND

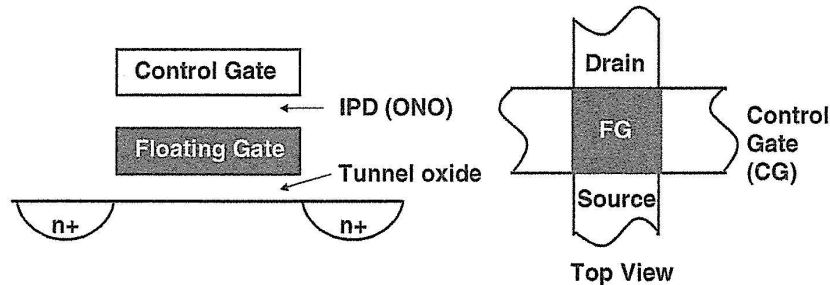
43. This section provides technology background for the '941 patent. The '941 patent and the prior art discussed in this petition relate generally to NAND flash memory. "Flash memory" represents an example of "non-volatile memory." Ex. 1020, 1:24-26. 43. "Non-volatile memory is memory that can retain its data values for some extended period without the application of power." *Id.*, 1:26-28. NAND

flash has been known and in use for quite some time, having been first developed in 1987. *See* Ex. 1009, 22.

44. Like all semiconductor memory, NAND flash includes memory “cells.” In NAND flash, data is stored in a “cell” formed from a single transistor that is the logical equivalent of what a POSITA would have understood to be a “NOT AND” circuit. Ex. 1009, 38-39. This is why this type of memory is referred to as “NAND.” A NAND memory cell can be formed from a “Nch MOS transistor with [a] polysilicon floating gate (FG).” *Id.*, 38-39. “Charge is stored in [the] FG” and the “potential of [the] FG is controlled by a control gate (CG) voltage with capacitive coupling between [the] CG and FG.” *Id.*

45. The presence and the amount of the charge on the floating gate of this type of cell will determine how well the memory cell transistor conducts current under a given control gate voltage. For an Nch memory cell, the more electrons that are stored on the floating gate, the higher the control gate voltage will need to be to turn on the transistor. This control gate voltage that turns on the transistor is known as the threshold voltage.

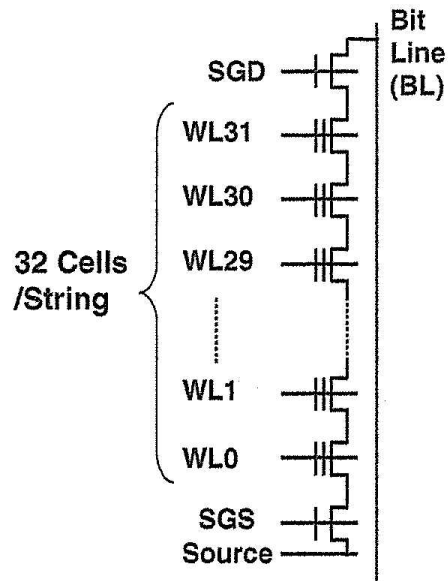
46. The below figure provides an example of one type of flash memory cell:



Id., 39 (Figure 2.1).

47. Other NAND flash employs a transistor with a “charge trap.” *See* Ex. 1010, 109. In this type of transistor, charge is stored in “non-conducting dielectric layer”—like silicon nitride—instead of a floating gate. *See id.*, 109. NAND flash memory cells are arranged in “strings” with multiple cells “connected in series.” Ex. 1009, 38. The cells are surrounded by “two select transistors (SGD, SGS).” *Id.* The “SGD connects” at the string’s drain side “to isolate it from the bit line (BL),” and the “SGS connects at” the string’s source side “to isolate it from a source line (SL).” *Id.*

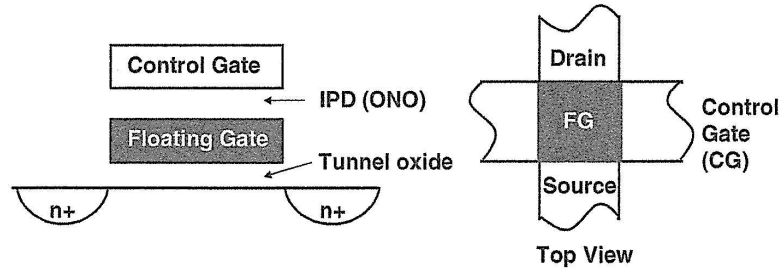
48. Wordlines¹ are connected to the control gate of each memory cell. Ex. 1009, 38-39. An example of one NAND memory string with memory cells arranged in series along the string is shown below:



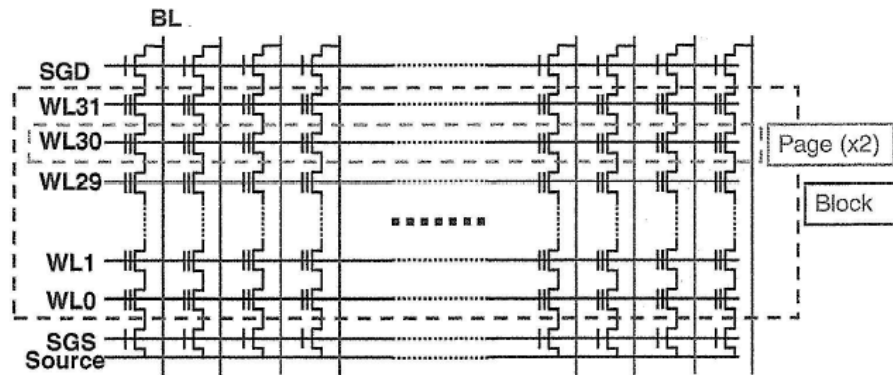
Id., 39.

49. Early NAND flash devices were formed as two-dimensional arrays on a substrate surface. Here, the “memory array is formed by straight active area lines ... and straight gate lines” located on a substrate surface. Ex. 1009, 38. Memory cells are located at the “intersection[s]” of the lines. *Id.* An example of this is shown below, with an example memory cell again shown in the top figure, and a two-dimensional array of such cells shown in the figure below:

¹ To maintain consistency, throughout this declaration I uses the spelling “wordline” as one word instead of “word line,” which is sometimes used in the references.



Id., 30.



Id., 40.

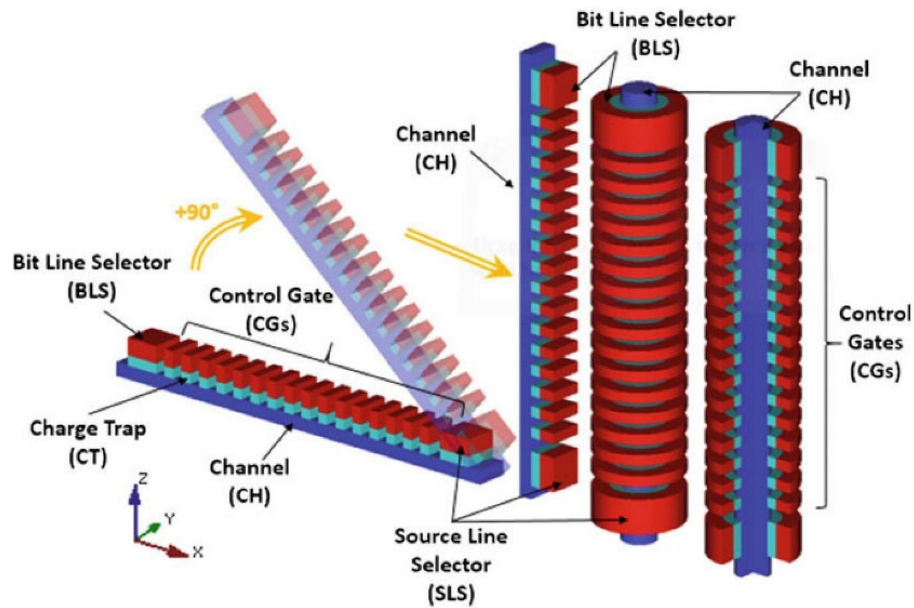
50. As the illustrated two-dimensional array above shows, multiple memory cells are connected in series and sandwiched between the “SDS” and “SGD” select transistors (on the source and drain side respectively), to constitute a NAND string of memory cells. Given that the select transistors, along with a respective data sensing circuitry for each NAND string are a fixed overhead cost, it is advantageous to place as many transistors as possible in a string to increase the capacity and density of the memory, and to amortize this overhead cost across as many memory cells as possible. In this manner, bit density per a given area can be

maximized while the bit cost per memory cell is minimized. *See generally* Ex. 1009, §§7.1-7.2

51. The demand for higher density and lower bit cost in NAND flash memory have largely been insatiable as such devices have gained traction in the marketplace for mass data storage applications and various consumer electronic products, like laptop computers. *See* Ex. 1022. In order to sustain the trend of and meet demand for reduced bit cost and increased the bit density, the linear shrink of the manufacturing patterns have been aggressively driven by adopting advanced fabrication process technologies and lithographic tools, along with storing multiple bits of data in a single memory cell (call multi-level-cell (“MLC”) technology). However, the linear scaling of the NAND Flash memories approached the physical, electrical and reliability limitations as the technology nodes advanced to around 15-30 nm dimensions.

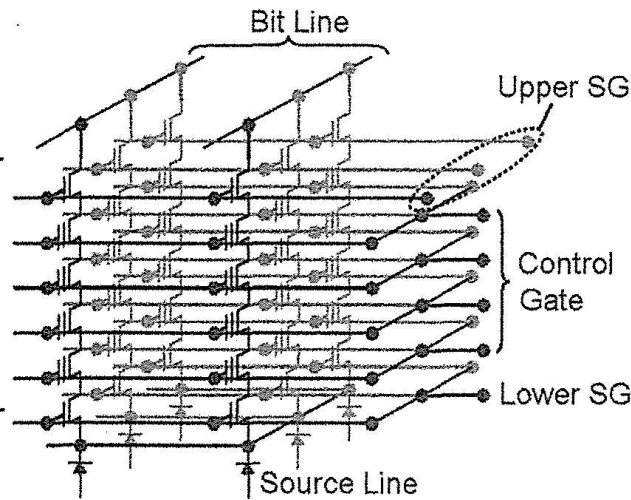
52. “In order to further scale down the memory cell size of NAND flash memory,” the art developed so-called “three-dimensional (3D) NAND flash cells” starting in about 2006. Ex. 1009, 292. In 3D NAND, rather than arranging memory cells in a two-dimensional array, 3D NAND arranges cells in a three-dimensional array that is built up in layers. A POSITA would have known that “[c]onceptually, a modern 3D NAND structure could be conceived by starting from the 2D NAND string building block” and “turning the 2D NAND string upside,” “resulting” in a

“3D structure” with a “vertical channel.” Ex. 1011, 3. The following image provides a demonstrative example of the relationship between the structure of a 2D NAND string and a 3D NAND string:



Ex. 1010, 122.

53. Multiple such vertical channels are then repeated in a three-dimensional array:



Ex. 1009, 296.

54. “Production of 3D NAND flash memory ... started in August 2013.”

Ex. 1009, 393. Given that the origins of 3D NAND came from 2D NAND, there are many similarities between 2D and 3D NAND flash memory devices, including basic principles of operation, memory storage techniques, the techniques and algorithms used to read, erase, and program memory cells and memory cell strings. In this context, the term “program” refers to writing/storing/saving data to a memory cell.

55. Memory cells are frequently organized into “blocks,” “pages,” or other groupings that are used for reading, writing, and erasing operations. “[R]ead and program operations are performed per page” while “the erase operation is performed per block.” Ex. 1009, 39.

56. As the illustration for the 3-D NAND flash array shows, the strings of memory cells are placed in the vertical direction. In contrast, the wordlines (which

function as the memory cell control gates), are horizontal conductive planes which intersect numerous memory strings at the same cell location in each string. Thus, any voltage that is applied on a given wordline will be applied as the control gate voltage to all the memory cells intersecting that plane.

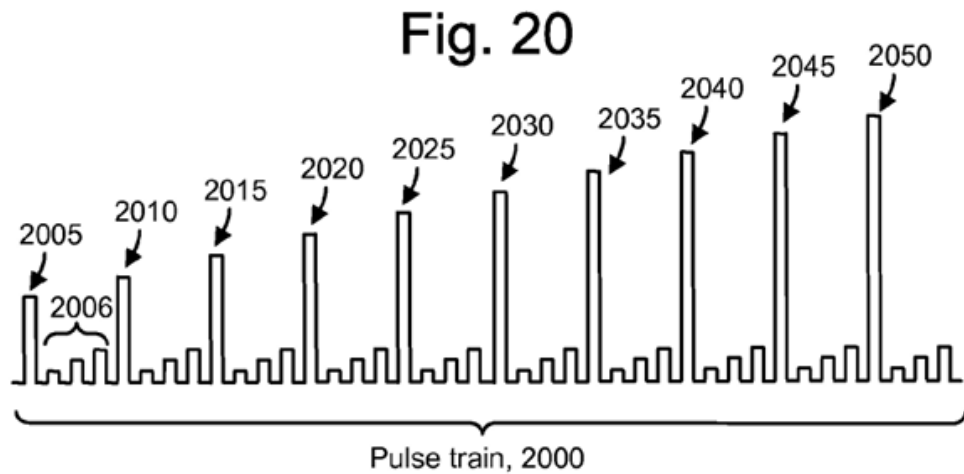
57. To program a particular NAND memory cell in a memory string, a “high-voltage V_{pgm} ” is applied to the wordline connected to the cell, which is often referred to as the “selected” wordline, while “keeping” the string’s bit line at “0V.” Ex. 1009, 44. The other unselected cells in this selected memory string will have a passing voltage, V_{pass} , lower in value than the programming voltage V_{pgm} , being applied to their wordlines to allow the 0V to be passed to the channel of the selected memory cell. Application of this program voltage causes electron movement into the cell and raises the cell’s threshold voltage to a value within a specific range (or one of several different specific ranges) for purposes of storing data. *Id.*, 45.

58. Because the selected wordline to which this program voltage is applied also passes through cells in many other strings, an “inhibit” voltage is applied to these strings’ bit lines to prevent undesired programming. Ex. 1009, 46.

59. As mentioned earlier, a lower “ V_{pass} ” voltage is also applied to all unselected wordlines (Ex. 1009, 44-46) to pass the 0V from the bit line to facilitate the programming of the selected cell. Due to the close proximity of memory cells, along with the application of these various different voltages during the

programming operation, NAND is subject to a well-known “phenomenon” referred to as “[p]rogram disturb.” *Id.*, 52.

60. During a typical programming operation, the memory cells are programmed according to a wordline programming order as the operation is performed from one end of the string to the other end. NAND memory programming typically involves iteratively repeating two related, but distinct, operations. These operations are applying a programming voltage (sometimes called a “pulse”) to actually programming the memory cell by transferring charge to it to change its threshold voltage. The other phase of a typical programming operation is the verify phase where the memory cell being programmed is read to determine it has been sufficiently programmed (*i.e.*, its threshold voltage has moved to the desired level). If not, additional programming pulses are applied, which may increase in intensity with each iteration. As such, the programming operation iterates through alternating programming pulses and verification/read operations. Many references describe this process, including Dong Figure 20 below. Dong (Ex. 1005), [0132] (describing incremental increasing pulses 2005, 2010, etc. interleaved with verify pulses 2006).



61. “Program disturb” involves “the threshold voltage ... of unselected cells” undesirably “increas[ing] during program operation.” Ex. 1009, 52-53. This is problematic, as cells not identified for programming modification may nonetheless enter a different/unintended programmed state due to the voltages applied to the other, adjacent lines. *Id.* Program disturb has been a concern in flash memory devices since the days of 2D NAND, and many techniques and approaches to mitigate its effects have been developed over the years. Because many of the same fundamental principles of device physics are at play in both 2D and 3D NAND, many techniques for mitigating program disturb originally developed for 2D NAND are of equal applicability to 3D NAND devices.

62. “Program disturb” can occur both in the NAND string that includes the programmed cell and in adjacent strings. Ex. 1009, 52-53. It was known that applied voltages can be tailored to either avoid or minimize program disturb. *Id.* For

instance, increasing the Vpass voltage applied to unselected wordlines increases the so-called “boosting” voltage in adjacent NAND strings—a voltage that is induced in part by wordline voltage—thereby reducing program disturb in those strings. *Id.* But this also has the effect of increasing program disturb in the string that includes the programmed cell. *Id.* A lower Vpass has the opposite effect, reducing program disturb in the string that includes the programmed cell, but increasing it in adjacent strings. *Id.*

VII. THE '941 PATENT

A. Effective Filing Date

63. I have been informed that a U.S. patent may claim “priority” to the filing date of one or more earlier filed applications. I have been informed that the priority date, also called the “effective filing date,” is the earliest filing date to which a claim is entitled – and that such date is relevant for determining what qualifies as prior art. I understand that a claim must have written description support in the application(s) to which it claims priority to be entitled to an earlier effective filing date. I have been informed that the priority date determination is done on a claim-by-claim basis, and that claims in the same patent may have different priority dates.

64. I have been instructed by counsel for Petitioner to use February 10, 2020 (the “Priority Date”) as the priority date for all challenged claims of the '941 patent. My opinions in this declaration were formed from the perspective of a

POSITA as of the Priority Date, including both the knowledge of a POSITA at that time as well as how a POSITA would have understood the prior art.

B. Overview

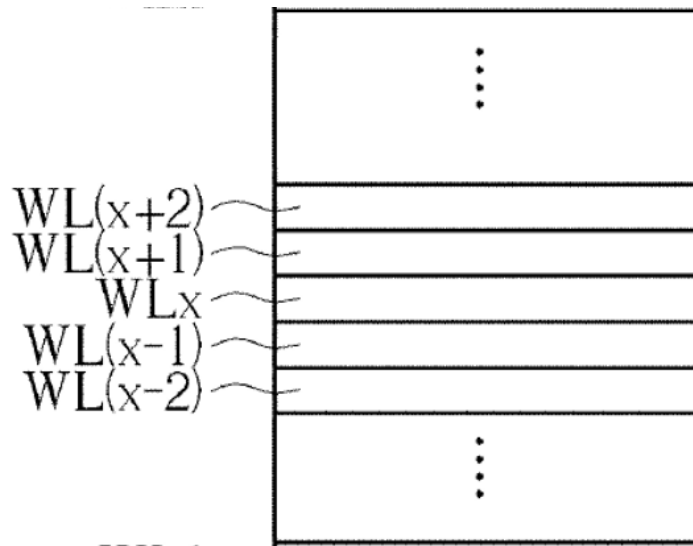
65. The '941 patent is titled “Memory Including a Plurality of Portions and Used for Reducing Program Disturbance and Program Method Thereof.” Ex. 1001, Cover. The U.S. application that resulted in the '941 patent was filed on September 10, 2021 based on a PCT application filed February 20, 2020. *Id.*

66. The specification of the '941 patent discusses “[p]rogram disturbance” and “pass voltage disturbance” problems that “lead to a higher failure rate of programming a memory.” *Id.*, 1:34-38. It is worth noting that pass voltages are applied to the cell memory string during both the programming pulse cycle and the verification pulse cycle of the programming operation.

67. To supposedly try and address these challenges, the '941 specification proposes applying various voltages to particular wordlines when programming the cells in the memory string. *See generally* Ex. 1001. I note that the '941 specification fails to disclose any actual voltage levels to be applied to the wordlines as part of its alleged solution. *Id.* Even relative voltage levels are only given in rare instances. *See, e.g., id.*, 1:66-67 (“The fifth voltage is lower than the fourth voltage.”). It is telling that there are no voltages identified (even relative voltages) in the specification for the memory cells that are directly adjacent to the cell being

programmed. Consistent with this lack of disclosure, the '941 specification mentions that “experiments” and “simulations” were performed, but says nothing about what the experiments and simulations were or any results therefrom. *Id.*, 6:39-43. More generally, the '941 specification gives no explanation at all of how its programming method(s) allegedly reduce program disturb and pass voltage disturbance. *See generally* Ex. 1001.

68. It is helpful to understand how the '941 patent identifies wordlines before discussing the topic of programming voltages. In the jargon of the '941 patent, the memory cell being programmed is identified by the index “x,” and “WLx” refers to the wordline for that cell. *Id.*, Abstract, 3:9-11. Other wordlines are identified relative to their location of the memory cell “x.” As an example, WL(x+1) and WL(x-1) refer to the wordlines immediately above and below WLx, respectively. *See id.*, Figures 1-5, and 7-11. This pattern continues for other wordlines such as WL(x+2) and WL(x-2), as an excerpt of Figure 1 below shows.



Ex. 1001, Figure 1 (excerpted and annotated).

69. The '941 patent uses a different numbering approach to identify what it calls “portions” of a memory string (along with the corresponding wordlines for any cells that are part of that “portion”). The '941 patent illustrates a so-called “portion” as a contiguous group of memory cells (and their corresponding wordlines). *See id.*, Figures 1-11. To show what “portion” a wordline belongs to, the '941 patent uses the variables “k” and “m” (and in some embodiments “n”). *Id.* As with the variable “x,” an increment of “+1” identifies the wordline immediately above “k,” “m,” etc., and decrement of “-1” to denote the wordline immediately below. *See id.*, Figures 1-11. As an example, in the excerpted version of Figure 1 below, the “first portion 110” “include[s] a first wordline WL1 to a kth wordline” and “second portion 120” includes “wordline $WL(k+1)$ to an mth wordline WLm .” *Id.*, 2:63-3:3.

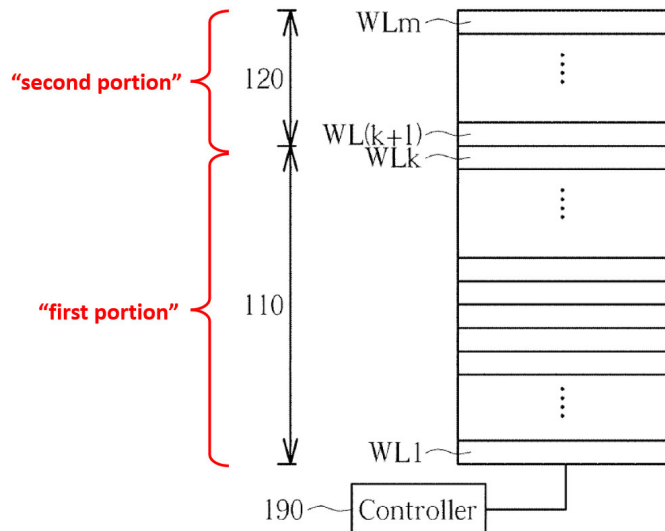
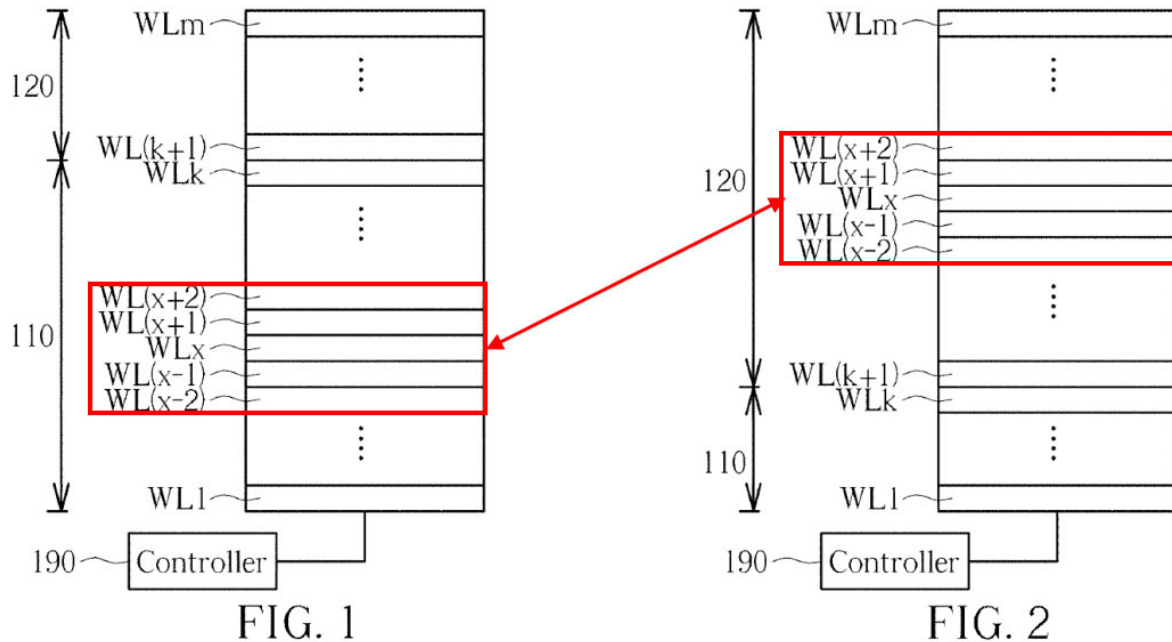


FIG. 1

Id., Figure 1 (excerpted and annotated).

70. The '941 patent applies both of these numbering schemes to a single memory stack to identify in which "portion" a cell is being programmed. Comparing Figures 1 and 2 below demonstrates this where the wordline of the cell being programmed (WLx) is in the first portion 110 in Figure 1, but in Figure 2 the wordline used for programming "WLx is located in the second portion 120." *Id.*, 3:26-28.

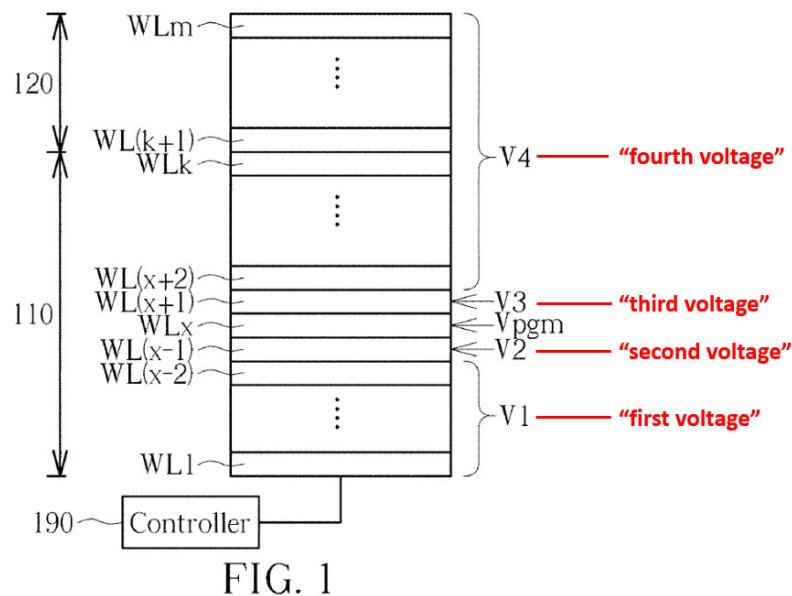


Id., Figures 1 and 2 (excerpted).

71. Turning now to the programming pulses applied during the program operation, the '941 patent says the voltage “ V_{pgm} ” is applied to the wordline for the cell being programmed. *Id.*, 3:9-11 (“When an xth wordline WL_x is used to performed a program operation, the controller 190 may apply a program voltage V_{pgm} to the xth wordline WL_x ”); Figures 1-5 and 7-11. The other wordline voltages are all generically referred to as “first voltage V_1 ,” “second voltage V_2 ,” “third voltage V_3 ,” etc. *See, e.g., id.*, 3:11-16. As noted elsewhere, the '941 patent fails to disclose any actual values (or even ranges) for these generic voltages. Further, the '941 patent is silent regarding the relationships (*e.g.*, whether the values can be the same, overlap, are mutually exclusive, etc.) between V_1 , V_2 , V_3 , etc. with only

very limited exceptions. It is worth reiterating that these are frequently the voltages of the wordlines for the cells that are directly adjacent to the cell being programmed.

72. In some embodiments, a named voltage is applied to a singular wordline and in other embodiments, the named voltage is applied multiple wordlines. *See id.*, Figures 1-5 and 7-11. Figure 1 below is an example where the “first voltage V1” is applied from WL1 to WL(x-2), “second voltage V2” is applied WL(x-1), Vpgm is applied to WLx, V3 is applied to WL(x+1), and V4 is applied from WL(x+2) to WLm. *Id.*, 3:11-21. Figure 2 is the same, except that WLx is in the second portion 120. *See id.*, 3:28-30, Figure 2.



Id., Figure 1 (annotated).

73. I flag one other example here (Figure 8) to show that the voltage label assignments V1, V2, V3, etc. in the '941 patent are rather arbitrary as they differ

significantly from the assignments in Figures 1 and 2. In Figures 7-11, the prefix “7” is appended to the front of the generic voltage names, calling them V71 (first voltage), V72 (second voltage), V73 (third voltage), etc. *Id.*, 4:37-61. Unlike Figure 1, the second voltage in Figure 8 voltage is located **above** the cell being programmed and the third voltage is **below** the cell being programmed. *Id.*, 4:37-61, Figure 8.

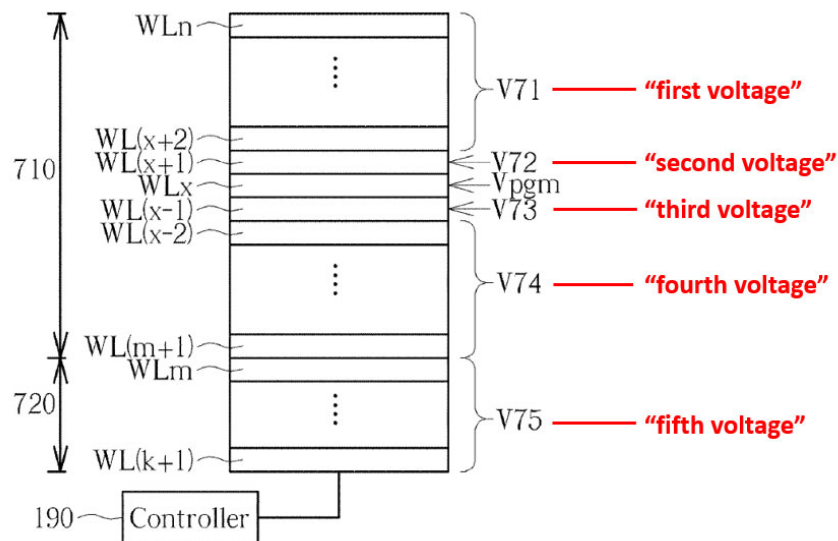


FIG. 8

Id., Figure 8 (annotated). As I previously noted, only in rare instances are relative voltage levels specified, but Figure 8 is one such example that requires the fifth voltage to be less than the fourth voltage. *Id.*, 4:60-61.

74. The last concept I will discuss that is disclosed in the '941 patent has to do with the so-called “first voltage.” The '941 patent states that “the first voltage V1 may have a first level if the xth wordline WL_x is in the first portion 110 and a second level if the xth wordline WL_x is in the second portion 120, where the first

level may be lower than the second level. For example, the first voltage V1 in FIG. 1 may have a lower value than the first voltage V1 in FIG. 2.” *Id.*, 3:35-41. In other words, as programming advances up a memory string (*e.g.*, from source to drain), the pass voltage that is applied to the previously programmed cells may increase.

C. Prosecution History

75. I have reviewed the file history for the '941 patent. The '941 patent is a divisional of U.S. Pat. 11,195,590 (the "'590 patent"). Ex. 1001, Cover. I have also reviewed the file history for the '590 patent. Both file histories are rather short, as the claims in patent were the result of a first-action allowance. *See* Ex. 1002, 378-382 (for the '941 patent) and Ex. 1003, 388-389, 417, and 424-425 (for the '590 patent). The only additional observation I will make here is the Examiner's reasons for allowance in the '941 patent simply parroted most of the claim language for each independent claim without identifying any specific point of novelty. Ex. 1002, 379-380. This comports with my conclusions below—that the challenged claims are not novel.

VIII. CLAIM CONSTRUCTION

76. I am not a legal expert or an attorney and offer no opinions on the law. I understand that claim construction is a matter of law. However, I have been informed by counsel of the legal standards that apply to claim construction in an *inter partes* review, and I have applied them in forming my opinions. I understand

that during an inter partes review, the same claim construction standard used in federal district court is applied when addressing the meaning of claim terms.

77. I have been informed that the words of a claim are generally given the ordinary and customary meaning that the term or phrase would have to POSITA at the time of the invention in view of the surrounding claim language, the specification and the file history (collectively, the “intrinsic evidence” or “intrinsic record”).

78. I also understand that courts may consider extrinsic evidence, such as expert and inventor testimony, dictionaries, and learned treatises (collectively, the “extrinsic evidence” or the “extrinsic record”), but that such extrinsic evidence should be given less weight than the intrinsic evidence.

79. In my opinion, the terms of the '941 patent's claims do not require further construction and can be afforded their plain and ordinary meaning for purposes of the analysis of the prior art that I am conducting in this declaration. Accordingly, I have given the terms in the challenged claims their plain and ordinary meanings, as would have been understood by a person of ordinary skill in the art at the time of the alleged invention, taking into consideration the language of the claims, the specification, and the prosecution history of the '941 patent. I have applied these understandings of the claim terms in my analysis and in forming my opinions where appropriate in this declaration.

IX. DESCRIPTION OF THE MAIN PRIOR ART REFERENCES

80. This section provides brief overviews of the main prior art references that I use in my analysis (which will be detailed in §XI).

A. The Dong Reference (Ex. 1005)

81. The Dong reference refers to U.S. Pub. 2008/0279008, which is titled “Non-volatile Storage with Boosting Using Channel Isolation Switching.” Dong, Cover. Dong was filed on May 7, 2007 and published November 13, 2008. *Id.* This was more than 10 years before the ’941 patent Priority Date.

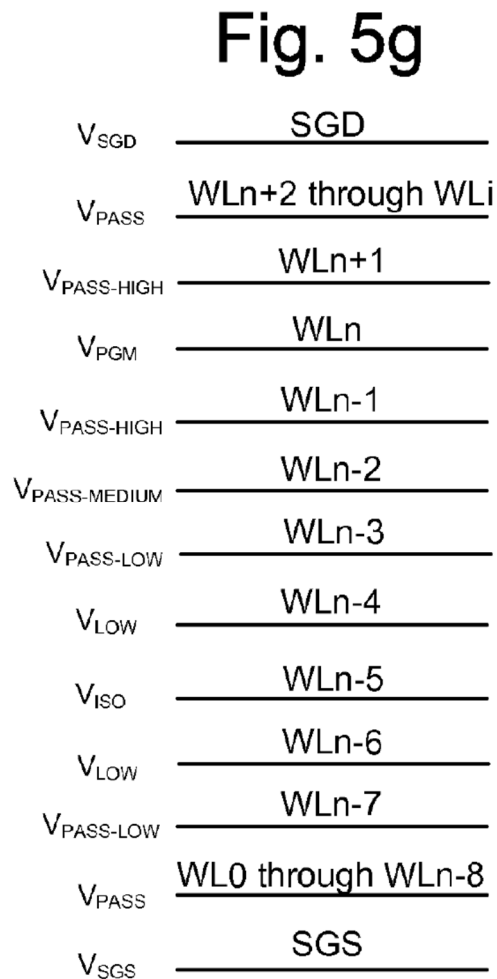
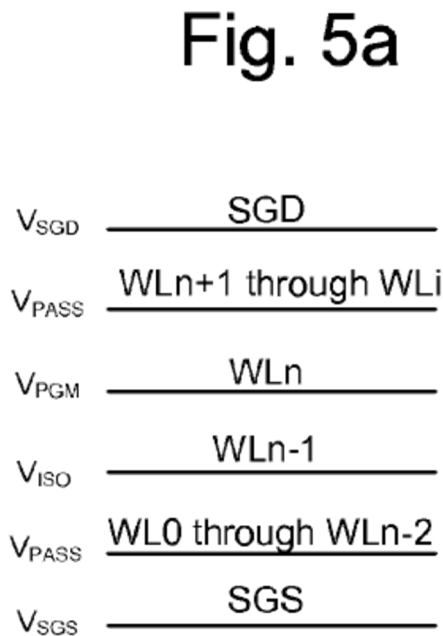
82. Dong is concerned with reducing a phenomenon called “program disturb” in NAND flash memory devices. *Id.*, Abstract. Dong discloses a variety of programming techniques to try and combat program disturb. *See generally* Ex. 1001. Dong discloses that its programming techniques are applicable to both two-dimensional and three-dimensional memory arrays. Dong, [0095].

83. Dong explains that “[t]ypically, a program voltage V_{PGM} [is] applied to the control gate during a program operation ... as a series of pulses that increase in magnitude over time.” Dong, [0009]. I discussed this aspect of the program operation in the Technology Background section. Dong observes that program disturb “continues to be problematic,” and “can occur at inhibited NAND strings during programming of other NAND strings, and sometimes at the programmed NAND string itself.” *Id.*, [0011].

84. “Program disturb occurs when the threshold voltage of an unselected non-volatile storage element is shifted due to programming of other non-volatile storage elements. Program disturb can occur on previously programmed storage elements as well as erased storage elements that have not yet been programmed.”
Id.

85. To reduce program disturb related issues, Dong discloses various programming methods that apply program voltages (V_{PGM}), isolation voltages (V_{ISO}), pass voltages (V_{PASS}), etc. to the wordlines that control the cells in the memory strings. *See generally, e.g., id.*, Figures 4-9. Some of the pass voltages used in Dong include $V_{\text{PASS-LOW}}$, $V_{\text{PASS-MEDIUM}}$, and $V_{\text{PASS-HIGH}}$. Dong, [0056]-[0060]. Dong discloses that “multiple V_{PASS} voltages can be used at the same time. ... Further, multiple V_{PASS} voltages can be used at both the drain and source sides.”
Id., [0057].

86. While Dong discloses several different programming approaches, I noted here only the two that are most relevant to my analysis. These are the embodiments of Figures 5a (described at [0049]-[0051]) and 5g (described at [0058]-[0059]). Both figures are shown below.



Dong, Figures 5a and 5g. At a high level, as described by Dong, “FIGS. 5 a-h depict different examples of ... boosting modes [which] have been developed to combat program disturb. During programming of storage elements on a selected wordline, the boosting modes can be implemented by applying a set of voltages to unselected wordlines which are in communication with storage elements which are not currently being programmed. The storage elements which are being programmed are

associated with selected NAND strings while other storage elements are associated with unselected NAND strings.” *Id.*, [0047].

B. The Costa Reference (Ex. 1012)

87. The Costa reference refers to U.S. Pub. 2018/0012667, which is titled “Wordline Dependent Pass Voltages in Non-volatile Memory.” Costa, Cover. Costa has a U.S. filing date of July 7, 2016 and published on January 11, 2018. *Id.*

88. Costa discloses techniques for programming and reading NAND memory strings. Costa, Abstract. Costa’s abstract provides a good high-level summary of the reference and is repeated below.

Sensing in non-volatile memory is performed using bias conditions that are *dependent on the position of a selected memory cell* within a group of non-volatile memory cells. During sensing, a selected memory cell receives a reference voltage while the remaining memory cells receive a read or verify pass voltage. For at least a subset of the unselected memory cells, *the pass voltage that is applied is dependent upon the position of the selected memory cell in the group*. As programming progresses from a memory cell at a first end of a NAND string toward a memory cell at a second end of the NAND string, for example, *the pass voltage for at least a subset of the unselected memory cells that have already been subjected to programming may be increased*. This technique may reduce the effects of an increased channel resistance that occurs as more memory cells are programmed.

Costa, Abstract (emphasis added). The balance of Costa elaborates on this concept and provides specific examples. The programming techniques disclosed in Costa have applicability to both two-dimensional and three-dimensional NAND memories. *Id.*, [0038], [0048], [0169].

89. Costa's is concerned with an undesirable phenomenon that it refers to as "back-pattern effect," and explains that "later programmed memory cells may be programmed to higher levels which can cause a widening of the final threshold voltage distributions and potential read errors." *Id.*, [0101]. This results from the fact that as a memory string is programmed from the source end toward the drain, "channel resistance increases as each memory cell is programmed." *Id.*, [0029].

90. To address this problem, Costa discloses that "[a]s programming progresses ..., the pass voltage for at least a subset of the unselected memory cells that have already been subjected to programming may be increased." *Id.*, Abstract; *see also id.*, [0030], [0104], Figures 8-10, [0111] ("the second pass voltage $V_{read}+100$ is higher than the first read pass voltage V_{read} by 100 mV"). Costa sometimes describes this approach as "variable pass voltage." *Id.*, [0033].

91. Costa explains how/why increasing the pass voltage is beneficial: "The higher pass voltage applied to the subset of unselected memory cells when verifying a later programmed memory cell increases the channel conductance to compensate for the increase in channel resistance from programming. In this manner,

overprogramming the later programmed memory cells may be avoided.” *Id.*, [0104]. In this manner “the pass voltage that is applied is dependent upon the position of the selected memory cell.” *Id.*, [0028]; *see also id.*, [0030], [0104] (“the pass voltage applied to a subset of unselected memory cells can be varied based on the selected wordline”). Costa provides more specific examples and detailed explanation in connection with Figures 8-10. *See id.*, [0096], [0100]-[0134] (esp. [0110]-[0111] and [0114]-[0116]).

C. The Shirai Reference (Ex. 1018)

92. The Shirai reference refers to U.S. Pub. 2019/0214405, which is titled “Semiconductor Device and Method of Manufacturing the Same.” Shirai, Cover. Shirai has a U.S. filing date of March 13, 2019, and claims priority to a PCT application filed September 21, 2016. Shirai published on July 11, 2019. *Id.*

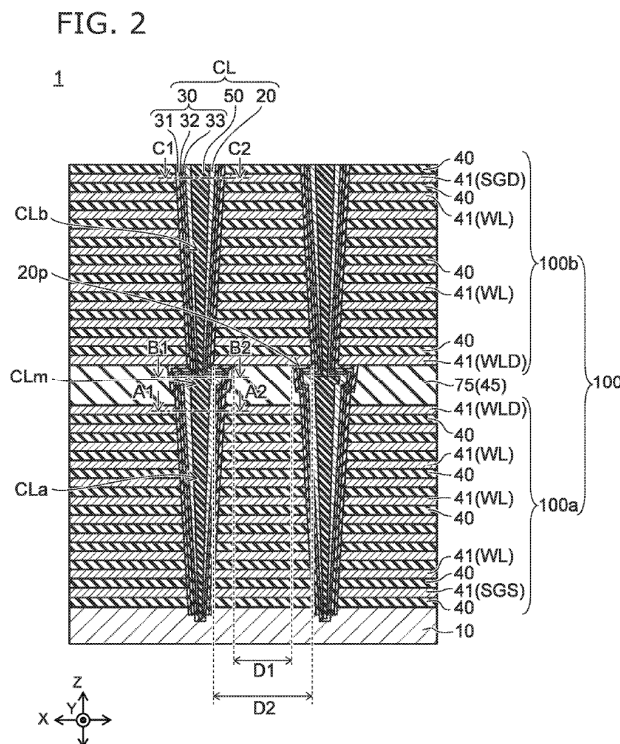
93. Shirai’s abstract provides a good high-level summary of the reference and is repeated below.

According to one embodiment, the stacked body includes a ***first stacked portion*** including a plurality of electrode layers, a ***second stacked portion*** including a plurality of electrode layers, and being disposed separately from the first stacked portion in the first direction, and a ***connection portion*** including a high dielectric layer provided between the first stacked portion and the second stacked portion and having a dielectric constant higher than a dielectric constant of the insulator. The column-shaped portion includes a first portion provided in the first

stacked portion and extending in the first direction of the stacked body, a second portion provided in the second stacked portion and extending in the first direction, and an intermediate portion provided in the connection portion and connected the first portion to the second portion.

Shirai, Abstract (emphasis added).

94. Shirai discloses a 3D NAND memory device that includes multiple memory decks, which Shirai refers to as “stacked portions.” *See, e.g.*, Shirai, [0003]; Figures 2 and 7-15. Shirai’s figures are quite illustrative and I reproduce Figure 2 below to show “a first stacked portion 100a, a connection portion 45, and a second stacked portion 100b.” *Id.*, [0030].



Id., Figure 2. The so-called connection portion 45 is located between the upper deck 100b and lower deck 100a and serves to connect them together. Shirai discloses that

the connection portion 45 may be “made of the same material (for example, a silicon oxide) as the insulator 40.” *Id.*, [0045]; *see also id.*, [0031] (“The connection portion 45 includes a high dielectric layer 75,” which may comprise “a hafnium oxide, a zirconium oxide, and an aluminum oxide.”).

95. Shirai explains that while there are physically two memory decks, the memory strings in each deck align and are electrically connected together to form a single memory string that spans both memory decks and may operate as a single memory string. Shirai, [0033]-[0042]. One artifact of this approach is that memory decks 100a and 100b “may include the electrode layer 41 functioning as a dummy wordline WLD.” *Id.*, [0032]. In lower deck 100a, “electrode layer 41 closest to the connection portion 45 functions as a dummy wordline WLD,” and in upper deck 100b, “electrode layer 41 closest to the connection portion 45 functions as a dummy wordline WLD.” *Id.* The dummy wordlines are added to improve performance of the memory string.

X. UNPATENTABILITY ANALYSIS

96. As previously noted, I have been asked to evaluate the patentability of claims 1, 3-11, and 13-14 of the '941 patent in view of the prior art references listed above in §IX. For the reasons I explain in greater detail below, it is my opinion that as of the Priority Date of the '941 patent, a POSITA would have considered all of the challenged claims to be unpatentable on the following grounds:

Ground	Rationale	Reference(s)	Claims
1	Anticipation	Dong	1, 3, 5-7, 11, and 13
2	Obviousness	Dong	1, 3, 5-7, 11, and 13
3	Obviousness	Dong combined with Costa	8 and 9
4	Obviousness	Dong combined with Shirai	4, 10, and 14

97. I note here that throughout my declaration I indicate claim language using *red italics*.

A. Grounds 1 and 2: Claims 1, 3, 5-7, 11, and 13 are Anticipated (Ground 1) and Rendered Obvious (Ground 2) by Dong

98. Dong anticipates (Ground 1) claims 1, 3, 5-7, 11, and 13. And Dong, by itself, also renders obvious (Ground 2) the same claims (*i.e.*, single reference obviousness). Of course, the analysis is performed in view of the knowledge of a POSITA and I point out those occasions where I rely on a POSITA's knowledge.

99. Before conducting my element-by-element analysis, I make some initial comments about how the analysis is structured. Grounds 1 and 2 both rely on Dong as the sole reference and share much of the same analysis. Accordingly, to avoid repetition, I analyze both grounds together in this section.

100. I also observe that the '941 patent has three independent claims: 1, 5, and 11. Claims 1 and 11 are substantially similar, with claim 1 being an apparatus

claim and claim 11 being the method version of claim 1. Thus, claims 1 and 11 (and their dependent claims) can be treated as one group of claims, and claim 5 (and its dependent claims) can be treated as another group of claims.

101. The two groups of claims invoke different embodiments from Dong for the analysis. Dong Figure 5a reads on claim 5 (and its dependent claims), whereas Dong Figure 5g reads on claims 1 and 11 (and their dependent claims).

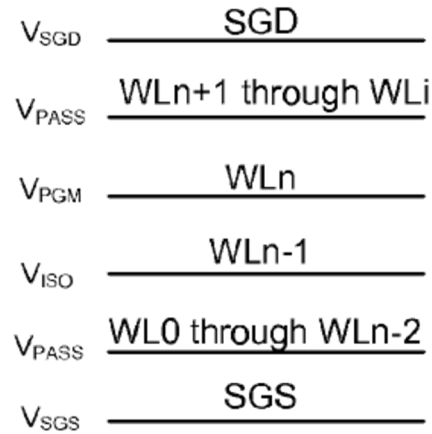
102. The last point I note is that independent claim 5 is broader in various respects than independent claims 1 and 11. For example, claim 5 only recites three voltages, whereas claims 1 and 11 recite five voltages. Accordingly, I begin my analysis with claim 5.

1. Claim 5

103. With respect to claim 5, it is helpful to note that it corresponds to Figure 1 in the '941 patent. It also reads on other Figures in the '941 patent but it is not necessary to discuss them.

104. Dong by itself both anticipates and renders claim 5 obvious. As I mentioned above, the embodiment in Dong that maps to claim 5 is Figure 5a. Before examining each claim element, I explain here several points about Dong Figure 5a that apply throughout the analysis so I don't need to repeat them over and over. Dong Figure 5a is shown below.

Fig. 5a



Dong, Figure 5a. 1. At the top and bottom of the Figure are “SGD” (top) and “SGS” (bottom). These acronyms would have been familiar to a POSITA. Consistent with industry usage, Dong teaches that SGS is “the source-side select gate control line” and SGD is “the drain-side select gate control line.” Dong, [0048].

105. As was also known to a POSITA, and as shown in Figure 5a, the wordlines that are connected to memory cells storing data begin with the letters “WL” (an acronym for WordLine). The wordlines for cells storing data do not include SGS or SGD. Said another way, even though a memory string may have cells controlled by the SGS and SGD lines, those control lines are not termed “wordlines” because they are not used for storing data.

106. In Dong Figure 5a, many wordlines are identified by their relative location to wordline “n,” which is Dong’s notation to identify the wordline being

used for programming. Dong, [0049] (“a program voltage, V_{PGM} [] is applied to the selected wordline WL_n ”).

107. I further note that the first wordline in Dong Figure 5a is termed “WL0.” Using such a convention (*i.e.*, numbering a series starting with “0” instead of “1”) was commonplace among engineers and is frequently used even today. While Dong assigned the first/bottom wordline a specific number (“WL0”), the uppermost wordline is denoted “WL i ” (*i.e.*, it is not assigned a specific number). This, too, was (and is) commonplace among engineers and a POSITA would understand that Dong’s index “ i ” is used to indicate that the total number of wordlines (and corresponding memory cells in the memory string) is open ended. In other words, Dong is disclosing that a wide range of wordlines may be used in the memory cell strings, and the exact number to use in a given design is left to the designer as a design choice. Dong itself gives some examples of different numbers of wordlines. For instance, Dong Figure 4 shows an example with 8 wordlines. And Dong [0090] discloses “a 32 wordline NAND string.” Dong [0108] says that “In the example provided, four storage elements are connected in series to form a NAND string. Although four storage elements are shown to be included in each NAND string, more or less than four can be used (e.g., 16, 32, 64 or another number).” In Dong’s disclosure, the number of wordlines in a string would be $(i+1)$, since the wordline numbering starts with WL0.

108. In this regard, I note that Dong incorporated by reference U.S. Pat. 6,859,397 (Ex. 1015, “Lutze”). Dong, [0010]. I am informed and understand that incorporation by reference means that the incorporated material is considered part of the document which incorporates it. Thus, in this case, the Lutze disclosure is considered part of the Dong reference.

109. The Lutze reference discloses that “some NAND strings will include eight memory cells, 16 memory cells, 32 memory cells, etc.,” and goes on to explicitly say that its discussion “is not limited to any particular number of memory cells in a NAND string.” Lutze, 2:11-18. This is the same as what I just explained. Thus, a POSITA would have understood that Dong discloses an open-ended length for its memory string. Given the disclosure in both Dong and Lutze, a POSITA would have understood that the memory string in Dong Figure 5a to at least include a 32-cell memory string. Moreover, a POSITA would have understood, or at least found it obvious, to use a longer memory string, such as 64 or 128 cells, in Dong Figure 5a. Dong, [0108] (referring to, *e.g.*, 64 bit memory string).

110. I note that memory strings with lengths of 32, 64, and 128 cells were well-known to POSITAs and commonplace by the Priority Date and all of them would have been obvious to use, as reflected by a number of references. For example, U.S. Pub. 2008/0159003 (Ex. 1006, “Dong II”) includes examples of 32 wordlines. Dong II, [0010], Figure 1 (showing 32 wordlines). As another example,

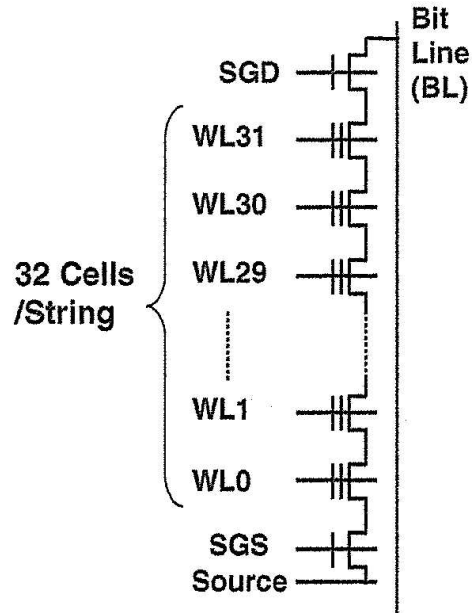
the Costa reference (discussed further for Ground 3), discloses that “[t]he NAND string in this example includes 128 data wordlines.” Costa, [0093]. As a third data point, the Shirai reference (discussed further for Ground 4) states that “[a]ny number of electrode layers 41 may be stacked.” Shirai, [0019]. As further example, U.S. Pub. 2019/0102104 (Ex. 1013, “Righetti”) Figure 6 shows a memory device with 74 wordlines. And U.S. Pub. 2012/0163096 (Ex. 1017, “Futatsuyama”) explains that memory strings may have “8, 16, 32, 33, 34, 64, 66, 68 or 88” cells. Futatsuyama, [0050]. More generally, as discussed in the Technology Background, there has been a constant demand for increased memory density that has led to increasing number of memory cells in a memory string. Certainly by the 2020 Priority Date, using memory strings with 64 cells or more was common.

111. Even though claim 5 does not recite any specific number of wordlines, I found it useful when performing my analysis to utilize a specific example and picked a 32-cell NAND string since that was expressly disclosed in Dong, but I note that other memory string lengths would work just as well and would have been obvious to a POSITA to achieve higher memory densities.

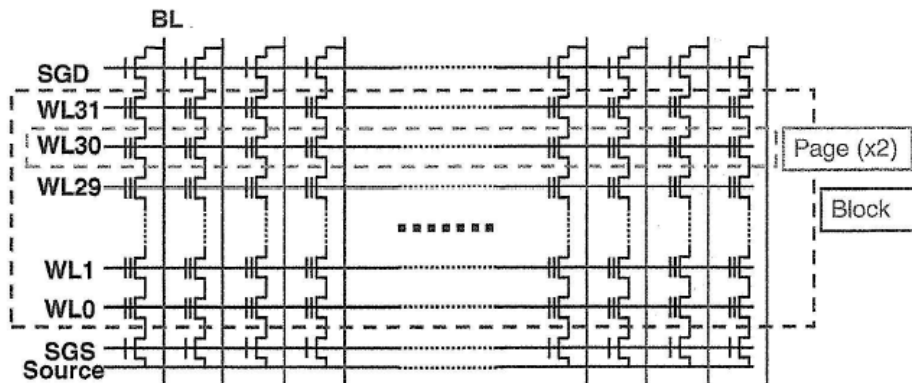
112. With respect to memory string programming, it was well-known to POSITA’s that programming typically proceeded one cell at a time, either from source to drain or from drain to source depending on the architecture of the system. Indeed, Dong Figure 10 shows that programming continues sequentially until all

memory cells in a string are programmed. Dong, Figure 10, [0089]. In particular, Dong explains that “[i]f programming for the current wordline is complete but programming for all wordlines is not complete, at decision step 1065, the next wordline is selected for programming at step 1075. If programming for the current wordline and all wordlines is complete, programming ends at step 1070.” *Id.*, [0089].

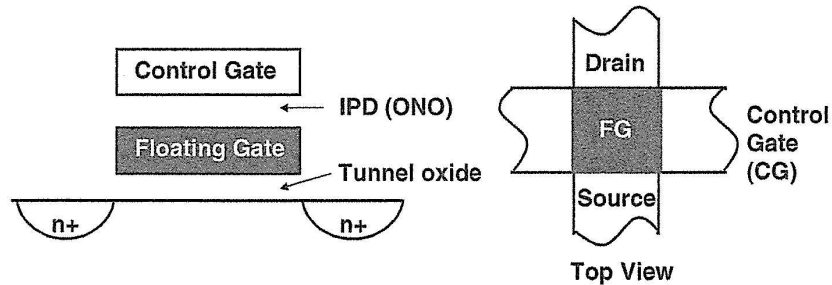
113. I make one general observation here that applies more generally than just the Dong reference. Consistent with the figures in this declaration, a POSITA would be familiar the convention that a memory array, consisting of many memory strings, would have the source end of the NAND string at the bottom, and the drain end of the NAND string at the top. This is illustrated, for example, in the figures below from the Aritome textbook (Ex. 1009), which are discussing two-dimensional NAND flash memories. *See* Ex. 1009, 39-40.



Ex. 1009, Figure 2.2.



Id., Figure 2.4. This is consistent even with the schematic of a regular NMOS transistor that always has drain end at the top and source end at the bottom, as shown below.



Id., Figure 2.1. This convention applies irrespective of whether the physical device is in a horizontal or vertical direction.

114. I return now to the 32-cell memory string example and its application to Dong Figure 5a. As noted, every cell in a memory string will often be programmed and, to provide a concrete analysis, I select the fourth cell in Dong Figure 5a as the cell being programmed. This corresponds to WL3 in Dong Figure 5a because Dong begins at WL0. In this example, the wordlines have the wordline assignments and voltage labels shown below in annotated Figure 5a.

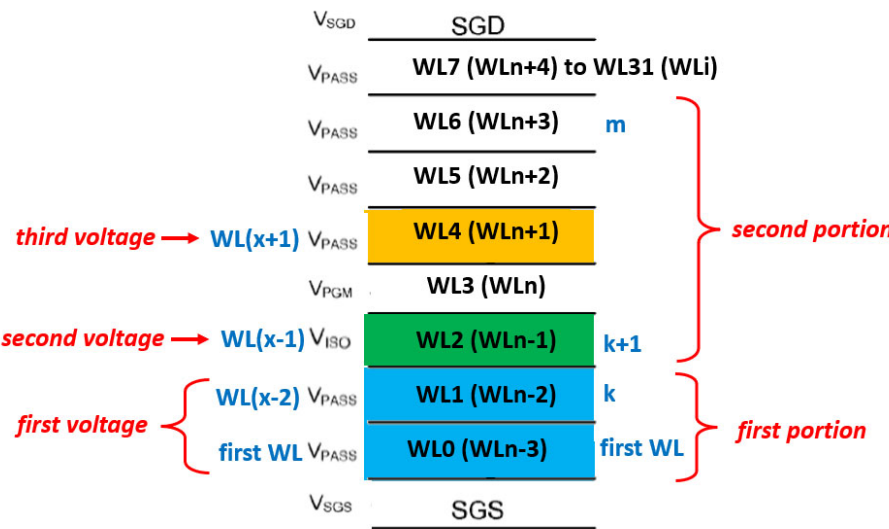
Fig. 5a

V_{SGD}	<u>SGD</u>
V_{PASS}	<u>WL7 (WL_n+4) to WL31 (WL_i)</u>
V_{PASS}	<u>WL6 (WL_n+3)</u>
V_{PASS}	<u>WL5 (WL_n+2)</u>
V_{PASS}	<u>WL4 (WL_n+1)</u>
V_{PGM}	<u>WL3 (WL_n)</u>
V_{ISO}	<u>WL2 (WL_n-1)</u>
V_{PASS}	<u>WL1 (WL_n-2)</u>
V_{PASS}	<u>WL0 (WL_n-3)</u>
V_{SGS}	<u>SGS</u>

Dong, Figure 5a (annotated).

115. The image below simply adds colors and additional labels to the figure above to show how Dong Figure 5a maps to claim 5. I discuss each claim element in detail below.

Fig. 5a



Dong, Figure 5a (annotated). I also found it helpful in conducting my analysis to locate all of the variables, etc. being sued in one place for reference. I wish to emphasize that the mappings below are just examples and that other mappings could also satisfy claim 5.

$k = 2$	$WL(m) = WL6$	$WL(x+1) = WL(n+1) = WL4$
$m = 7$	$WL(k+1) = WL2$	$WL(x) = WL(n) = WL3$
$x = 4$	$WL(k) = WL1$	$WL(x-1) = WL(n-1) = WL2$
	$first\ wordline = WL0$	$WL(x-2) = WL(n-2) = WL1$

- a. *[5.PRE] “A program method of a memory, the memory comprising a first portion and a second portion, the first portion comprising a first wordline to a kth wordline from bottom to top, the second portion formed above the first portion and comprising a (k+1)th wordline to an mth wordline from bottom to top, the program method comprising:”*

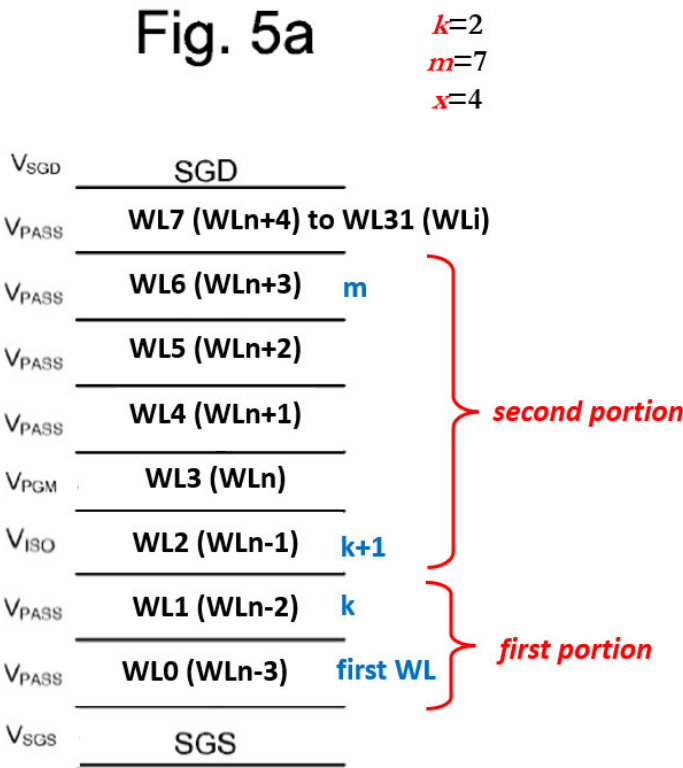
116. I offer no opinion on whether the preamble is limiting. I was instructed by counsel to analyze the preamble as if the it was a limitation. In my opinion, Dong discloses the preamble.

117. To begin, Dong is replete with disclosure of a *memory*. See Dong, Figure 12 (“memory device 1296,” “memory die 1298,” and “memory array 100”); [0095] (“Memory device 1296 may include one or more memory die 1298. Memory die 1298 includes a two-dimensional array of storage elements 1100, control circuitry 1210, and read/write circuits 1265. In some embodiments, the array of storage elements can be three dimensional.”).

118. And Dong’s primary focus is with improving *program method[s]* for programming a memory. See, e.g., Dong, Figures 5 and 10, [0009]-[0010], [0048] (“A set of voltages which is applied to the control lines is also depicted. Programming can proceed in a programming sequence one wordline at a time”).

119. I note that the term *portion* is not a term of art and I afford that term it plain and ordinary meaning. The limitations of the claimed *first* and *second portion[s]* are set forth by the claim itself: *the first portion compris[es] a first*

wordline to a kth wordline from bottom to top; and *the second portion formed above the first portion and comprising a (k+1)th wordline to an mth wordline from bottom to top*. Dong’s *first* and *second portion[s]* are identified in Figure 5a below.



Dong, Figure 5a (annotated).

120. Some discussion of the nomenclature used in claim 5 is in order. In particular, two aspects of the claim language merit explanation. I begin with the phrase *bottom to top*. The '941 patent does not provide a definition for *bottom to top*. But, as I explained previously, it was (and is) common when discussing NAND flash memory to refer to the source end of a memory string as the “bottom,” and the drain end as the “top.” I apply that common usage here. In any event, even if *bottom*

to top were interpreted as requiring vertical memory strings in a 3D NAND device, Dong discloses that its “array of storage elements can be three dimensional.” Dong, [0095]. And Figure 5a itself shows a vertical orientation (consistent with Dong’s disclosure of a “three dimensional” device) which satisfies the *bottom to top* language. Thus, regardless of whether Figure 5a is read as a 2D or 3D memory string it meets the claim language. And, given the similarities between 2D and 3D NAND discussed in the Technology Background section, even a 2D memory string would at a minimum render obvious the *bottom to top* requirement, even if narrowly construed.

121. Next, I turn to the claim’s usage of variables, in particular the variables *k* and *m*. The claim uses the variables as indices for referencing a relative wordline location corresponding to a memory string cell. Neither *k* nor *m* is a term of art that has a specific meaning in the context of NAND flash memory. They are just variables for denotation purposes, and there is nothing special about the claim’s selection of those particular variables. Indeed, other variables (e.g., “x”, “y,” “z,” etc.) could have been used instead without substantively altering the meaning of the claim.

122. As would be readily apparent to a POSITA based on known conventions, a POSITA would understand that the claimed *(k+1)th wordline* refers to the wordline immediately above the wordline *k*. In like manner, the “k-1”

wordline would refer to the wordline immediately below the wordline k . I note that this is similar to how the '941 patent specification illustrates its indexing scheme. *See, e.g.*, Ex. 1001, Figure 1. The same naming convention applies to identifying wordlines that are further away from a reference wordline (*e.g.*, the “ $k+2$ ” wordline would be two wordlines above the wordline k). The same principle applies regardless of what variable is chosen to denote the reference wordline. The discussion in this and the preceding paragraph regarding variables applies generally wherever other variables (such as n and x) are used in the challenged claims.

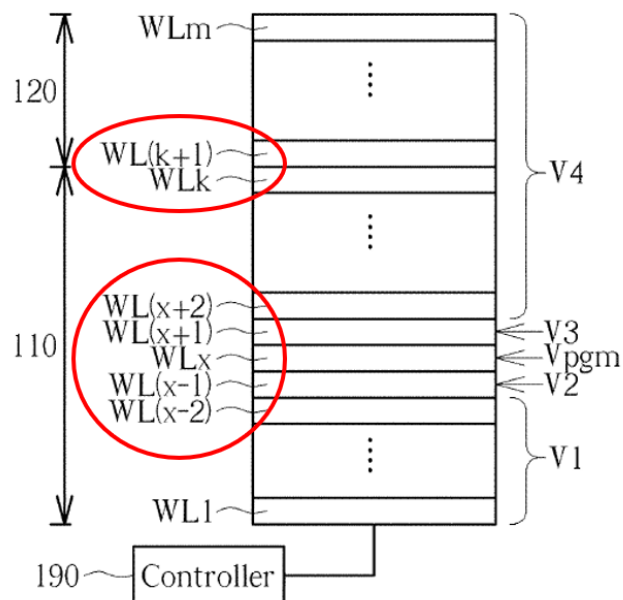


FIG. 1

123. Continuing with the analysis of how the variables k and m are used in claim 5, the only specific requirements the claim places on these variables are that k

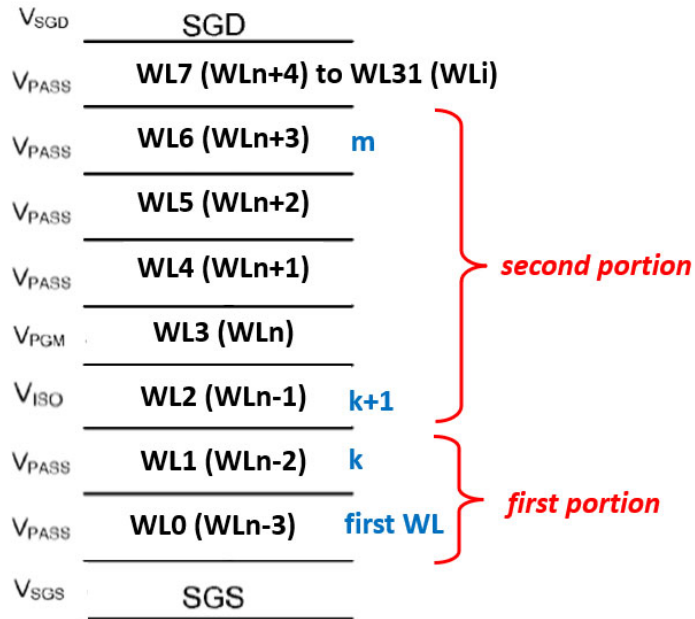
and m are positive integers, $1 < k < m$. These conditions are imposed by claim element [5.D] and I discuss them further for when analyzing that claim element.

124. The claimed *first portion* in Dong corresponds to wordlines WL0 through WLn-2 (*viz.*, WL1) in Figure 5a as shown below. The claimed *second portion* in Dong's corresponds to wordlines WLn-1 (*viz.*, WL2) through WLn+3 (WL6) in Figure 5a.

125. Instead of using the variable *x* to identify the wordline being used for programming in the '941 patent, Dong instead uses the variable "n" to identify the wordline being sued for programming (and by extension to indicate other wordline locations relative thereto). Dong, [0049]. Dong says "a program voltage, V_{PGM} [] is applied to the selected wordline WLn." *Id.*; *see also id.*, Figure 5a (showing wordline "WLn-1" as one wordline below "WLn").

Fig. 5a

$k=2$
 $m=7$
 $x=4$



Dong, Figure 5a (annotated).

126. Claim 5 further specifies that the *first portion* includes *a first wordline to a kth wordline from bottom to top*. Dong meets this requirement. As previously noted, Dong says that SGS refers to “the source-side select gate control line” and SGD is “the drain-side select gate control line.” Dong, [0048]. The convention used in Dong Figure 5a is that “WL” refers to wordlines connected to cells that actually store data, and do not include SGS or SGD. Accordingly, Dong’s wordline WL0 corresponds to the *first wordline from bottom to top*. And WLn-2 (viz., WL1) in Dong is the claimed *kth wordline*.

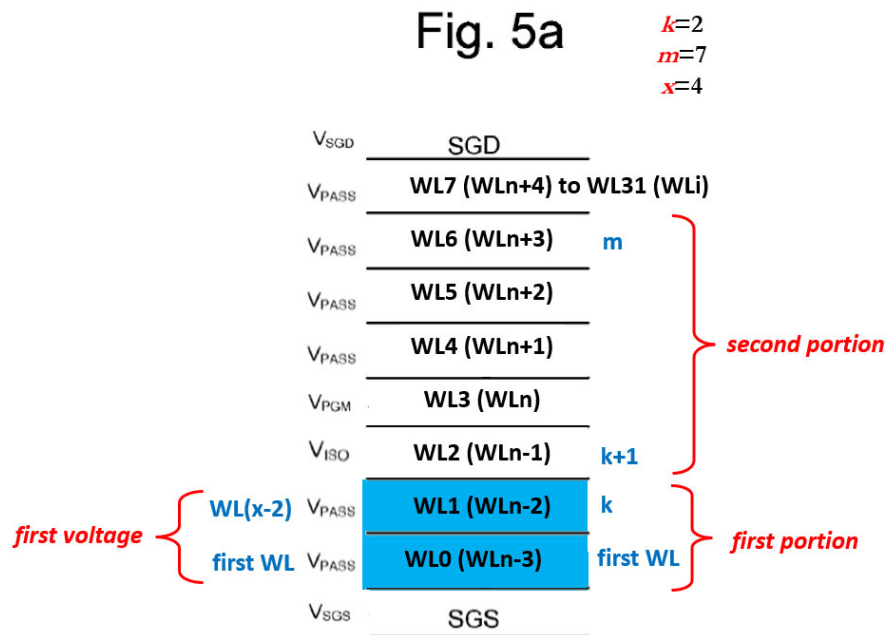
127. Turning now to the *second portion*, the claim specifies it is *formed above the first portion* and includes *a (k+1)th wordline to an mth wordline from bottom to top*. Dong Figure 5a meets this requirement. The wordline WLn-1 (viz., WL2) in Dong Figure 5a corresponds to the *(k+1)th wordline*. And wordline WLn+3 (viz., WL6) in Dong maps to the *mth wordline* in the claim. Using the specific example I introduced in the claim 5 prefatory discussion, the means that *m=7*. As can be seen in Figure 5a, Dong's *second portion* (from WL2 to WL6) is *formed above the first portion* (from WL0 to WLn-2). Accordingly, in summary, Dong satisfies the claim 5 preamble.

b. [5.A] “applying a first voltage to the [sic] to the first wordline to an (x-2)th wordline when an xth wordline is used to perform a program operation;”

128. In my opinion, Dong discloses claim element [5.A]. For starters, the variable *x* in claim 5 identifies the wordline that *perform[s] a program operation*. Dong uses a different variable (“n”) to indicate the wordline being used for programming a memory cell. Dong, [0049] (“a program voltage, V_{PGM} , which is applied to the selected wordline WLn”); *see also id.*, [0048] (“the selected wordline is WLn”), Figure 5a. Thus, the claimed *xth* wordline corresponds to the selected wordline WLn in Dong Figure 5.

129. The claim specifies that when the *xth* wordline is used for programming, a *first voltage* is applied *to the first wordline to an (x-2)th wordline*.

In other words, the *first voltage* is applied from the first/bottom wordline through the wordline that is two cells below the cell being programmed. Dong discloses this. Recall that in Dong, WLn is used for programming. Accordingly, in Dong Figure 5a, a *first voltage* must be applied to the wordlines from $WL0$ (the *first wordline*) to $WLn-2$ (viz., $WL1$), which is the *(x-2)th wordline*. As shown in Figure 5a below, Dong meets this requirement by applying V_{PASS} (the *first voltage*) from $WL0$ (the *first wordline*) through $WL1$ (viz., $WLn-2$, the *(x-2)th wordline*). Dong, Figure 5a.



Dong, Figure 5a (annotated).

130. Dong plainly discloses that “a pass voltage, V_{PASS} , [] is applied to each of the unselected wordlines, $WL0$ through $WLn-2$.” Dong, [0049]. I note that neither the ’941 patent specification nor claim 5 specifies any particular voltage (or even a range of voltages) that correspond to the *first voltage*. See generally Ex.

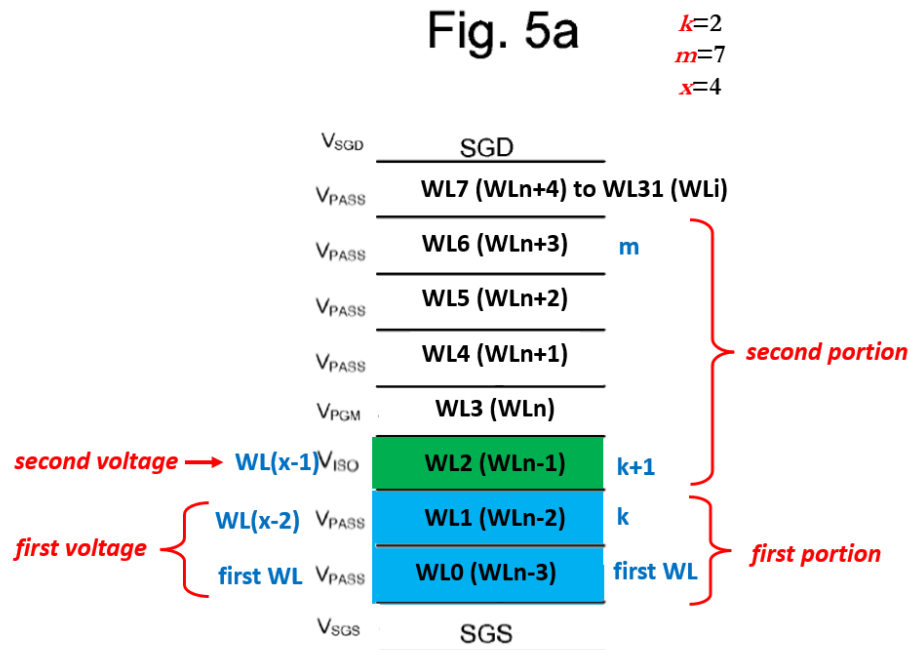
1001. Indeed, the '941 patent specification never even gives an example of an actual voltage. Given that different manufacturing processes and architectures may be used for NAND flash memory, a POSITA would not understand the claim as requiring any particular voltage. Accordingly, all that is required by claim element [5.A] is that the same voltage (whatever voltage it may be) is applied to the recited wordlines. Thus, because, V_{PASS} in Dong is applied to the recited wordlines, Dong satisfies element [5.A].

c. [5.B] “applying a second voltage to an (x-1)th wordline; and”

131. In my opinion, Dong discloses element [5.B]. The same discussion in the previous paragraph regarding the lack of an actual voltage for the *first voltage* applies to the *second voltage*. The claimed *second voltage* in Dong Figure 5a is V_{ISO} , which refers to an isolation voltage. Dong, [0046]. As shown in Figure 5a below, Dong's V_{ISO} is applied to $\text{WL}(n-1)$, which corresponds to the *(x-1)th wordline*. Dong confirms this when it says that “an isolation voltage V_{ISO} [] is applied to $\text{WLn}-1$, the wordline which is adjacent to the selected wordline on the source side ... A low isolation voltage V_{ISO} , in a typical range of 0-4 V, is applied to the wordline which is adjacent to the selected wordline on the source side, in the example of FIG. 5 a.” Dong, [0049]. As has previously been explained, the *(x-1)th wordline* is the wordline immediately beneath the wordline used for programming

(i.e., the wordline below x). In Dong, the $(x-1)$ th wordline is $WLn-1$ (viz., $WL2$).

In summary, Dong satisfies *applying a second voltage* [V_{ISO}] *to an $(x-1)$ th wordline* [$WLn-1$].

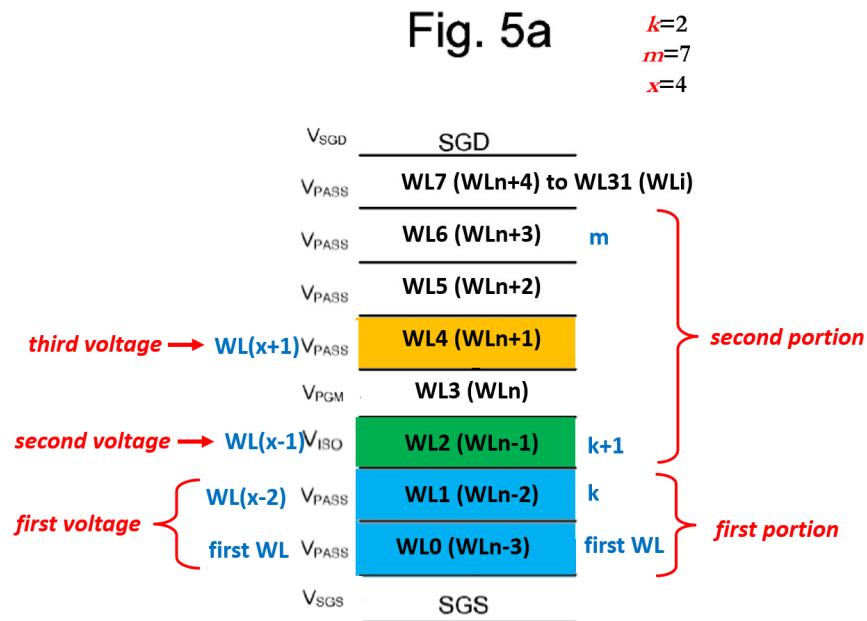


Dong, Figure 5a (annotated).

d. [5.C] “applying a third voltage to an $(x+1)$ th wordline;”

132. In my opinion, Dong discloses claim element [5.C]. The same discussion regarding the lack of an actual voltage for the *first voltage* applies to the *third voltage*. The claimed *third voltage* in Dong Figure 5a is V_{PASS} as shown below. Dong teaches that “a pass voltage, V_{PASS} , [] is applied to each of the unselected wordlines,” including “ $WLn+1$.” Dong, [0049]. As I discussed previously, the $(x+1)$ th wordline is the wordline immediately above the wordline used for

programming (*i.e.*, the wordline above x). Accordingly, in Dong Figure 5a, the $(x+1)$ th wordline is W_{Ln+1} (*viz.*, WL_4). Thus, Dong discloses *applying a third voltage* [V_{PASS}] *to an $(x+1)$ th wordline* [W_{Ln+1}].



Dong, Figure 5a (annotated).

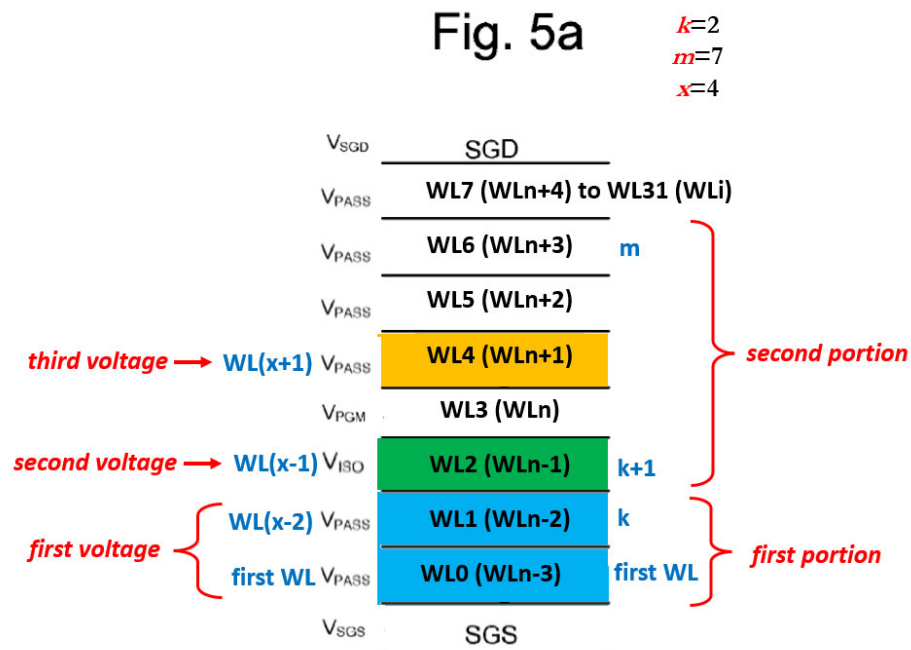
133. Nothing in claim 5 or the '941 patent specification specifies that the *third voltage* has to be different from the *first voltage*. See generally Ex. 1001. Accordingly, even if V_{PASS} applied on WL_0 - WL_1 were the same value as V_{PASS} applied on WL_4 , Dong Figure 5a would still satisfy claim element [5.C]. As a point of contrast, I note that elsewhere the '941 patent claims do require different relative voltages, such as in claim 1 (*the fifth voltage is lower than the fourth voltage*). Thus, the patentee knew how to require different voltage values, but claim 5 recites no such thing.

e. [5.D] “wherein x , k and m are positive integers, $1 < k < m$ and $3 \leq x$.”

134. In my opinion, Dong discloses element [5.D]. Integers are a well-known concept in mathematics and the term has the same meaning here: a whole number such as 1, 2, 3, etc. (e.g., as opposed to a fraction). See <https://dictionary.cambridge.org/us/dictionary/english/integer>. A POSITA would understand that the index “n” as used Dong to identify wordlines is always an integer. See generally Dong. Just like the ’941 patent never provides any actual numbers/values for the wordline x (or the variables k and m for that matter), Dong also refers to the wordline being used for programming as “WL n ” (and references other wordline locations relative thereto). Thus, Dong satisfies element [5.D] in the same way as described in the ’941 patent.

135. Even though the ’941 patent fails to give any concrete examples, I do so here with respect to Dong as it is helpful for purposes of understanding. As noted in my prefatory comments for claim 5, WL3 (viz., WL n) is the wordline being used for programming in Figure 5a. When WL3 is being programmed, this means that $x=4$ in Dong. Even though $x=4$, WL3 is being programmed because Dong’s first wordline is WL0 instead of WL1. I note that engineers frequently begin numbering a series of items with “0” instead of “1.” Thus, this convention would have been well-known to a POSITA, and is especially common in systems involving digital logic where binary is used, including NAND memory devices.

136. Given that Dong's index "n" correlates to the claimed variable x , the claimed condition $3 \leq x$ is satisfied that (i.e., $3 \leq 4$ because $x=4$). In the example being used $k=2$, which satisfies the inequality condition that $1 < k$ (i.e., $1 < 2$). Continuing with the example where $k=2$, the *first portion* has a *first wordline* [WL0] *to a kth wordline from bottom to top* [WL1]. Even though $k=2$, the *kth wordline* is WL1 in Dong Figure 5a because Dong's first wordline is WL0. Recall that in the example being used $m=7$. Thus, the *second portion* has a *(k+1)th wordline* [WL2] *to an mth wordline from bottom to top* [WL6]. Again, because Dong starts with WL0, WL6 is the seventh wordline from the bottom. Because $k=2$ and $m=7$, the condition is satisfied that $k < m$ (i.e., 2 is less than 7). Thus, Dong satisfies all the claimed conditions and meets element [5.D].



Dong, Figure 5a (annotated).

2. Claim 6: “The program method of claim 5, further comprising:”

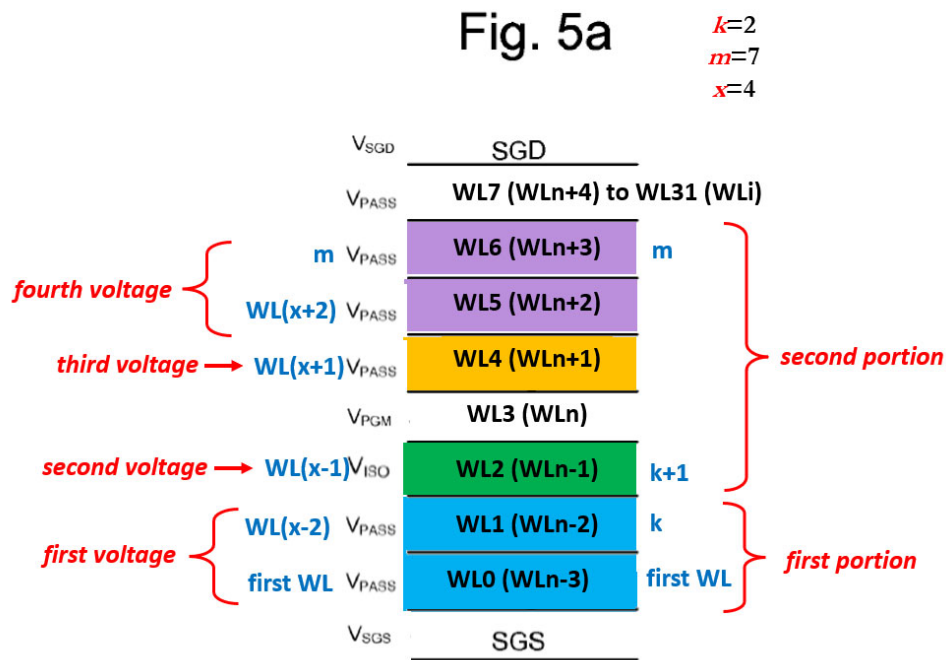
137. Claim 6 is a dependent claim and depends from claim 5. Because Dong satisfies claim 5 as discussed above, the preamble of claim 6 is met as well. Claim 6 has two additional elements. As discussed below, Dong satisfies both of them.

a. [6.A] “applying a fourth voltage to an (x+2)th wordline to the mth wordline; and”

138. Dong discloses claim element [6.A]. The *fourth voltage* in Dong is the V_{PASS} voltage applied to WL5 and WL6 in the example of Figure 5a being used. As I have already discussed for the *first*, *second*, and *third voltages*, there is likewise no requirement that the *fourth voltage* be a different voltage than any of the previously recited voltages. Neither the '941 patent specification nor claims say that, or even given an example where the voltages are different. Rather, the only requirement is simply that the same *fourth voltage* (whatever voltage it is) be applied to the specified wordlines (*i.e., to an (x+2)th wordline to the mth wordline*).

139. As a reminder, in Dong's nomenclature, “n” is the wordline that is used for programming. Dong, [0049] (“a program voltage, V_{PGM} [] is applied to the selected wordline WLn ”). This corresponds to the variable *x* as recited in claim element [5.A] (the *xth wordline is used to perform a program operation*). Thus, the *(x+2)th* wordline in element [6.A] corresponds to $WLn+2$ (*viz.*, WL5) in Dong Figure 5a because WLn (*viz.*, WL3) is the wordline being used for programming.

140. Recall that $m=7$ in the specific example being used for analysis. Dong states that, in Figure 5a, “a pass voltage, V_{PASS} , [] is applied to each of the unselected wordlines, WL_0 through WL_{n-2} and WL_{n+1} through WL_i .” Dong, [0049]. As a result, wordlines WL_{n+2} (viz., WL_5) through WL_{n+3} (viz., WL_6) will necessarily receive the V_{PASS} voltage because they are unselected wordlines within Dong’s recited range of WL_{n+1} through WL_i . Thus, *a fourth voltage* (V_{PASS}) is applied *to an $(x+2)$ th wordline* (WL_5) *to the m th wordline* (WL_6) as shown below in purple. Accordingly, Dong satisfies element [6.A].



Dong, Figure 5a (annotated).

b. [6.B] “wherein $x < (m-1)$.”

141. In my opinion, Dong discloses element [6.B]. Because Dong begins numbering its wordlines from WL0, the wordline used for programming (WLn) is WL3, as Figure 5a above demonstrates. And in the specific example being used, the index $x=4$ (i.e., the fourth wordline from the bottom is being used for programming).

142. Independent claim 5 specifies that the *second portion* extends to the *mth wordline from bottom to top*. As stated in various places throughout the analysis of claim 5, $m=7$ in the example being used. Accordingly, $m-1=6$. Plugging in the values for x and m into the inequality (i.e., $x < (m-1)$), it is apparent that the inequality is satisfied by Dong. The substitution results in $4 < 6$, which is a true statement. Accordingly, Dong satisfies element [6.B].

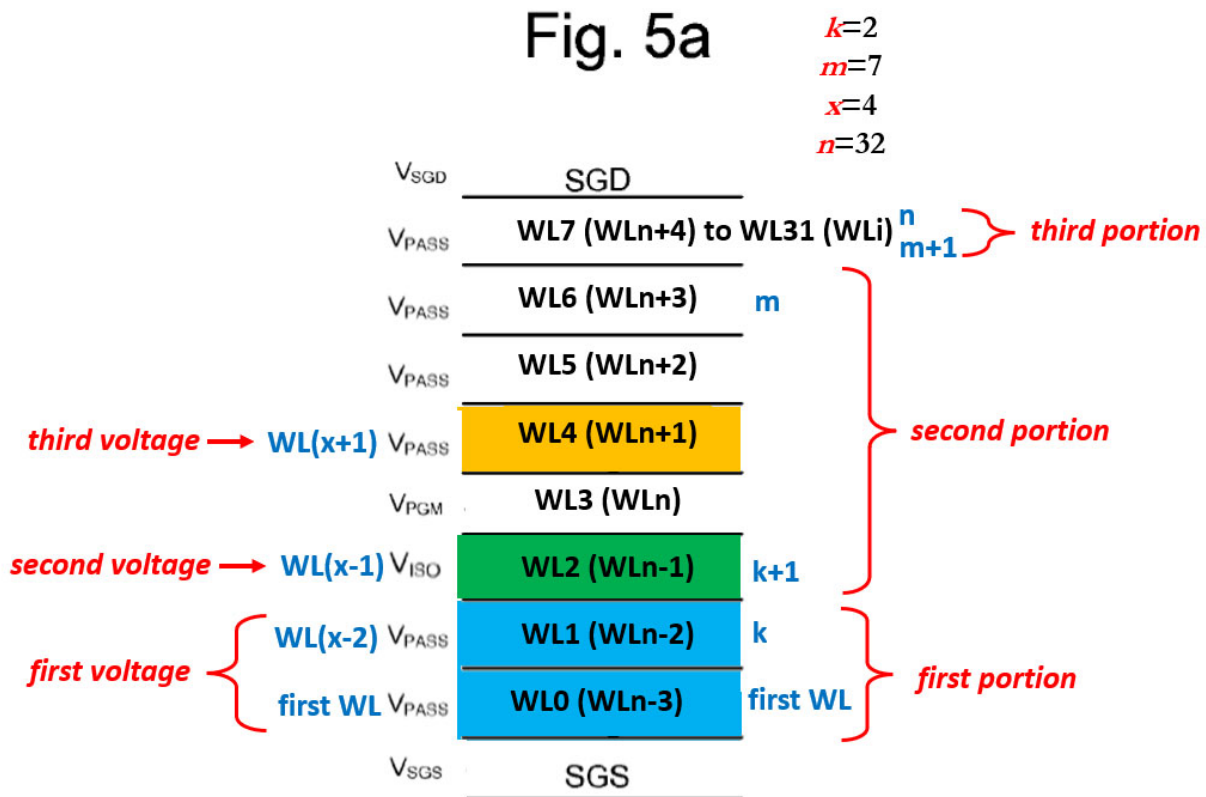
3. Claim 7

143. Claim 7 is a dependent claim and depends from claim 5. As discussed below, Dong satisfies claim 7.

a. [7.PRE] “The program method of claim 5, wherein the memory further comprises a third portion formed above the second portion, the third portion comprises an $(m+1)$ th wordline to an n th wordline from bottom to top, and the program method further comprises:”

144. I offer no opinion on whether the preamble is limiting. I was instructed by counsel to analyze the preamble as if it was a limitation. In my opinion, Dong discloses the preamble.

145. Continuing with the same 32-wordline memory string for Figure 5a where WL3 is being used for programming, an example *third portion* (e.g., from WL7 to WL31) is shown below. As is apparent from the figures, the *third portion* is *above the second portion*. Recall that in the example being used, $m=7$. Claim 7 introduces another variable (n) which it uses to define the upper bound of the *third portion*. It is important in this regard to note that the claimed variable n is different from the variable “ n ” in Dong, which is the variable Dong uses to identify the wordline used for programming. Again, to make it possible to analyze a specific example, the claimed variable n can be taken as $n=32$. Using this example, Dong’s *third portion comprises an $(m+1)$ th wordline (WL7) to an n th wordline (WL31) from bottom to top*, which satisfies [7.PRE].



Dong, Figure 5a (annotated).

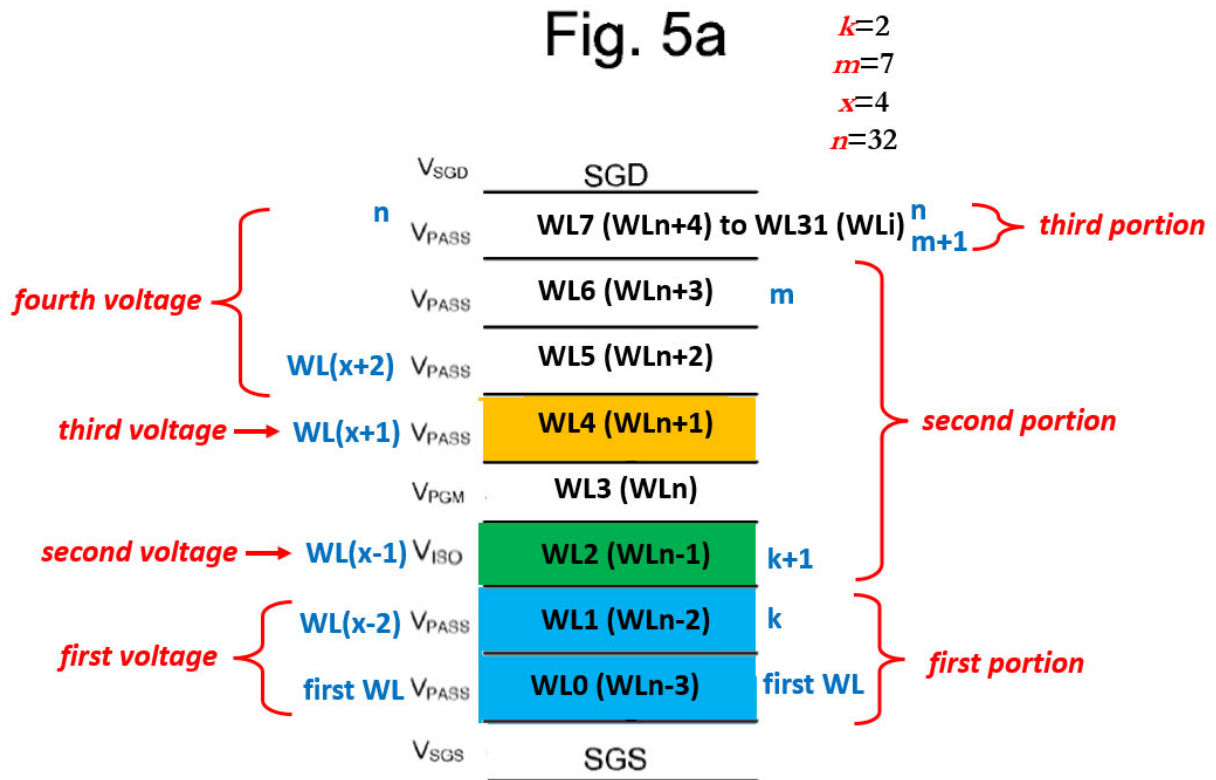
b. [7.A] “applying a fourth voltage to the (x+2)th wordline to the nth wordline;”

146. In my opinion, Dong discloses claim element [7.A]. The **fourth voltage** in Dong is the V_{PASS} voltage applied to WL5 through WL31 in the example of Figure 5a being used. As I have already discussed for the **first**, **second**, and **third voltages**, there is likewise no requirement that the **fourth voltage** be a different voltage than any of the previously recited voltages. Neither the '941 patent specification nor claims say that, or even given an example where the voltages are different. Rather, the only requirement is simply that the same **fourth voltage** (whatever voltage it is)

be applied to the specified wordlines (*i.e.*, *to an (x+2)th wordline to the mth wordline*).

147. Independent claim 5, on which claim 7 depends, specifies that the *xth wordline is used to perform a program operation*. And as noted throughout my analysis, Dong uses the variable “n” to identify the wordline being used for programming. *See* Dong, [0049]. This means that the *(x+2)th* wordline maps to $WLn+2$ (*viz.*, WL5) in Dong Figure 5a. Given that $n=32$ in the example being used, the *nth wordline* as is the 32nd wordline, which maps to wordline WLi (*viz.*, WL31) in Dong Figure 5a.

148. Applying these wordline assignments to Dong Figure 5a, WL5 through WL31 must receive the *fourth voltage*. In Figure 5a of Dong $WLn+1$ through WLi (*viz.*, WL31, which is the uppermost wordline) receives V_{PASS} . *See* Dong, [0049] (“a pass voltage, V_{PASS} , [] is applied to each of the unselected wordlines, WL0 through $WLn-2$ and $WLn+1$ through WLi ”). Accordingly, wordlines $WLn+2$ (*viz.*, WL5) through WLi (*viz.*, WL31) receive the V_{PASS} voltage. In summary, Dong discloses *a fourth voltage* (V_{PASS}) is applied *to an (x+2)th wordline* (WL5) *to the mth wordline* (WL31), as shown below.



Dong, Figure 5a (annotated).

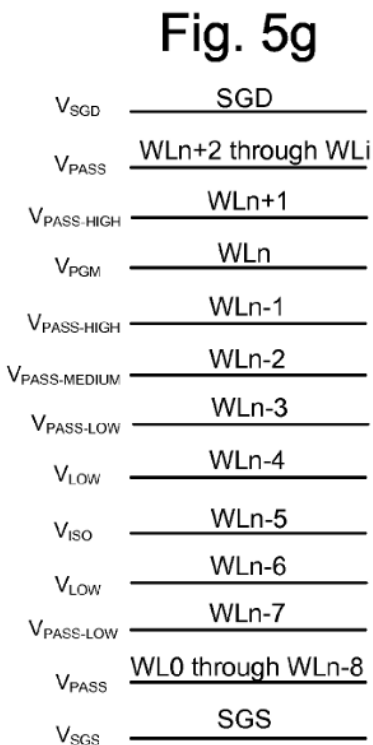
c. [7.B] “wherein $m < n$, and $x < (n-1)$.”

149. In my opinion, Dong discloses claim element [7.B]. In the specific example being used with Dong Figure 5a, WL3 (the fourth wordline) is being used for programming, which means $x=4$. The values for the other variables being used are $m=7$ and $n=32$. Accordingly, $n-1=31$. Using these values, simple arithmetic confirms that $m < n$ (i.e., $7 < 31$) and $x < (n-1)$ (i.e., $4 < 31$) in Dong Figure 5a. Thus, Dong satisfies claim element [7.B].

4. Claim 1

150. With respect to claim 1, it is helpful to note that it corresponds to Figure 8 in the '941 patent. It also reads on other Figures in the '941 patent but it is not necessary to discuss them.

151. Dong by itself both anticipates and renders claim 1 obvious. As I mentioned in §X.A, the embodiment in Dong that maps to claim 1 (and 11) is Figure 5g. The same background points I explained in §X.A.1 concerning Dong Figure 5a apply to Figure 5g (*e.g.*, the concept of variables, identifying wordlines by their relative locations to other wordlines/variables, the meaning of SGS and SGD, the meaning of *bottom to top*, etc.). Dong Figure 5g is shown below.



Dong, Figure 5g.

152. Even though claim 1 does not recite any specific number of wordlines, as was the case for Dong Figure 5a and for the same reasons, I utilize the specific example of a 32-cell memory string for Dong Figure 5g throughout my analysis of claim 1. I note that other memory string lengths, including much longer memory strings, would certainly have been obvious to a POSITA by the 2020 Priority Date of the '941 patent to achieve higher memory densities.

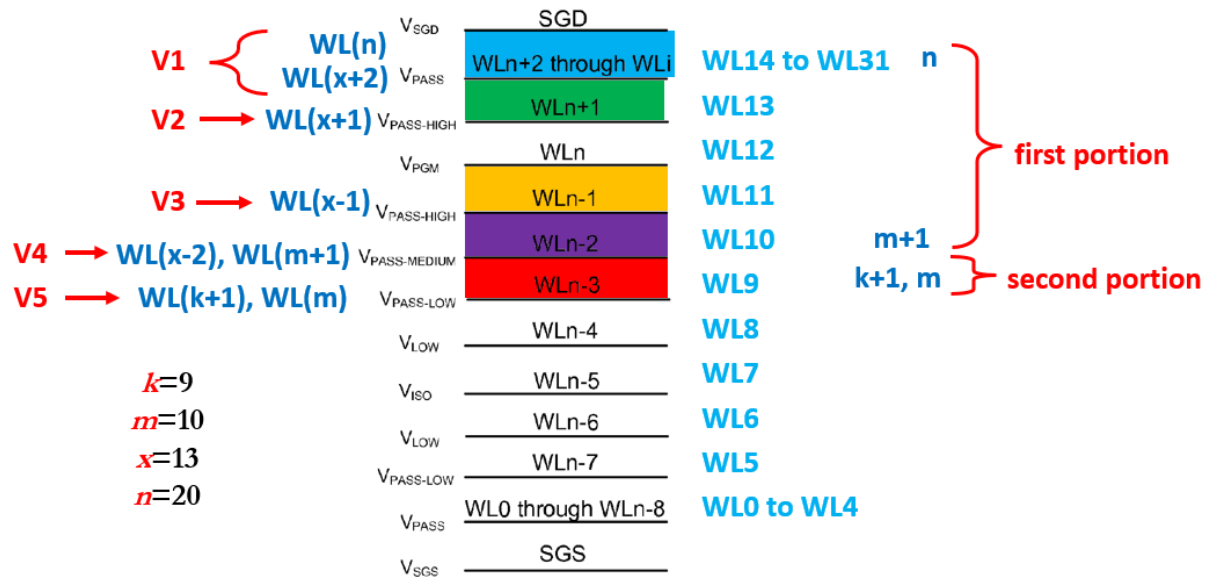
153. In this example, I use WL12 (which is the 13th wordline from the bottom) as the wordline being used for programming (which is WLn in Dong's verbiage), but I note that programming typically progresses one wordline at a time until all memory string cells are programmed. Dong, Figure 10 (especially steps 1060, 1065, and 1075), [0089]. In the specific example I use, the wordlines have the wordline assignments and voltage labels shown below in annotated Figure 5g.



Dong, Figure 5g (annotated).

154. The image below simply adds colors and additional labels to the figure above to show how Dong Figure 5g maps to claim 1. I discuss each claim element in detail below.

Fig. 5g



Dong, Figure 5g (annotated). I also found it helpful in conducting my analysis to locate all of the variables, etc. being sued in one place for reference. I wish to emphasize that the mappings below are just examples and that other mappings could also satisfy claim 1.

$k = 9$	$WL(n) = WL19$	$WL(x+2) = WL(n+2) = WL14$
$m = 10$	$WL(m+1) = WL10$	$WL(x+1) = WL(n+1) = WL13$
$x = 13$	$WL(m) = WL9$	$WL(x) = WL(n) = WL12$
$n = 20$	$WL(k+1) = WL9$	$WL(x-1) = WL(n-1) = WL11$
		$WL(x-2) = WL(n-2) = WL10$

a. [1.PRE] “A memory comprising:”

155. I offer no opinion on whether the preamble is limiting. I was instructed by counsel to analyze the preamble as if the it was a limitation. In my opinion, Dong

discloses the preamble for the same reasons I explained in connection with the preamble of claim 5, which also recites a *memory*.

b. [1.A] “a first portion comprising an (m+1)th wordline to an nth wordline from bottom to top; and”

156. In my opinion, Dong discloses claim element [1.A]. As I explained in connection with claim 5, the term *portion* is not a term of art, and I afford the term its plain and ordinary meaning in view of how the claim uses it. In particular, claim element [1.A] says that the *first portion compris[es] an (m+1)th wordline to an nth wordline from bottom to top*. My discussion concerning the meaning of *bottom to top* and the usage of variables (e.g., *m*, *n*, *k*, and *x*) for the preamble of claim 5 applies equally here.

157. Regarding the use of variables in claim 1, the only limitations the claim places on them are that *x, k and m are integers, $0 \leq k < m$, $(m+2) < x < (n-1)$* , as claim element [1.I] states. I discuss how Dong Figure 5g meets these conditions in my analysis of claim element [1.I]. But because claim element [1.A] refers to the variables *m* and *n*, I note that *m* can be assigned the value 10, and *n* can be assigned the value of 20 to allow for analyzing a concrete example. With these assignments, Dong’s *(m+1)th wordline* maps to WL10 (viz., WLn-2). This is the eleventh wordline from the bottom because of Dong’s numbering convention which begins with first wordline as WL0 instead of WL1. Dong’s *nth wordline from bottom to*

top maps to WL19. These mappings result in the **first portion** (viz. WL10 to WL19) of Dong Figure 5g as shown below.



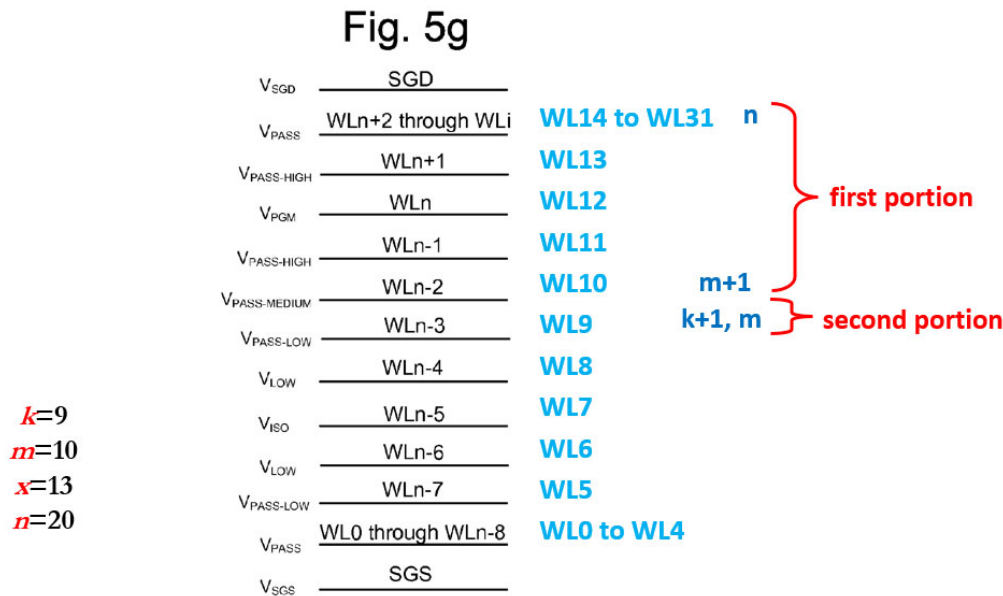
Dong, Figure 5g (annotated).

- c. [1.B] “a second portion formed below the first portion and comprising a (k+1)th wordline to an mth wordline from bottom to top; and”*

158. In my opinion, Dong discloses claim element [1.B]. As noted for the previous element, term **portion** is not a term of art. That is true regardless of whether it is a **first portion**, **second portion**, or any other **portion**. But the claimed itself specifies that the **second portion** comprises **a (k+1)th wordline to an mth wordline from bottom to top**.

159. I continue to use the same specific example for Dong Figure 5g throughout claim 1 in which $m=10$ (which maps to WL9) and $n=20$ (which maps to

WL19). Additionally, in the example being used, $k=9$. This mean that the $(k+1)th$ *wordline* in Dong Figure 5g is WL9 (*viz.*, $WLn-3$). This is the tenth wordline from the bottom because of Dong’s numbering convention in which the first wordline is WL0 instead of WL1. Putting this all together results in the *second portion* of Dong Figure 5g spanning one wordline, *viz.*, from WL9 to WL9 as shown below.



Dong, Figure 5g (annotated).

160. I make two observations in regards to the *second portion* spanning one wordline. First, there is no requirement in the claim that the *first* and *second portions* be of the same size (*i.e.*, contain the same number of wordlines). And a POSITA would understand that they need not be the same size. This is confirmed by other references, such as Costa, which teaches that “the size of the zones may not be equal or otherwise contain an equal number of wordlines.” Costa, [0116].

161. Second, any notion that the *second portion* must contain more than one wordline is misplaced for several reasons. To begin, the claim itself does not set any upper or lower size limit on the *second portion*. To the contrary, the claim allows for a single wordline by specifying the conditions of the variables, which are used to define the boundaries of the *second portion*, that must be met (*see* claim element [1.I]). As long as these conditions are met, the *second portion* may be satisfied. I discuss those conditions further (and how Dong Figure 5g meets them) in my analysis of claim element [1.I]. But I note here that the Dong example below satisfies those conditions. In summary, both *m* and *k* are integers, and the condition $0 \leq k < m$ is satisfied because $0 \leq 9 < 10$, which satisfies the claimed inequality. The wordline being used for programming is *x* according to the claim. In the example being used, $x=13$, which maps to WL12 (*viz.*, WL_n) in Dong Figure 5g. Accordingly, the condition $(m+2) < x < (n-1)$ is satisfied because $12 < 13 < 19$ meets the claimed inequality.

162. Another reason the *second portion* can be a single wordline is apparent by contrast with claim element [3.A], which refers to *a third portion formed above the first portion and comprising a plurality of wordlines*. This shows that when the patentee wanted to require that a *portion* include more than one wordline, it said so. But claim element [1.B] has no language about a “plurality” of wordlines. Instead,

it just sets conditions for the variables that define the *second portion* as discussed in the previous paragraph.

163. Yet another reason the *second portion* can be a single wordline is because that is how a POSITA would understand the term. This understanding is reflected in other references such as Costa, which says that “[a]ny number of zones may be used, including a zone for each wordline.” Costa, [0116]; *see also id.*, [0132], [0162] (“each wordline may be its own zone”). I note that Costa is a good example because its discussion is in the specific context of memory string programming in NAND flash memory.

164. Returning now to the balance of claim element [1.B], it additionally requires that the *second portion* is *formed below the first portion*. Dong discloses this. Dong’s *second portion* (*viz.*, WL9) is below the *first portion*, which (as explained for claim element [1.A]) spans WL10 to WL20. Accordingly, Dong satisfies claim element [1.B].

d. [1.C] “a controller configured to apply [various voltages]”

165. In my opinion, Dong satisfies claim element [1.C]. Dong Figure 12 is block diagram of an exemplary memory system architecture. Dong, Figure 12, [0027], [0095]. Dong’s “Controller 1250” is the claimed *controller*. Dong, Figure 12, [0096]-[0098] (describing controller operation), [0128] (“a ‘program’ command is issued by the controller”). A POSITA would have understood the role a controller

plays in reading, writing, and erasing memory cells and that it controls the voltages for doing so.

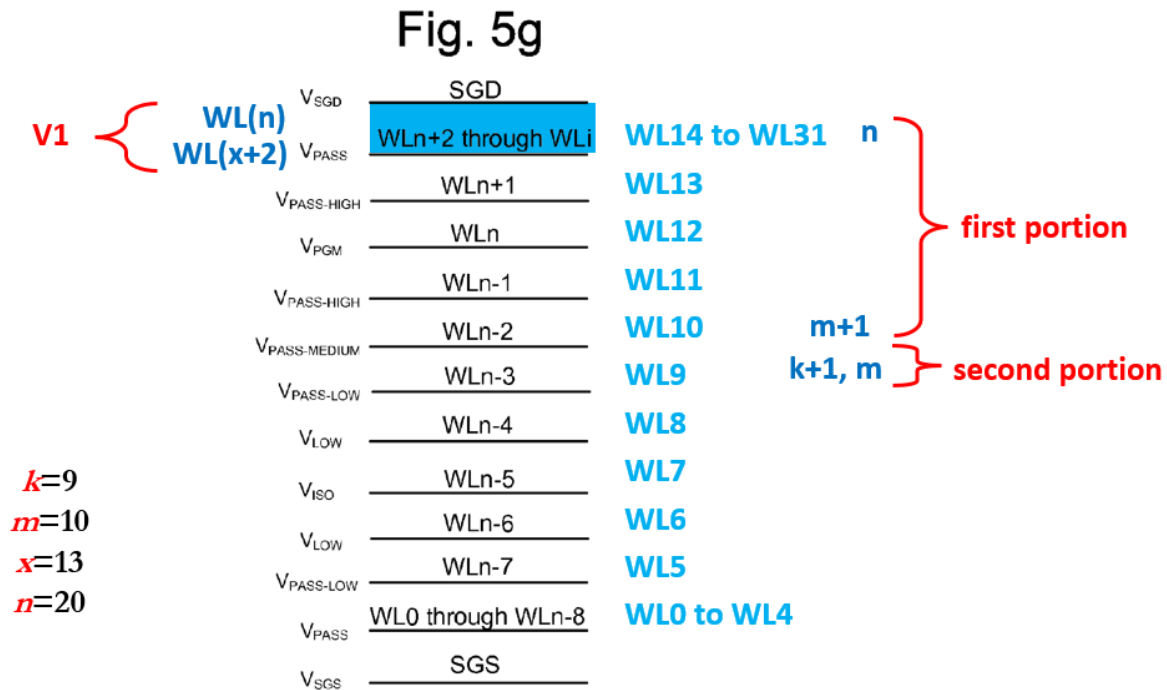
e. [1.D] “a first voltage to an (x+2)th wordline to the nth wordline,”

166. In my opinion, Dong discloses claim element [5.A]. As was the case for claim 5, claim 1 uses the variable *x* to specify which wordline is being *used to perform a program operation*. This is set forth in claim element [1.H] which says that *an xth wordline is used to perform a program operation*. And, as was the case for claim 5, the *xth* wordline in Dong is WLn (this is true for both Dong Figure 5a and Figure 5g). Dong, [0049] (“a program voltage, V_{PGM} , which is applied to the selected wordline WLn”); *see also id.*, [0048] (“the selected wordline is WLn”).

167. With this in mind, claim element [1.D] requires that a *first voltage* be applied *to an (x+2)th wordline to the nth wordline*. Dong meets this requirement. The claimed *first voltage* corresponds to V_{PASS} in Dong Figure 5g and is applied from WL14 to WL19), as I explain further below.

168. The claimed *first voltage* in Dong Figure 5g is V_{PASS} . In the specific example being used, WLn (the wordline being used for programming) maps to WL12. This means that the *(x+2)th wordline* in Dong is WL14 (*viz.*, WL(n+2)). Given that *n*=20 (which corresponds to WL19), the *first voltage* (V_{PASS}) must be applied to wordlines WLn+2 (*viz.*, WL14, the *(x+2)th wordline*) to WL19 (the *nth*

wordline). Dong Figure 5g below shows this. V_{PASS} is applied to wordlines WL14 through WL19. The fact that V_{PASS} is also applied to additional wordlines is irrelevant and not precluded by the claim.



Dong, Figure 5a (annotated)

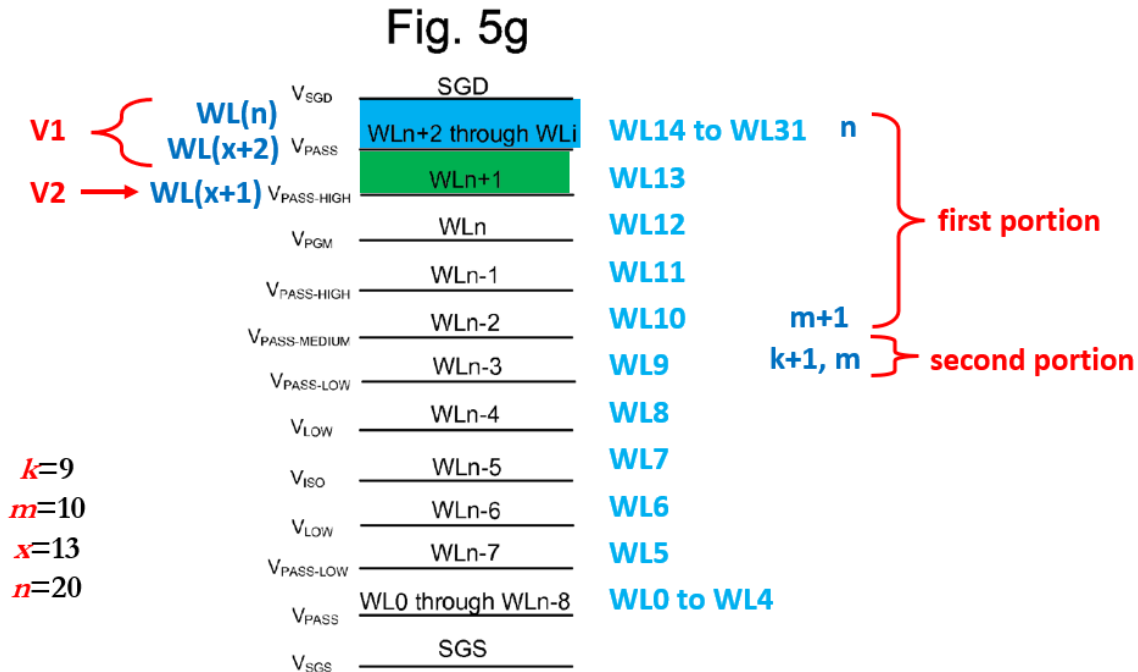
169. Dong explains in connection with Figure 5f that “ V_{PASS} is applied to a second group of storage elements associated with WLn+1 through WLi.” Dong, [0058]. In relevant part, this also holds true for Dong Figure 5g. In particular, Dong explains that the voltages and operation of Figure 5g are the same as Figure 5f, except that “(WLn+1) adjacent to the selected wordline (WLn) receives $V_{PASS-HIGH}$ instead of V_{PASS} .” *Id.*, [0059]. Accordingly, Dong’s statement regarding V_{PASS} means that V_{PASS} is applied from WL14 (which is WL(n+2) in the Figure 5g example

being used and maps to the *(x+2)th wordline* through WL19 (which is the *nth wordline* since $n=20$ and Dong numbers its wordlines starting with WL0). Indeed, a POSITA would understand from Dong's disclosure that V_{PASS} is also applied through WL31 (*viz.*, WL i) in the Dong Figure 5g example. But this is not prohibited by the claim and is irrelevant.

170. As was the case for claim 5, neither claim 1 nor the '941 patent specification specific any particular voltage level (or range of voltages) for the claimed *first voltage*. See generally Ex. 1001. The only thing required is that the same voltage (whatever voltage it may be) is applied to the claimed *(x+2)th wordline to the nth wordline*. Because V_{PASS} is applied in Dong Figure 5g to the appropriate wordlines, Dong satisfies claim element [5.D].

f. [1.E] "a second voltage to an (x+1)th wordline,"

171. In my opinion, Dong satisfies claim element [1.E]. The claimed *second voltage* in Dong Figure 5g is $V_{\text{PASS-HIGH}}$. This voltage is applied *to an (x+1)th wordline*, which in Dong Figure 5g example being used is WL $n+1$ (which maps to WL13). Dong, Figure 5a, [0059] ("the drain-side wordline (WL $n+1$) adjacent to the selected wordline (WL n) receives $V_{\text{PASS-HIGH}}$ "). This is illustrated below in Dong Figure 5g. Thus, Dong satisfies claim element [1.D]

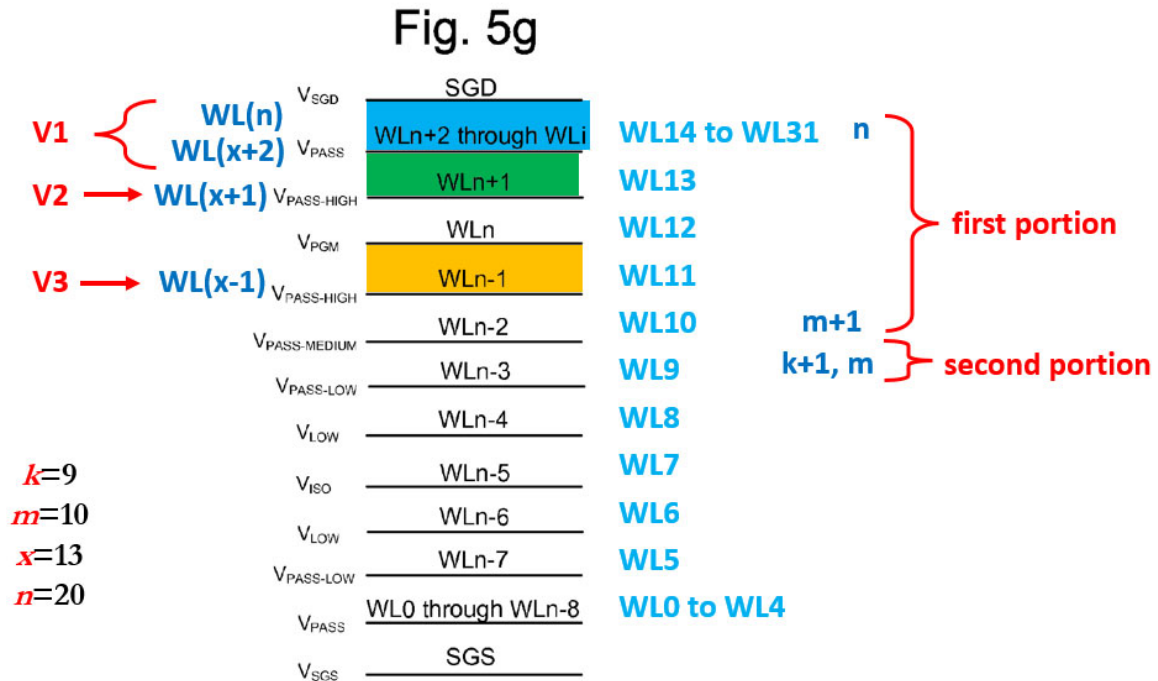


Dong, Figure 5g (annotated).

g. [1.F] “a third voltage to an $(x-1)$ th wordline,”

172. In my opinion, Dong discloses claim element [1.F]. The claimed **third voltage** in Dong is $V_{PASS-HIGH}$. This voltage is applied **to an $(x-1)$ th wordline**, which in Dong Figure 5g corresponds to $WLn-1$ (which, in turn, maps to $WL11$). With reference to Figure 5f, Dong explains that “the source-side wordline ($WLn-1$) adjacent to the selected wordline (WLn) receives $V_{PASS-HIGH}$.” Dong, [0058]. But this statement applies equally to Figure 5g. Not only is it shown in Figure 5g, but Dong explains that the only difference between Figures 5g and 5f relates to the voltage applies to the $WL(n+1)$ wordline, which is not at issue in claim element

[1.F]. See Dong, [0058]-[0059]. Accordingly, Dong satisfies claim element [1.F] as shown below.



Dong, Figure 5g (annotated).

173. Any notion that the actual voltages of the claimed *first voltage* and *third voltage* must be different would be incorrect. Nothing in claim 1 or the '941 patent specification require that they be different. Indeed, the '941 patent discloses no actual voltages at all. See generally Ex. 1001. And I note that when the patentee wanted to require a relationship between he claimed *voltages*, the claims included language to that effect. As an example, claim element [1.I] states that *the fifth voltage is lower than the fourth voltage*. I address that requirement in my analysis of element [1.I]. Here, however, claim element [1.F] does not impose a similar type

of requirement. Accordingly, it is irrelevant whether $V_{\text{PASS-HIGH}}$ is also applied to the wordlines receiving the *first voltage*, and Dong satisfies element [1.F].

h. [1.G] “a fourth voltage to the (m+1)th wordline to an (x-2)th wordline, and”

174. In my opinion, Dong discloses claim element [1.G]. The claimed *fourth voltage* in Dong Figure 5g is $V_{\text{PASS-MEDIUM}}$. This voltage is applied from *the (m+1)th wordline* (which in the Dong example being used is WL10 because $m=10$ and Dong numbers its wordlines beginning with WL0) *to an (x-2)th wordline* (which in the Dong example being used is WL(n-2), which maps to WL10 because the wordline used for programming (WLn) is WL12). *See* Dong, Figure 5g, [0049]. This mapping is shown below in Figure 5g. With reference to Figure 5f, Dong explains that “the next wordline (WLn-2) receives $V_{\text{PASS-MEDIUM}}$.” Dong, [0058]. But this statement applies equally to Figure 5g. Not only is it shown in Figure 5g, but Dong explains that the only difference between Figures 5g and 5f relates to the voltage applies to the WL(n+1) wordline, which is not at issue in claim element [1.F]. *See* Dong, [0058]-[0059].

175. In summary, the specified wordline(s) that the *fourth voltage* must be applied to is just WL10. This conclusion is easily derived from a simple algebraic transformation of the relationship specified by the ‘941 claim. Claim limitation has a specific relationship between m, x and n:

$$(m + 2) < x < (n - 1)$$

The first part of this condition could be met when:

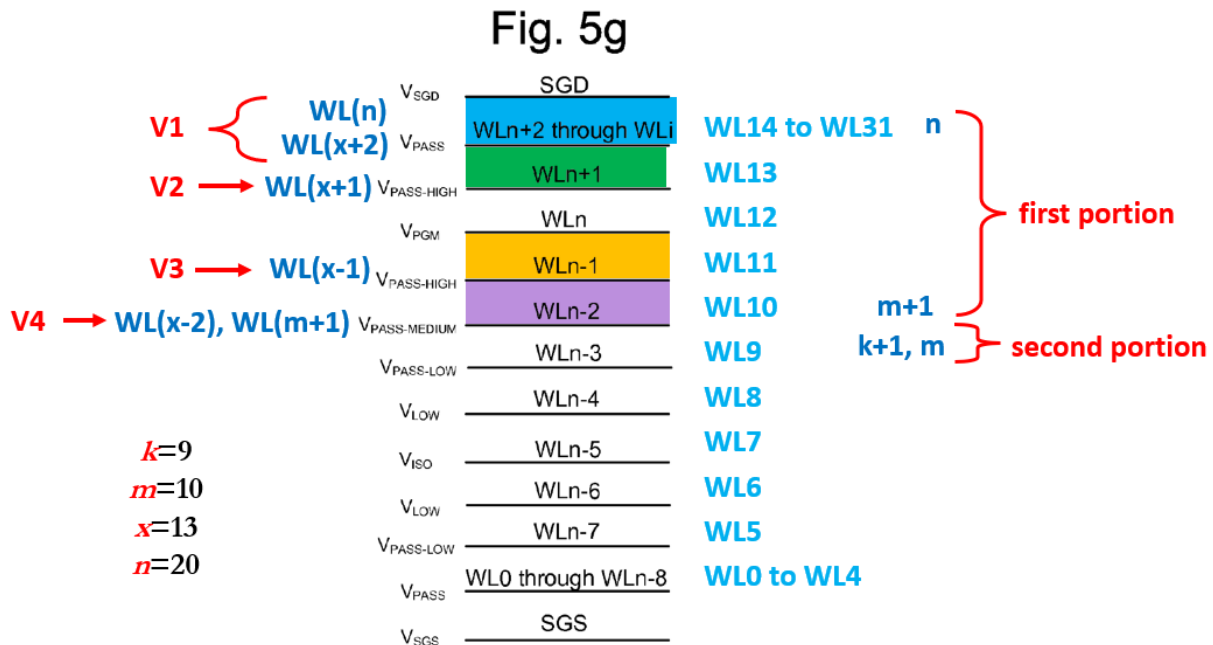
$$(m + 2) = (x - 1)$$

Which is equivalent to:

$$m = x - 3$$

$$m + 1 = x - 2$$

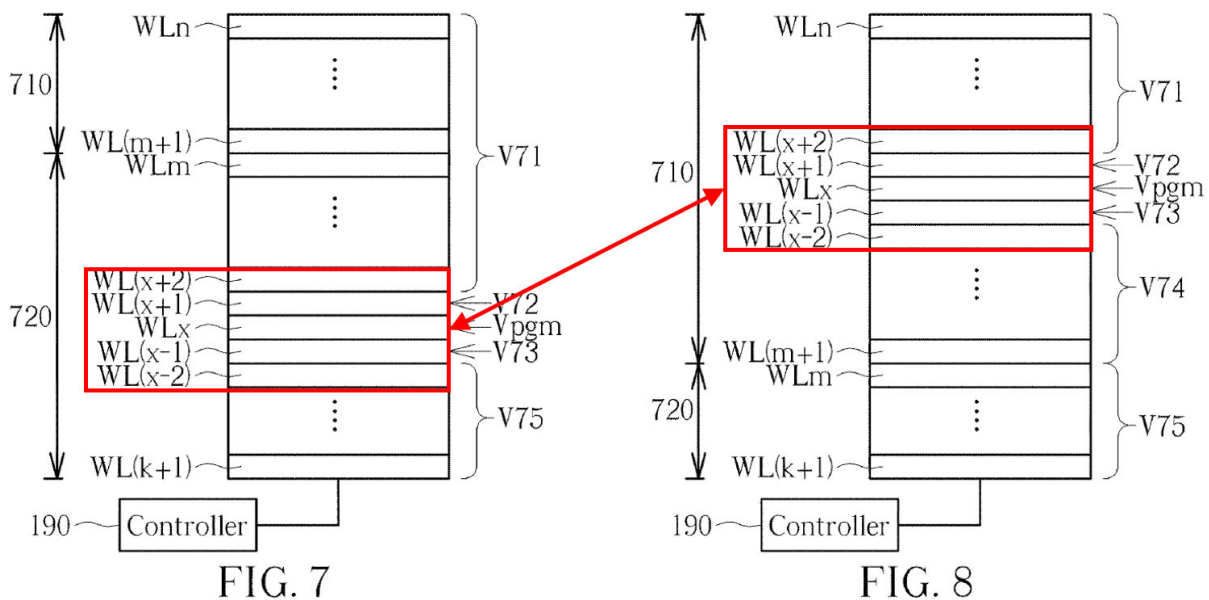
In this case, the range of $m+1$ to $x-2$ spans exactly just one wordline. As seen below, in Dong Figure 5g, the *fourth voltage* ($V_{\text{PASS-MEDIUM}}$) is applied to WL10 and satisfies claim element [1.G].



Dong, Figure 5g (annotated).

176. Any notion that the *fourth voltage* must be applied to more than one wordline would be incorrect for similar reasons to those I discussed regarding the

second portion in claim element [1.B]. I additionally note here that the '941 specification further confirms this point. In particular, the '941 specification illustrates a programming operation occurring at two different memory cell locations in Figures 7 and 8. In Figure 7, the wordline used for programming (WL_x) is located in second portion 720. Ex. 1001, 4:29-36. In Figure 8, the wordline used for programming (WL_x) is located in first portion 710. Ex. 1001, 4:29-36.



Ex. 1001, Figures 7-8 (annotated). A POSITA would have understood that memory string programming typically occurs sequentially one cell at a time. Thus, as programming advances from Figure 7 to Figure 8, WL_x moves up the memory string one wordline at a time. As programming continues from the second portion 720 (where WL_x and its adjacent wordlines are all beneath $WL(m+1)$) to the first portion 710 (where WL_x and its adjacent wordlines are all above $WL(m+1)$), WL_x will

advance through the *(m+1)th wordline* and beyond (as Figure 8 shows). When it does, the *(x-2)th wordline* will necessarily be the same wordline as the *(m+1)th wordline*, which means range that comprise the fourth voltage will span a single wordline.

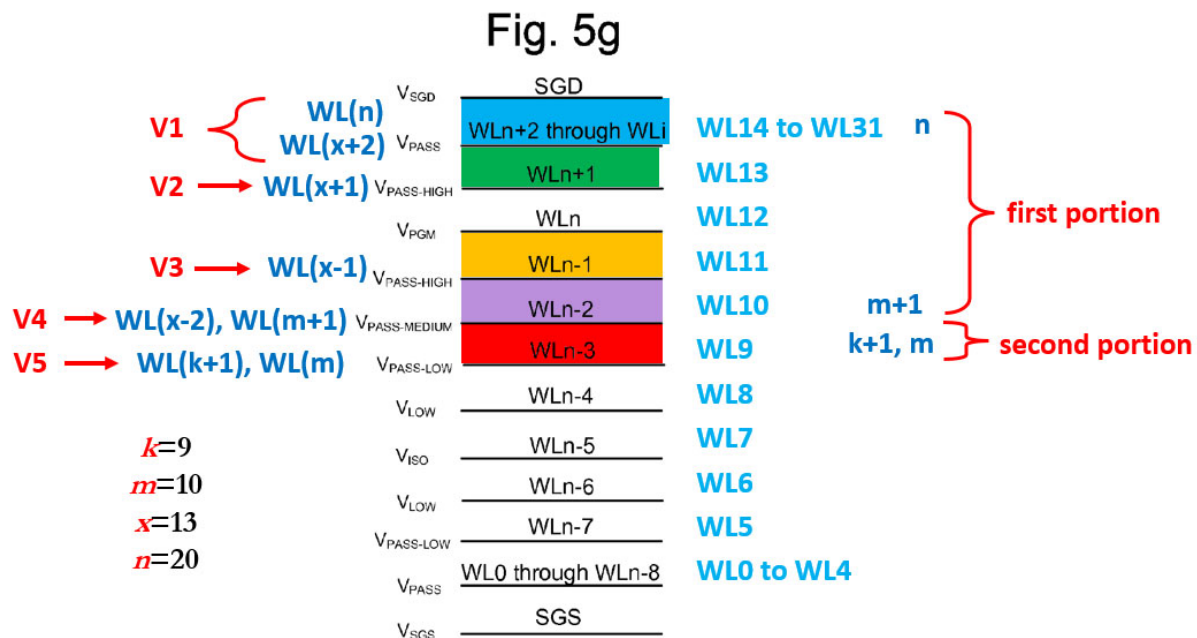
- i. *[1.H] “a fifth voltage to the (k+1)th wordline to the mth wordline when an xth wordline is used to perform a program operation;”*

177. In my opinion, Dong satisfies claim element [1.H]. To begin, as I noted multiple times, the *xth wordline [] used to perform a program operation* maps to WLn in Dong Figure 5g. *See* Dong, Figure 5g, [0049] (“a program voltage, V_{PGM} , which is applied to the selected wordline WLn”). The *fifth voltage* in Dong Figure 5g is $V_{PASS-LOW}$. Dong, Figure 5g. With reference to Figure 5f, Dong explains that “the next wordline (WLn-3) receives $V_{PASS-LOW}$ ” Dong, [0058]. But this statement applies equally to Figure 5g. Not only is it shown in Figure 5g, but Dong explains that the only difference between Figures 5g and 5f relates to the voltage applies to the WL(n+1) wordline, which is not at issue in claim element [1.H]. *See* Dong, [0058]-[0059].

178. As I have explained, in the Figure 5g example being used, $k=9$ and $m=10$. It follows that the *(k+1)th wordline* in Dong is the tenth wordline from the bottom, which is WL9 (*viz.*, WLn-3) because Dong’s first wordline is WL0 instead of WL1. Likewise, when $m=10$, the corresponding wordline in Dong Figure 5g is

WL9. As such, the wordline(s) to which the *fifth voltage* must be applied is only WL9. For the same reasons I have explained why neither the *second portion* nor the *fourth voltage* must span multiple wordlines, there is likewise no requirement that the *fifth voltage* must span more than one wordline.

179. In summary, Dong discloses applying a *fifth voltage* ($V_{\text{PASS-LOW}}$) to the *(k+1)th wordline* (WL9 in the Figure 5g example being used) to the *mth wordline* (also WL9 as explained above) and thus discloses claim element [1.H]. This is shown in Dong Figure 5g below.



Dong, Figure 5g (annotated).

- j. [1.I] “wherein x , k and m are integers, $0 \leq k < m$, $(m+2) < x < (n-1)$, and the fifth voltage is lower than the fourth voltage.”

180. In my opinion, Dong discloses claim element [1.I]. Before addressing the inequalities, I begin with the requirement that *the fifth voltage is lower than the fourth voltage*. This is the first time claim 1 imposes any sort of restriction on the voltage level. Even here, there are no actual voltages specified. Rather, it is only a relative requirement. I previously explained at claim element [1.G] that Dong’s *fourth voltage* is $V_{\text{PASS-MEDIUM}}$ in Figure 5g. And I explained at claim element [1.H] that Dong’s *fifth voltage* is $V_{\text{PASS-LOW}}$ in Figure 5g.

181. In connection with Dong’s discussion of various V_{PASS} that “can be used at the same time,” Dong explains that “multiple wordlines [] are biased to different V_{PASS} values, e.g., $V_{\text{PASS-LOW}}$, $V_{\text{PASS-MEDIUM}}$ and $V_{\text{PASS-HIGH}}$. In one implementation, $V_{\text{PGM}} > V_{\text{PASS-HIGH}} > V_{\text{PASS-MEDIUM}} > V_{\text{PASS-LOW}} > V_{\text{LOW}} > V_{\text{ISO}}$.” Dong, [0057] (emphasis added). Thus, by Dong’s own disclosure, Dong’s *fifth voltage* ($V_{\text{PASS-LOW}}$) is less than Dong’s *fourth voltage* ($V_{\text{PASS-MEDIUM}}$), with satisfies the relative voltage requirement of claim element [1.I].

182. I now turn to an analysis of the variables and inequalities. For ease of reference, I repeat here the values for each variable that I have used throughout my analysis of claim 1: $x=13$, $k=9$, $m=10$, and $n=20$. Again, the reason for using a specific example was for ease of understanding since the claim was otherwise

somewhat abstract. The variables x , k , and m in Dong are all integers, as I have discussed previously. Now, the only thing left to confirm is that Dong Figure 5g satisfies the two claimed inequalities.

183. Before evaluating the inequalities using specific values, I note that neither the '941 patent nor Dong provide actual values/numbers for all the wordlines used during a program operation (*e.g.*, wordlines associated with or surrounding the variables x , in the '941 patent and the variable “n” in Dong). Rather, both use relative offset like “+1” and “-1” to identify wordlines. I further note that the '941 patent does not give specific values for the variables k , m , and n , even by way of example. Likewise, Dong does not limit the value that the variable “i” may take, which denotes the top cell in a memory string. Thus, Dong satisfies element [1.I] in the same way as disclosed in the '941 patent. Regardless, I continue to apply the same specific example for Dong Figure 5g of a 32-cell memory string where WLn (the wordline used for programming) is WL12.

184. The first claimed inequality requires that $0 \leq k < m$. Substituting the values from the specific example being used for Dong Figure 5g into the inequality produces the following result: $0 \leq 9 < 10$. This is a true statement according to basic math principles. Accordingly, the substitution satisfies the claimed inequality that $0 \leq k < m$.

185. The second inequality requires that $(m+2) < x < (n-1)$. Because $m=10$, $x=13$, and $n=20$ in the Dong Figure 5g example being used, $m+2=12$ and $n-1=19$. This is nothing more than simple math. Substituting these values into the inequality results in the following: $12 \leq 13 < 19$. This is a true statement according to basic math principles. Accordingly, the substitution satisfies the inequality that $(m+2) < x < (n-1)$. In summary, both claimed inequalities are satisfied and Dong discloses element [1.I].

5. Claim 3 “The memory of claim 1, further comprising”

186. Claim 3 is a dependent claim and depends from claim 1. Because Dong satisfies claim 1 as discussed above, the preamble of claim 3 is met by Dong as well for the same reasons. Claim 3 has two additional elements. As discussed below, Dong satisfies both of them.

a. [3.A] “a third portion formed above the first portion and comprising a plurality of wordlines”

187. In my opinion, Dong discloses element [3.A]. As previously noted with respect to the *first* and *second portion* limitations, the term *portion* is not a term of art. Nor is the term *third portion* a term of art. Rather, as element [3.A] recites, it must simply *compris[e] a plurality of wordlines* located *above the first portion*.

188. The claimed *third portion* differs from the *first* and *second portion* limitations in two respects. First, unlike the *first* and *second portions*, the claimed

third portion requires *a plurality of wordlines*, and cannot be satisfied by just a single wordline. Second, claim element [3.A] does not use a variable to identify where the *third portion* starts or stops. Regarding which wordlines must comprise the *third portion*, the only requirement is that it is located *above the first portion*. Even then, the claim does not specify that the *third portion* be immediately above the *first portion*. Regardless, even under such a construction, Dong meets element [3.A].

189. As I discussed for claim element [1.A], in the specific example used for claim 1, Dong's *first portion* spans WL10 to WL19. Accordingly, the *third portion* in Dong Figure 5g has to be located above WL19. Recall that the Dong example being used has 32 bit memory strings controlled by wordlines WL0 to WL31. Because the *third portion* must have at least two wordlines, the *third portion* can comprise WL20 to WL21 (or any other wordlines up to WL31 for that matter), which are all located *above* the *first portion*. Thus, Dong discloses element [3.A].

b. [3.B] “wherein the controller is further configured to apply the first voltage to the plurality of wordlines when the xth wordline is used to perform the program operation.”

190. In my opinion, Dong discloses claim element [3.B]. Claim elements [1.C] and [1.H] already collectively recite *a controller configured to apply [various recited voltages] ... when an xth wordline is used to perform a program operation*.

Thus, I have already addressed these aspects of claim element [3.B] and how Dong satisfies them in my discussion of claim elements [1.C] and [1.H].

191. This leaves only the requirement that the *first voltage* is applied to *the plurality of wordlines*, which is the same *plurality of wordlines* introduced in claim element [3.A] (*i.e.*, the *first voltage* must be applied to the *third portion*). As discussed for claim element [3.A], in the specific example being used, the *plurality of wordlines* in the Dong Figure 5g is WL20 to WL21. And the claimed *first voltage* in Dong Figure 5g is V_{PASS} , as I discussed in connection with claim element [1.D].

192. Dong states that “ V_{PASS} is applied to a second group of storage elements associated with $WLn+1$ through WLi .” Dong, [0058]. While this statement is describing Dong Figure 5f, it is equally applicable (in relevant part) to Dong Figure 5g. Dong itself makes this clear when it explains that Figure 5g only “differs from that of FIG. 5f in that the drain-side wordline ($WLn+1$) adjacent to the selected wordline (WLn) receives $V_{\text{PASS-HIGH}}$ instead of V_{PASS} .” *Id.*, [0059]. Here, claim element [3.B] is not concerned with the $WLn+1$ wordline (*viz.*, WL13), and Dong’s statement that V_{PASS} is applied to the wordlines at “ $WLn+1$ through WLi ” applies to Dong’s *plurality of wordlines* (*viz.*, WL20 to WL21) that form the *third portion*. Dong, [0058]. This same holds true for wordlines WL22 through WL31 (which maps to WLi in Dong Figure 5g) if those wordlines are considered part of the *third*

portion. In summary, because, V_{PASS} is applied to the requisite wordlines, Dong satisfies claim element [3.B].

6. Claim 11

193. Independent claim 11 is a method claim and claim 1 is an apparatus claim, but they are otherwise substantially similar to another claim 11 is satisfied by Dong for the same reasons as claim 1.

- a. [11.PRE] “A program method of a memory, the memory comprising a first portion and a second portion formed below the first portion, the first portion comprising an (m+1)th wordline to an nth wordline from bottom to top, the second portion comprising a (k+1)th wordline to an mth wordline from bottom to top, the program method comprising:”*

194. I offer no opinion on whether the preamble is limiting. I was instructed by counsel to analyze the preamble as if the it was a limitation. In my opinion, Dong discloses the preamble for the same reasons as [1.PRE], [1.A], and [1.B].

195. Even though claim 11 is a method claim, the preamble does not contain any method steps and just recites the same **first portion** (as called for by claim element [1.A]) and **second portion** (as called for by claim element [1.B]). The color-coded table below confirms that the balance of [11.PRE] is the same a [1.PRE].

Claim 1	Claim 11
[1.PRE] A memory comprising: [1.A] a first portion comprising an (m+1)th wordline to an nth wordline from bottom to top; and [1.B] a second portion formed below the first portion and comprising a (k+1)th wordline to an mth wordline from bottom to top; and	[11.PRE] A program method of a memory, the memory comprising a first portion and a second portion formed below the first portion, the first portion comprising an (m+1)th wordline to an nth wordline from bottom to top, the second portion comprising a (k+1)th wordline to an mth wordline from bottom to top, the program method comprising:

- a. *[11.A] “applying a first voltage to an (x+2)th wordline to the nth wordline when an xth wordline is used to perform a program operation;”*

196. The requirement of applying a ***first voltage to an (x+2)th wordline to the nth wordline*** in claim element [11.A] is the same as claim element [1.D] as can be seen in the table below and is thus met by Dong for the same reasons. The limitation ***when an xth wordline is used to perform a program operation*** in element [11.A] is present in element [1.H] and is met by Dong for the same reasons.

Claim 1	Claim 11
[1.D] a first voltage to an (x+2)th wordline to the nth wordline,	[11.A] applying a first voltage to an (x+2)th wordline to the nth wordline when an xth wordline is used to perform a program operation;

- b. *[11.B] “applying a second voltage to an (x+1)th wordline;”*

197. Claim element [11.B] is substantively identical to claim element [1.E] as shown in the table below and is met by Dong for the same reasons.

Claim 1	Claim 11
[1.E] a second voltage to an (x+1)th wordline,	[11.B] applying a second voltage to an (x+1)th wordline;

c. [11.C] “applying a third voltage to an (x−1)th wordline;”

198. Element [11.C] is substantively identical to claim element [1.F] as shown in the table below and is met by Dong for the same reasons.

Claim 1	Claim 11
[1.F] a third voltage to an (x−1)th wordline,	[11.C] applying a third voltage to an (x−1)th wordline;

d. [11.D] “applying a fourth voltage to the (m+1)th wordline to an (x−2)th wordline; and”

199. Claim element [11.D] is substantively the same as claim element [1.G] as shown in the table below, and Dong satisfies element [11.D] for the same reasons.

Claim 1	Claim 11
[1.G] a fourth voltage to the (m+1)th wordline to an (x−2)th wordline, and	[11.D] applying a fourth voltage to the (m+1)th wordline to an (x−2)th wordline; and

e. [11.E] “applying a fifth voltage to the (k+1)th wordline to the mth wordline;”

200. Element [1.H] contains substantively the same limitation as element [11.E] and [11.E] is met by Dong for the same reasons. The fact that claim element [1.H] recites the phrase *when an xth wordline is used to perform the program*

operation changes nothing since claim 11 (on which claim 13 depends) already requires the same thing.

Claim 1	Claim 11
[1.H] a fifth voltage to the (k+1)th wordline to the mth wordline when an xth wordline is used to perform a program operation;	[11.E] applying a fifth voltage to the (k+1)th wordline to the mth wordline;

f. [11.F] “wherein x , k and m are integers, $0 \leq k < m$, $(m+2) < x < (n-1)$, and the fifth voltage is lower than the fourth voltage.”

201. Claim element [11.F] is identical to element [1.I] as shown in the table below and is thus satisfied by Dong for the same reasons.

Claim 1	Claim 11
[1.G] wherein x , k and m are integers, $0 \leq k < m$, $(m+2) < x < (n-1)$, and the fifth voltage is lower than the fourth voltage.	[11.F] wherein x , k and m are integers, $0 \leq k < m$, $(m+2) < x < (n-1)$, and the fifth voltage is lower than the fourth voltage.

7. Claim 13: “The program method of claim 11, wherein the memory further comprises”

202. Claim 13 is a dependent claim and depends from claim 11. As discussed below, Dong satisfies claim 13 for the same reasons that Dong satisfies claim 3.

- a. [13.PRE] “The program method of claim 11, wherein the memory further comprises a third portion formed above the first portion, the third portion comprises a plurality of wordlines, and the program method further comprises:”***

203. I offer no opinion on whether the preamble is limiting. I was instructed by counsel to analyze the preamble as if it was a limitation. In my opinion, Dong discloses the preamble for the same reasons Dong satisfies reasons claim element [3.A]. As the side-by-side comparison table below shows, claim [13.PRE] is substantively the same as claim element [3.A]. This is true even though the preamble of claim 13 uses a few more words to say the same thing.

Claim 3	Claim 13
[3.A] a third portion formed above the first portion and comprising a plurality of wordlines	[13.PRE] a third portion formed above the first portion, the third portion comprises a plurality of wordlines, and the program method further comprises:

- b. [13.A] “applying the first voltage to the plurality of wordlines of the third portion.”***

204. Claim element [13.A] is satisfied by Dong for the same reasons that Dong satisfies claim element [3.B] because claim element [3.B] includes the same requirement as element [13.A], as can be seen in the table below. The fact that claim 3 repeats the phrase ***when the xth wordline is used to perform the program operation*** changes nothing since claim 11 (on which claim 13 depends) already requires the same thing.

Claim 3	Claim 13
[3.B] wherein the controller is further configured to apply the first voltage to the plurality of wordlines when the xth wordline is used to perform the program operation.	[13.A] applying the first voltage to the plurality of wordlines of the third portion.

B. Ground 3: Claims 8 and 9 are Rendered Obvious by the Combination of Dong and Costa

205. In my opinion, Dong in combination with Costa, considered in view of the knowledge of a POSITA, renders claims 8 and 9 obvious.

1. Motivation to Combine Dong and Costa

206. In this section I explain reasons why a POSITA would be motivated to combine Dong and Costa. In my opinion, a POSITA would have been motivated to combine Dong and Costa to obtain their respective benefits for improving the reliability of memory devices.

207. As an initial matter, both Dong and Costa address issues in the field of NAND flash memory. Dong, [0011]; Costa, [0014], [0041]. Moreover, both references address reliability issues specifically related to program operations. Dong, Abstract; Costa, Abstract.

208. As I discussed briefly in the Technology Background section, a POSITA would have known that a program operation is an iterative process with each iteration involving application of a programming pulse followed by the

application of program verification pulses (*i.e.*, for reading the cell being programmed). *See, e.g.*, Dong, Figures 19-20 (flow chart and accompanying illustrative graphic showing alternating program/verification pulses); Costa, Fig. 6 (showing program and verification pulses). A POSITA would therefore understand that only a disturb-free application of a programming pulse followed by a disturb-free application of verification pulses would constitute a disturb free programming operation. In other words, a POSITA would have known that both aspects of programming should be addressed to improve the overall reliability of flash devices during the programming operation. Accordingly, a POSITA looking to improve device reliability would have considered Dong's techniques (focused on program disturb resulting from the programming pulses) alongside other techniques (focused on undesired artifacts from the verification phase) for more comprehensive improvement of memory reliability.

209. In this regard, Dong and Costa focus on two different (but complementary) phenomena that impact memory device reliability during programming. Taking Dong first, Dong teaches reducing the effects of program disturb that are caused by applying the actual programming pulses. *See* Dong, Figure 4, 19-20, [0042]-[0043] ("Program disturb occurs when the threshold voltage of an unselected non-volatile storage element is shifted due to programming of other non-volatile storage elements"). More specifically, Dong's programming

techniques are directed to reducing unintended program disturb on memory cells that result from the high electric field in the channel region when the program pulses are applied. Dong, [0015]. But Dong says nothing about problems that may arise in connection with the errors that may be introduced as a result of the verify phase, such as increased channel resistance that progressively occurs as more memory cells in a memory string are programmed.

210. Costa, on the other hand, is primarily concerned with solving program disturb impacts that result from the verification phase of the programming operation. Costa, [0029]-[0030] (discussing channel resistance impact on programming verification). As a POSITA would have known, Costa teaches that “[t]he channel resistance increases as each memory cell is programmed, leading to the highest resistance and lowest channel conductance when programming the later programmed memory cells.” Costa, [0029]; *see also id.*, [0101]. The increased channel resistance may cause “later programmed memory cells ... to be programmed to higher levels than intended.” *Id.*, [0029]. Costa combats this undesirable consequence affecting memory cell threshold voltage by applying “a variable pass voltage” that is dependent on where the cell being programming falls in the memory string. Costa, Abstract, [0029], [0101]-[0104].

211. Dong says nothing about increased channel resistance or how to address it. That would have motivated a POSITA to investigate techniques that could

address programming disturb or programming errors on the memory cells of a string as a function of the location of the selected wordline that is undergoing the programming operation. The channel resistance problem discussed by Costa is exacerbated as the number of cells in a memory string increases. And certainly by the 2002 Priority Date of the '941 patent memory strings in excess of 100 cells were known and used. This would have further motivated a POSITA to employ Costa's technique with Dong.

212. Even more, Dong acknowledges that program disturb may occur on both erased memory cells and previously programmed cells. Dong [0011]. A POSITA would understand that this means program disturb is an issue that occurs as programming progresses across a memory string. Again, this is further motivation for a POSITA to Costa's variable pass voltage approach to reduce program disturb/errors that occur as function of the location of a cell being programmed.

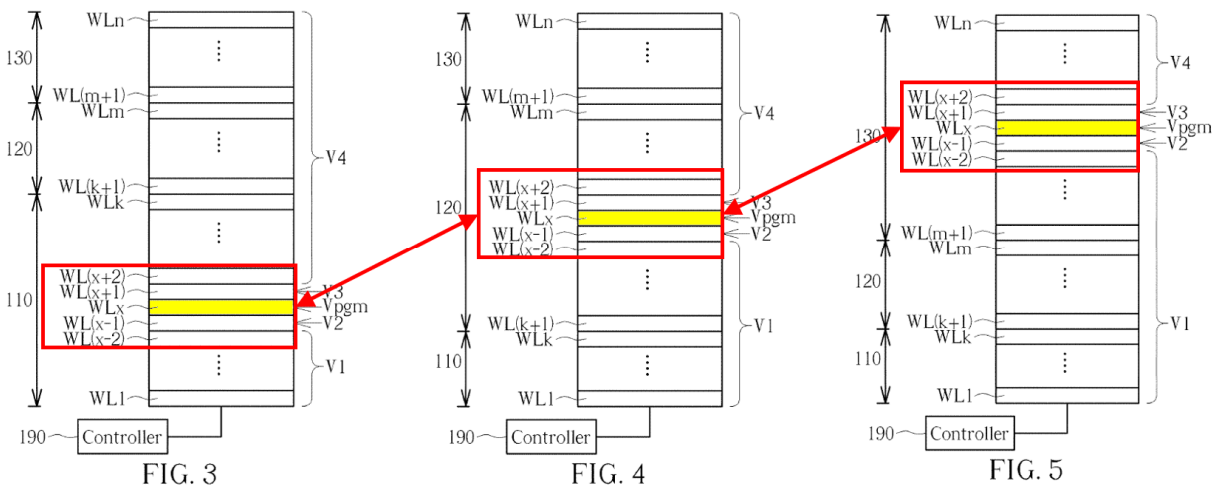
213. A POSITA would have reasonably expected the combined teaching of Dong and Costa to work together because they mutually compatible. In other words, they are complementary to each other in the sense that Dong focuses on the program pulse aspect of programming and Costa focuses on program disturb problems caused by the verify phase of programming. Thus, the combination would address memory cell errors introduced during both phases of the program operation (*viz.*, the programming pulse and verification pulse), without significantly impacting or

changing the benefits achieved by the other reference (or how those benefits are achieved). Stated another way, a POSITA would have had a reasonable expectation of success because both references disclose similar control circuitry, memory arrays, read-operation processes, etc. that would not need to be modified in view of the complementary reference. Indeed, no new or additional hardware would be required; and implementing Costa's teachings would require only minimal software tweak to the algorithms implemented by Dong's controller. Such changes were well within the knowledge and skill of skill of a POSITA and would have involved nothing more nothing more than routine application of known techniques.

2. **Claim 8: "The program method of claim 7, wherein: [A] the first voltage has a first level if the xth wordline is in the first portion; [B] the first voltage has a second level if the xth wordline is in the second portion; [C] the first voltage has a third level if the xth wordline is in the third portion; and [D] the first level is lower than the second level, and the second level is lower than the third level."**

214. Claim 8 is a dependent claim and depends from claim 7. As I previously discussed, Dong satisfies claim 7. In my opinion, Dong combined with Costa renders claim 8 obvious. I have already discussed the reasons for combining Dong with Costa in the preceding section. Although somewhat wordy, the concept described in claim 8 is rather simple. It just requires that the voltage level of the *first voltage* increases as memory programming progresses upward through the memory string from the *first portion*, to the *second portion*, to the *third portion*. In

this regard, it is instructive to look at Figures 3, 4, and 5 in the '941 patent, which shows the aforementioned progression where the wordline used for programming (in yellow) moves from one portion to the next. In Figure 3, the *xth wordline* (WL_x) is in the *first portion*, in Figure 4 the *xth wordline* is in the *second portion*, and in Figure 5 the *xth wordline* is in the *third portion*.



Ex. 1001, Figures 3-5 (annotated).

215. Dong does not say one way or the other what happens to the V_{PASS} voltage depending on the location of the memory cell being programmed. Costa, however, does specifically disclose the V_{pass} voltage applies to wordlines below the cell being programmed increases as programming proceeds upward through the memory string, as I describe in the summary of Costa in §IX.B.

216. Costa explains that memory cells (and their accompanying wordlines) are “partitioned into ranges or zones.” Costa, [0032]; *see also* [0114]-[0116].

Notwithstanding the difference in nomenclature, a POSITA would understand that Costa's so-called "ranges or zones" correspond to the claimed *portions*. Costa explains that the pass voltage applied to "the earlier programmed memory cells"—which would be the cells beneath the cell being programmed when memory string programming proceeds from bottom to top—"varie[s] based on the zone of the selected memory cell." Costa, [0031]-[0032]. Costa further explains that "[g]enerally, a lower pass voltage is applied to these memory cells when the selected wordline is in a lower zone corresponding to earlier programmed memory cells closer to a first select line. *Id.*, [0032].

217. This is one of Costa's central teachings, and it repeatedly explains that "the pass voltage for the memory cells is higher when programming memory cells in higher zones after programming for lower zone memory cells completes. A higher pass voltage is used for the memory cells as more wordlines of the group undergo programming." Costa, [0032]; *see also id.*, Abstract ("As programming progresses ..., the pass voltage for at least a subset of the unselected memory cells that have already been subjected to programming may be increased."), [0030], [0033] (describing "variable pass voltage"), [0104], [0028] ("the pass voltage that is applied is dependent upon the position of the selected memory cell."), [0104] ("the pass voltage applied to a subset of unselected memory cells can be varied based on the selected wordline"), Figures 8-10. Costa even gives specific examples of the voltage

increments that may be used: “the second pass voltage $V_{read+100}$ is higher than the first read pass voltage V_{read} by 100 mV.” *Id.*, [0111].

218. Costa provides further information and teachings about the size and number of zones that may be used. Costa provides what it calls “a simple example,” where “a NAND string [is divided] into two equal-sized zones.” Costa, [0115]. In this example, “[w]hen verifying a selected wordline in the first zone (e.g., WL3), a lower read pass voltage (e.g., $V_{read-100}$ [mV]) is applied to a subset of unselected wordlines on the already programmed (source) side of the selected wordline. When verifying a selected wordline in the second zone (e.g., WL124), a higher read pass voltage (e.g., V_{read}) is applied to the subset of unselected wordlines on the source side of the selected wordline.” *Id.*; *see also id.*, [0114].

219. But Costa is not limited to just that one scenario, and contemplates a more flexible approach. Costa expressly teaches that “the size of the zones may not be equal ... [and] more than two zones may be used. Any number of zones may be used, including a zone for each wordline. When verifying a selected wordline in a zone, the read pass voltage applied to the subset of unselected memory cells is higher than the read pass voltage that was applied when verifying a selected wordline in any of the previously programmed zones.” *Id.*, [0116].

220. Against this backdrop, applying the teachings of Costa’s “wordline dependent pass voltages” (Costa, Title) to Dong satisfies claim 8. More specifically,

I discussed for claims 5 and 7 (on which claim 8 depends) how Dong Figure 5a maps to the three *portions*. Mapping each *portion* in Dong to one of Costa's "zones," the claimed *first voltage* (viz., Dong's V_{PASS} voltage) applied to the previously programmed cells in a memory string increases as the wordline used for programming advances up the memory string from the *first portion* (first zone) to the *second portion* (second zone), and from the *second portion* (second zone) to the *third portion* (third zone).

3. **Claim 9: "The method of claim 5, wherein: [A] the first voltage has a first level if the xth wordline is in the first portion; [B] the first voltage has a second level if the xth wordline is in the second portion; and [C] the first level is lower than the second level."**

221. Claim 9 is broader than claim 8 as shown by the comparison table below and is satisfied by the combination of Dong and Costa for at least the same reasons as claim 8. Claim 9 requires the same underlying concept as claim 8 as it relates to the relative level of the *first voltage*, but claim 9 only employs it in the circumstance when the *xth wordline* being used for programming is in the *first* and *second portions*. Claim 8, by contrast extends the concept to the *third portion* as well.

Claim 8	Claim 9
[8.A] the first voltage has a first level if the xth wordline is in the first portion;	[9.A] the first voltage has a first level if the xth wordline is in the first portion;
[8.B] the first voltage has a second level if the xth wordline is in the second portion;	[9.B] the first voltage has a second level if the xth wordline is in the second portion; and
[8.C] the first voltage has a third level if the xth wordline is in the third portion; and	
[8.D] the first level is lower than the second level, and the second level is lower than the third level.	[9.C] the first level is lower than the second level.

C. Ground 4: Claims 4, 10, and 14 are Rendered Obvious by the Combination of Dong and Shirai

222. In my opinion, Dong in combination with Shirai, considered in view of the knowledge of a POSITA, renders claims 4, 10, and 14 obvious.

1. Motivation to Combine Dong and Shirai

223. In this section I explain reasons why a POSITA would be motivated to combine Shirai's teachings regarding a multiple deck memory structure with Dong. The primary reason relates to well-known semiconductor manufacturing/fabrication challenges imposed by limitations on forming anisotropic vertical etch profiles (*i.e.*, forming a deep etch with perfectly vertical sidewalls) as semiconductor devices become thicker. Memory strings in 3D NAND devices typically require etching a hole vertically through numerous layers of semiconductor material. Given the

number of holes, the etches need to be highly repeatable and controllable, and highly anisotropic. The importance of anisotropic etching only increases as the number of layers (*i.e.*, memory cells) increases. And, as noted in the Technology Background, there was (and is) constant demand for higher density memory which requires more and more layers in a memory string. *See, e.g.*, Ex. 1009, 292-295. The ability to successfully etch anisotropic memory string holes is an important factor in the resulting yield of a 3D memory device.

224. Dong expressly discloses its “array of storage elements can be three dimensional.” Dong, [0095]. And, through Lutze which is incorporated by reference, Dong discloses that its memory strings are “not limited to any particular number of memory cells.” Lutze, 2:11-18. Further, it was well-known by the ’941 patent’s Priority Date in 2020 that 3D NAND strings often exceed 100 cells. *See* §X.A.1 (discussing memory string lengths).

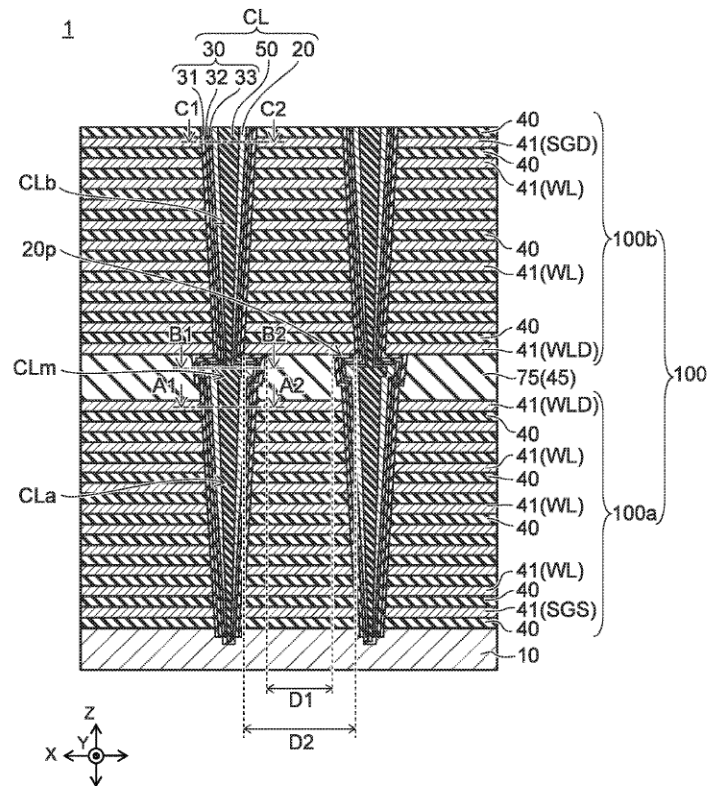
225. Shirai itself confirms the summary above regarding memory string formation and the importance of anisotropic etching. Shirai states that when manufacturing a 3D NAND memory devices, “holes are formed in a stacked body including multiple conductive layers. As the number of stacked layers in the stacked body increases, it becomes difficult to collectively form the holes.” Shirai, [0003]; *see also id.*, [0042]; (“As the number of stacked layers in the stacked body 100 increases, it becomes difficult to form the holes.”). Again, this is because

maintaining an anisotropic etch profile is increasingly difficult as the number of layers (*viz.*, memory cells and wordlines) increases.

226. As a result, Shirai (and numerous other references) teach a piece wise approach employing a multi-deck memory structure where memory holes are etched separately in each deck and then connected to form longer memory strings. Shirai, Figures 2 and 7-15, [0042], [0003] (“A method has been proposed in which holes can be easily formed even in a stacked body having a large number of stacked layers by repeating formation of the holes and formation of the stacked body (stacked portion).”); *see also*, *e.g.*, Ex. 1019 (U.S. Pat. 10,438,671, “Chen”), 4:10-32 (“Due to a desire to increase the height of the stack to allow for more layers and more memory cells in each NAND string, ... [t]he memory hole is etched separately in each tier so that a low aspect ratio can be achieved.”).

227. This alleviates the need for a single anisotropic etch that can extend the full depth of the semiconductor stack, and provides for larger acceptable manufacturing/etching tolerances. Shirai Figure 2 illustrates this. The etching profile of the memory holes is apparent as the width tapers from wider at the top to narrower at the bottom.

FIG. 2



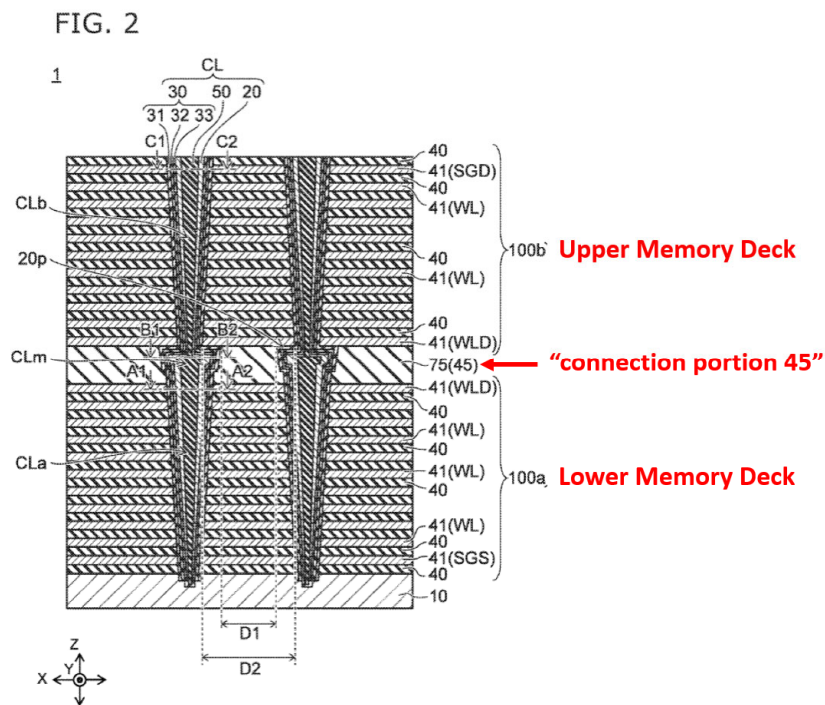
Shirai, Figure 2. As a POSITA would understand, in the example above, trying to form a memory hole using a single etch through the entire structure would result in memory strings that did not extend through all layers and/or were too wide at top (and/or too narrow at the bottom). Even assuming that the etch extended through the entire stack, the tapered profile would have the effect of limiting the number of memory strings that could fit in a given area because each hole would require more “horizontal” space at the top, thus limiting/reducing memory density. In other words, memory strings created from a single etch would typically be inoperable, have poor yield, and result in reliability issues. Accordingly, a POSITA would have

been motivated to employ a multi-deck memory architecture and manufacturing process (like that disclosed in Shirai) with Dong.

228. Because Shirai's two-deck memory architecture (and associated manufacturing process) was both well-known and taught by Shirai itself (*see* Shirai, Figures 5-15), a POSITA would have had a reasonable expectation of success in employing it with Dong as of the Priority Date of the '941 patent. I further note that the use of dummy wordlines at the boundary of the memory decks was taught by Sharai (and many other references) to mitigate artifacts associated with memory cells that often occur near the memory deck interface, further reinforcing the reasonable expectation of success. Examples of the knowledge of a POSITA on this topic further bolstering my opinion regarding the reasonable expectation of success include (1) U.S. Pub. 2012/0003800 (Ex. 1014, "Lee III"), Figures 28, 33, [0005], [0074]; (2) Righetti, [0002], [0017], Figure 1; and (3) Chen, Figures 6 and 16, 4:10-32 ("The memory hole is etched separately in each tier so that a low aspect ratio can be achieved."), 14:64-15:25.

2. **Claim 4:** “The memory of claim 1, further comprising: [A] an upper dummy wordline formed below the first portion; [B] a lower dummy wordline formed above the second portion; and [C] a joint oxide layer formed between the upper dummy wordline and the lower dummy wordline; [D] wherein the memory has a two-deck structure.”

229. In my opinion, the combination of Shirai’s two-deck teachings with Dong satisfies claim 4. I have already described Shirai’s two-deck memory structure in the paragraphs above and in §X.C. In summary, Shirai discloses a 3D NAND memory device that has multiple memory decks, which Shirai refers to as “stacked portions.” *See, e.g.*, Shirai, [0030] (Figure 2 “includes a first stacked portion 100a, a connection portion 45, and a second stacked portion 100b.”); Figures 2 and 7-15. Shirai’s *two-deck structure* is illustrated in Figure 2 below.



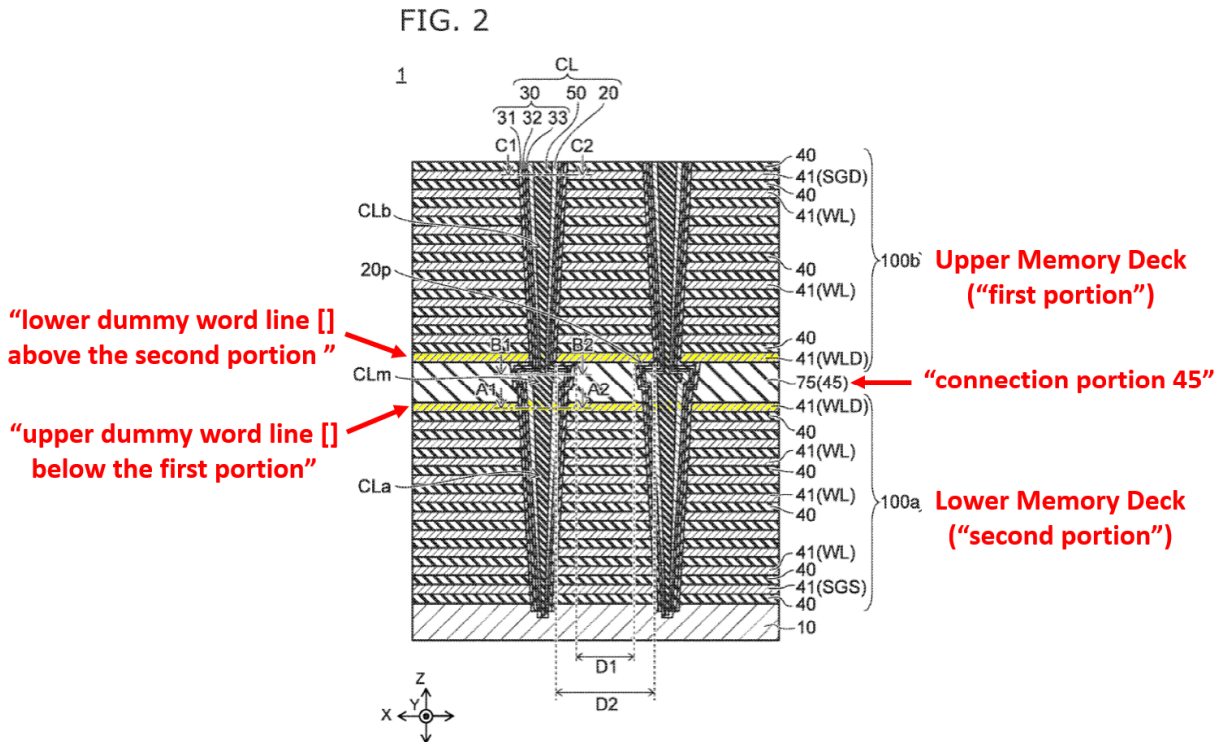
Id., Figure 2 (annotated).

230. After the lower deck 100a has been formed, Shirai discloses that a “connection portion 45 is provided on the first stacked portion 100a,” after which “[t]he second stacked portion 100b is provided on the connection portion 45.” *Id.*, [0030]. The connecting portion 45 may be “silicon oxide.” *Id.*, [0045]; *see also id.*, [0014] (“The connection portion includes an insulating layer between the high dielectric layer and the second stacked portion.”); [0031] (disclosing various oxides for connecting portion 45). Shirai’s connecting portion 45, when formed of, *e.g.*, silicon oxide is the claimed *joint oxide layer*.

231. I next discuss the claimed *dummy wordlines*. The term *dummy wordline* is a well-known term in the art and refers to a wordline which controls a memory cell that is not used for data storage purposes. The reasons for having so-called dummy wordlines (typically near the top or bottom of a memory string) are well-known in the art to POSITA’s. Shirai further discloses that in lower deck 100a, “the electrode layer 41 closest to the connection portion 45 functions as a dummy wordline WLD,” and in upper deck 100b, “the electrode layer 41 closest to the connection portion 45 functions as a dummy wordline WLD.” *Id.*, [0032]. As such, Shirai discloses the *dummy wordlines* recited in the claim 4. As Shirai Figure 2 above shows, the connection portion 45 is *formed between the [dummy wordlines]*.

232. A further requirement of claim 4 is that the *upper dummy wordline [is] formed below the first portion* and the *lower dummy wordline [is] formed above*

the second portion. Shirai satisfies these requirements as well. The locations of the Shirai's dummy wordlines relative to stack portions 100a and 100b (which correspond to the **[memory] decks**) are shown in Figure 2 below.



Shirai, Figure 2 (with dummy wordlines highlighted in yellow). Claim 1 (on which claim 4 depends) says that the **second portion [is] formed below the first portion**. See element [1.B]. Shirai's upper deck 100b corresponds to Dong's first portion as identified in Figure 5g. And Shirai's bottom deck 100a corresponds to Dong's second portion as identified in Figure 5g. Accordingly, Shirai's dummy wordline 41 in lower deck 100a is an **upper dummy wordline[s] formed below the first portion** (e.g., below upper deck 100b). And Shirai's dummy wordline 41 in

upper deck 100b is a *lower dummy wordline formed above the second portion* (e.g., above lower deck 100a).

- 3. Claim 10: “The program method of claim 5, wherein the memory further comprises: [A] a lower dummy wordline formed above the first portion; [B] an upper dummy wordline formed below the second portion; [C] a joint oxide layer formed between the lower dummy wordline and the upper dummy wordline; [D] wherein the memory has a two-deck structure.”**

233. The combination of Dong and Shirai satisfies claim 10. Claim 10 is a dependent claim and depends from claim 5. As with claim 4, claim 10 is divided into 4 elements. As the table below shows, claim element [4.C] is the same as claim element [10.C]. And claim element [4.D] is the same as claim element [10.D].

Claim 4	Claim 10
[4.A] an upper dummy wordline formed below the first portion;	[10.A] a lower dummy wordline formed above the first portion;
[4.B] a lower dummy wordline formed above the second portion; and	[10.B] an upper dummy wordline formed below the second portion;
[4.C] a joint oxide layer formed between the upper dummy wordline and the lower dummy wordline;	[10.C] a joint oxide layer formed between the lower dummy wordline and the upper dummy wordline;
[4.D] wherein the memory has a two-deck structure.	[10.D] wherein the memory has a two-deck structure.

234. Claim elements [4.A] and [4.B] are substantively the same as claim elements [10.B] and [4.B], respectively, even though they are worded slightly differently. Claim elements [4.A] and [10.B] both recite *an upper dummy wordline*.

And claim elements [4.B] and [10.A] both require *a lower dummy wordline*. The differences are that (1) claim element [4.A] says the *upper dummy wordline* is *formed below the first portion*,” whereas element [10.B] says it is *formed below the second portion*; and (2) claim element [4.B] says the *lower dummy wordline* is *formed above the second portion*,” whereas [10.A] says it is *formed above the first portion*. The reason these limitations amount to substantively the same thing is because their respective independent claims define the *first* and *second* portions differently. In particular, claim element [1.B] says *a second portion [is] formed below the first portion*, but the preamble of claim 5 says *the second portion [is] formed above the first portion*.

- 4. **Claim 14:** “[PRE] The program method of claim 11, wherein the memory further comprises: [A] an upper dummy wordline formed below the first portion; [B] a lower dummy wordline formed above the second portion; and [C] a joint oxide layer formed between the upper dummy wordline and the lower dummy wordline; [D] wherein the memory has a two-deck structure.”

235. The additional limitations of claim 14 and the same as claim 4 as shown in the table below. Accordingly, Dong combined with Shirai satisfies claim 14 for the same reasons as claim 4.

Claim 4	Claim 14
[4.A] an upper dummy wordline formed below the first portion;	[14.A] an upper dummy wordline formed below the first portion;

Claim 4	Claim 14
[4.B] a lower dummy wordline formed above the second portion; and	[14.B] a lower dummy wordline formed above the second portion; and
[4.C] a joint oxide layer formed between the upper dummy wordline and the lower dummy wordline;	[14.C] a joint oxide layer formed between the upper dummy wordline and the lower dummy wordline;
[4.D] wherein the memory has a two-deck structure.	[14.D] wherein the memory has a two-deck structure.

XI. SECONDARY CONSIDERATIONS

236. Based on my review of the '941 patent's prosecution history, I note that no secondary considerations of non-obviousness were identified in connection with the patent's claims.

237. I am not separately aware of any evidence of commercial success, skepticism, failure of others, industry praise, unexpected results, or the like. Instead, as discussed above, it is my opinion that the subject matter embraced by the '941 patent's claims was known in and obvious over the prior art. Moreover, it is also my opinion the teachings of the prior art would have been expected to successfully function as or be usable with working NAND memory devices.

XII. CONCLUSION

238. I declare 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 willfully false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1101 of Title 18 of the United States Code.

Date: November 26, 2024

Location: Fremont, California

Signature: 
David Kuan-Yu Liu