

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE PATENT TRIAL AND APPEAL BOARD

---

INTEL CORPORATION,  
Petitioner,

v.

MASAHIRO IIDA,  
Patent Owner.

---

Case No. IPR2023-00864  
U.S. Patent No. 6,812,737

---

**DECLARATION OF STEPHEN W. MELVIN, PH.D.**

|                                 |
|---------------------------------|
| Intel Ex. 1003<br>Intel v. Iida |
|---------------------------------|

## **TABLE OF CONTENTS**

|       |                                                                         |    |
|-------|-------------------------------------------------------------------------|----|
| I.    | PERSONAL AND PROFESSIONAL BACKGROUND .....                              | 1  |
| A.    | Education.....                                                          | 2  |
| B.    | Career Synopsis.....                                                    | 2  |
| II.   | MATERIALS REVIEWED AND CONSIDERED.....                                  | 4  |
| III.  | MY UNDERSTANDING OF PATENT LAW .....                                    | 8  |
| A.    | Anticipation.....                                                       | 10 |
| B.    | Obviousness.....                                                        | 10 |
| IV.   | PERSON OF ORDINARY SKILL IN THE ART .....                               | 13 |
| V.    | BACKGROUND OF PROGRAMMABLE LOGIC DEVICES AND LOOK-<br>UP TABLES .....   | 16 |
| A.    | Digital Circuits .....                                                  | 16 |
| B.    | Look-Up Tables.....                                                     | 19 |
| C.    | Overview of Programmable Logic Devices .....                            | 26 |
| VI.   | THE '737 PATENT .....                                                   | 31 |
| A.    | Preferred Embodiments.....                                              | 31 |
| B.    | Prosecution History .....                                               | 36 |
| C.    | The Challenged Claims .....                                             | 37 |
| VII.  | CLAIM INTERPRETATION.....                                               | 39 |
| A.    | “look up table of M inputs and N outputs” .....                         | 41 |
| B.    | “LUT units” .....                                                       | 43 |
| VIII. | GROUND 1.....                                                           | 44 |
| A.    | Overview of Fujii and Betz .....                                        | 44 |
| 1.    | Fujii (Ex.1005).....                                                    | 44 |
| 2.    | Betz (Ex.1006).....                                                     | 46 |
| B.    | Independent Claim 1 .....                                               | 47 |
| 1.    | [1PRE] “A look up table of M inputs and N outputs,<br>comprising” ..... | 47 |
| 2.    | [1A] “a plurality of LUT units; and” .....                              | 52 |

|    |                                                                                                                                                                                                                                                                                                |    |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 3. | [1B] “an internal configuration control circuit controlling an internal configuration of said plurality of LUT units,”.....                                                                                                                                                                    | 60 |
| 4. | [1C] “wherein said internal configuration control circuit comprises a plurality of selectors selecting I/O signals of said plurality of LUT units, and”.....                                                                                                                                   | 69 |
| 5. | [1D] [wherein said internal configuration control circuit comprises . . . ] “a selector control circuit having a memory, controlling said plurality of selectors in accordance with data stored in said memory, and defining the internal configuration of said plurality of LUT units.” ..... | 73 |
| C. | Claim 2 .....                                                                                                                                                                                                                                                                                  | 81 |
| 1. | [2A] “The look up table as claimed in claim 1, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and” .....                                                                              | 81 |
| 2. | [2B] [wherein said plurality of selectors include: . . . ] “an output signal selector provided at an output of said LUT units selecting an output signal,”.....                                                                                                                                | 82 |
| 3. | [2C] “said input signal selector and said output signal selector being controlled in accordance with the data stored in said memory.”.....                                                                                                                                                     | 83 |
| D. | Claim 3 .....                                                                                                                                                                                                                                                                                  | 84 |
| 1. | [3A] “The look up table as claimed in claim 1, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and” .....                                                                              | 84 |
| 2. | [3B] [wherein said plurality of selectors include: . . . ] “an output signal selector provided at an output of said LUT units selecting an output signal,”.....                                                                                                                                | 84 |
| 3. | [3C] “said input signal selector and said output signal selector being controlled in accordance with the input signal.”.....                                                                                                                                                                   | 85 |
| E. | Claim 4: “The look up table as claimed in claim 1, wherein said look up table of M inputs and N outputs is a 6-input 3-output look up table.”.....                                                                                                                                             | 88 |

|     |                                                                                                                                                                                                                                       |     |
|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| F.  | Claim 5: “The look up table as claimed in claim 4, wherein said 6-input 3-output look up table comprises eight 3-input 1-output LUT units.” .....                                                                                     | 92  |
| G.  | Independent Claim 7 .....                                                                                                                                                                                                             | 95  |
| 1.  | Motivation to Combine the Fujii-Betz PL1 with Betz’s FPGA Architecture.....                                                                                                                                                           | 95  |
| 2.  | [7PRE] “A programmable logic circuit device comprising:”...98                                                                                                                                                                         |     |
| 3.  | [7A] “a plurality of logic blocks;” .....                                                                                                                                                                                             | 99  |
| 4.  | [7B] “a plurality of routing wires connected to each of said logic blocks;” .....                                                                                                                                                     | 101 |
| 5.  | [7C] “a plurality of switch circuits provided at an intersection of each of said routing wires;” .....                                                                                                                                | 102 |
| 6.  | [7D] “a plurality of connection blocks provided between an I/O line of each of said logic blocks and each of said routing wires; and” .....                                                                                           | 103 |
| 7.  | [7E] “an I/O block performing an input/output operation with external equipment,” .....                                                                                                                                               | 104 |
| 8.  | [7F] “wherein each of said logic blocks has a look up table of M inputs and N outputs, comprising:” .....                                                                                                                             | 106 |
| 9.  | [7G] “a plurality of LUT units; and” .....                                                                                                                                                                                            | 108 |
| 10. | [7H] “an internal configuration control circuit controlling an internal configuration of said plurality of LUT units,” .....                                                                                                          | 108 |
| 11. | [7I] “wherein said internal configuration control circuit comprises a plurality of selectors selecting I/O signals of said plurality of LUT units, and” .....                                                                         | 108 |
| 12. | [7J] “a selector control circuit having a memory, controlling said plurality of selectors in accordance with data stored in said memory, and defining the internal configuration of said plurality of LUT units.” .....               | 108 |
| H.  | Claim 8 .....                                                                                                                                                                                                                         | 108 |
| 1.  | [8A] “The programmable logic circuit device as claimed in claim 7, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and” ..... | 108 |

|    |                                                                                                                                                                                                                                                                                                   |     |
|----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| 2. | [8B] “an output signal selector provided at an output of said LUT units selecting an output signal,” .....                                                                                                                                                                                        | 109 |
| 3. | [8C] “said input signal selector and said output signal selector being controlled in accordance with the data stored in said memory.” .....                                                                                                                                                       | 109 |
| I. | Claim 9 .....                                                                                                                                                                                                                                                                                     | 109 |
| 1. | [9A] “The programmable logic circuit device as claimed in claim 7, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and” .....                                                             | 109 |
| 2. | [9B] “an output signal selector provided at an output of said LUT units selecting an output signal,” .....                                                                                                                                                                                        | 109 |
| 3. | [9C] “said input signal selector and said output signal selector being controlled in accordance with the input signal.” .....                                                                                                                                                                     | 109 |
| J. | Claim 10: “The programmable logic circuit device as claimed in claim 7, wherein said look up table of M inputs and N outputs is a 6-input 3-output look up table.” .....                                                                                                                          | 110 |
| K. | Claim 11: “The programmable logic circuit device as claimed in claim 10, wherein said 6-input 3-output look up table comprises eight 3-input 1-output LUT units.” .....                                                                                                                           | 110 |
| L. | Independent Claim 13 .....                                                                                                                                                                                                                                                                        | 110 |
| 1. | [13PRE] “A method of configuring a look up table of M inputs and N outputs, comprising:” .....                                                                                                                                                                                                    | 110 |
| 2. | [13A] “providing a plurality of LUT units, and” .....                                                                                                                                                                                                                                             | 111 |
| 3. | [13B] “selectively controlling I/O signals of said plurality of LUT units to set a predetermined mode of an internal configuration.” .....                                                                                                                                                        | 111 |
| M. | Claim 14: “The method of configuring a look up table as claimed in claim 13, wherein the I/O signals of said plurality of LUT units are selectively controlled in accordance with data stored in the corresponding look up table.” .....                                                          | 111 |
| N. | Claim 15: “The method of configuring a look up table as claimed in claim 13, wherein an input signal input to at least one of said LUT units and an output signal output from said LUT units are selectively controlled in accordance with data stored in the corresponding look up table.” ..... | 112 |

|      |                                                                                                                                                                                                                                                                                                 |     |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| O.   | Claim 16: “The method of configuring a look up table as claimed in claim 13, wherein an input signal input to at least one of said LUT units and an output signal output from said LUT units are selectively controlled in accordance with a predetermined function of the input signal.” ..... | 114 |
| IX.  | GROUND 2.....                                                                                                                                                                                                                                                                                   | 115 |
| A.   | Claim 6: “The look up table as claimed in claim 4, wherein said 6-input 3-output look up table comprises four 3-input 2-output LUT units.” .....                                                                                                                                                | 115 |
| B.   | Claim 12: “The programmable logic circuit device as claimed in claim 10, wherein said 6-input 3-output look up table comprises four 3-input 2-output LUT units.” .....                                                                                                                          | 119 |
| X.   | GROUND 3 AND GROUND 4.....                                                                                                                                                                                                                                                                      | 120 |
| XI.  | CONCLUSION.....                                                                                                                                                                                                                                                                                 | 121 |
| XII. | APPENDIX: CLAIM LISTING .....                                                                                                                                                                                                                                                                   | 123 |

I, Stephen W. Melvin, do hereby declare as follows:

1. I have been retained by Munger, Tolles & Olson LLP as counsel for Petitioner Intel Corporation to assess claims 1-16 (the “challenged claims”) of U.S. Patent No. 6,812,737 (“the ’737 patent”). My employer, Exponent, Inc., is being compensated for my time at my standard rate of \$650 per hour, plus actual expenses. My compensation is not dependent in any way upon any outcome in the above-captioned proceeding, or any other proceeding, concerning the ’737 patent.

**I. PERSONAL AND PROFESSIONAL BACKGROUND**

2. My qualifications for forming the opinions set forth in this declaration are summarized below and explained in more detail in my current curriculum vitae, provided as Exhibit 1004, which also includes a list of my publications and patents.

3. During my career I have designed and developed Very Large Scale Integration (“VLSI”) technology, which has included the use of field programmable gate arrays (“FPGAs”) for prototyping, and I have developed and deployed embedded systems, which has included the use of programmable logic circuit devices or programmable logic devices (“PLDs”). As such, I have acquired expertise and am an expert in the area of PLDs, including those having look-up table architectures and methods.

## **A. Education**

4. In 1982, I obtained a bachelor's degree in Electrical Engineering and Computer Science from the University of California, Berkeley. In 1991, I obtained a Doctor of Philosophy (Ph.D.) degree in Computer Science, also from the University of California, Berkeley. My dissertation, titled "Performance Enhancement Through Dynamic Scheduling and Large Execution Atomic Units in Single Instruction Stream Processors," related to a high-performance processor design that exploited fine-grained parallelism in general-purpose programs.

## **B. Career Synopsis**

5. After obtaining my Ph.D., I led an architecture team designing a network processor at Clearwater Networks (formerly XStream Logic, Inc.) from 2000-2001. I also co-founded a network processor startup called Flowstorm, Inc., (later Consentry Networks) in 2001. Both of those professional experiences required an understanding of PLDs, as they involved studying the use of PLDs during prototyping phases.

6. From 2001-2002, I was a Visiting Scholar at the University of Texas at Austin, where I conducted research on high performance microprocessor design. I also served as General Chair of the 45th Annual Symposium in Microarchitecture in Vancouver, British Columbia, in December 2012.

7. My practical experience regarding PLDs includes designing and building numerous real-time embedded systems that have been commercially deployed, including machine vision systems used in the automotive industry and voice processing systems.

8. I am a named inventor of over 40 United States patents in the computer science field, including patents related to integrated circuits, and I am a registered patent agent before the United States Patent and Trademark Office.

9. In addition, on several occasions, I have served as an expert witness in matters where PLD (including FPGA) design, architecture, and technology analysis were required to render an opinion. These matters include *Mentor Graphics Corp. v. EVE-USA, Inc.*, Nos. 3:10-cv-00954, 3:12-01500, 3:13-cv-00579 (D. Or.), *Pact XPP Technologies, AG v. Xilinx, Inc.*, No. 2:07-cv-563 (E.D. Tex.), and others.

10. I am currently employed as a Principal with Exponent, Inc., which is an international multidisciplinary engineering and scientific consulting firm.

11. I am a member of The Institute of Electrical and Electronics Engineers (IEEE) and the Association for Computing Machinery (ACM), the two largest professional organizations in the field of computer science and electrical engineering.

12. In summary, I have extensive familiarity with the field of PLDs and FPGAs, including those with look-up tables. As I have been studying and working in this field since 1982, I am familiar with what the state of this field was at the effective filing date of the '737 patent and before.

## II. MATERIALS REVIEWED AND CONSIDERED

13. My findings, as explained below, are based on my years of education, research, experience, and background in the field of PLDs, as well as my investigation and study of relevant materials for this declaration. When developing the opinions set forth in this declaration, I have assumed the perspective of a person having ordinary skill in the art ("POSA"), as set forth in Section IV below. In forming my opinions, I have studied and considered the materials identified in the list below.

| <b>Exhibit No.</b> | <b>Description</b>                                                                                                                 |
|--------------------|------------------------------------------------------------------------------------------------------------------------------------|
| 1001               | U.S. Patent No. 6,812,737                                                                                                          |
| 1002               | Prosecution History of U.S. Patent No. 6,812,737                                                                                   |
| 1005               | Japanese Unexamined Patent Application Publication No. H09-121153A ("Fujii"), certified translation, and declaration of translator |
| 1006               | Vaughn Betz et al., <i>Architecture and CAD for Deep-Submicron FPGAs</i> (Kluwer Academic Publishers 1999) ("Betz")                |
| 1007               | U.S. Patent No. 6,097,212 ("Agrawal")                                                                                              |
| 1008               | U.S. Patent No. 5,926,036 ("Cliff")                                                                                                |
| 1009               | Declaration of Agnes O. Villero Regarding Betz, Brown-FPGA, Trimberger, and Bhat                                                   |

| <b>Exhibit No.</b> | <b>Description</b>                                                                                                                                                                                                                                                                |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1010               | Prosecution History of Japanese Patent Application No. 2001-199644 (which issued as Japanese Patent No. 3580785B2), certified translation, and declaration of translator                                                                                                          |
| 1011               | Stephen D. Brown et al., <i>Field-Programmable Gate Arrays</i> (Kluwer Academic Publishers 1992) (“Brown-FPGA”)                                                                                                                                                                   |
| 1012               | Stephen M. Trimberger et al., <i>Field-Programmable Gate Array Technology</i> (Springer Science+Business Media New York 1994) (“Trimberger”)                                                                                                                                      |
| 1013               | Stephen Brown & Jonathan Rose, “FPGA and CPLD Architectures: A Tutorial,” <i>IEEE Design and Test of Computers</i> , vol. 13, no. 2, 42-57 (Dec. 1996) (“Brown-Rose”) (from pages 4-19 of Declaration of Gordon MacPherson at Ex.1014)                                            |
| 1014               | Declaration of Gordon MacPherson regarding Stephen Brown & Jonathan Rose, “FPGA and CPLD Architectures: A Tutorial,” <i>IEEE Design and Test of Computers</i> , vol. 13, no. 2, 42-57 (Dec. 1996)                                                                                 |
| 1015               | Jianshe He & Jonathan Rose, “Advantages of Heterogeneous Logic Block Architectures for FPGAs,” <i>Proceedings of IEEE Custom Integrated Circuits Conference - CICC '93</i> (1993) (“He”) (from pages 4-8 of Declaration of Gordon MacPherson at Ex.1016)                          |
| 1016               | Declaration of Gordon MacPherson regarding Jianshe He & Jonathan Rose, “Advantages of Heterogeneous Logic Block Architectures for FPGAs,” <i>Proceedings of IEEE Custom Integrated Circuits Conference - CICC '93</i> (1993) (from pages 4-8 of Declaration of Gordon MacPherson) |
| 1017               | Jonathan Rose et al., “Architecture of Field-Programmable Gate Arrays,” <i>Proceedings of the IEEE</i> , vol. 81, no. 7, 1013-29 (July 1993) (“Rose”) (from pages 4-20 of Declaration of Gordon MacPherson at Ex.1018)                                                            |
| 1018               | Declaration of Gordon MacPherson regarding Jonathan Rose et al., “Architecture of Field-Programmable Gate Arrays,” <i>Proceedings of the IEEE</i> , vol. 81, no. 7, 1013-29 (July 1993)                                                                                           |
| 1019               | Satwant Singh et al., <i>The Effect of Logic Block Architecture on FPGA Performance</i> , <i>IEEE Journal of Solid-State Circuits</i> , vol. 27, no. 3, 281-                                                                                                                      |

| Exhibit No. | Description                                                                                                                                                                                                                                                                                                    |
|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | 87 (Mar. 1992) (“Singh”) (from pages 4-10 of Declaration of Gordon MacPherson at Ex.1020)                                                                                                                                                                                                                      |
| 1020        | Declaration of Gordon MacPherson regarding Satwant Singh et al., <i>The Effect of Logic Block Architecture on FPGA Performance</i> , IEEE Journal of Solid-State Circuits, vol. 27, no. 3, 281-87 (Mar. 1992)                                                                                                  |
| 1021        | Narasimha B. Bhat, “Novel Techniques for High Performance Field Programmable Logic Devices,” Technical Report no. UCB/ERL M93/80, Electronic Research Laboratory, University of California, Berkeley (Nov. 1993) (“Bhat”)                                                                                      |
| 1022        | Donald W. Cherepacha, Master’s Thesis, <i>A Field-Programmable Gate Array Architecture Optimized for Datapaths</i> , Department of Electrical and Computer Engineering, University of Toronto (1994) (“Cherepacha”)                                                                                            |
| 1023        | Paul Chow et al., “The Design of a SRAM-Based Field-Programmable Gate Array—Part II: Circuit Design and layout,” in <i>IEEE Transactions on Very Large Scale Integration (VLSI) Systems</i> , vol. 7, no. 3, pp. 321-330, Sept. 1999 (“Chow”) (from pages 4-13 of Declaration of Gordon MacPherson at Ex.1024) |
| 1024        | Declaration of Gordon MacPherson regarding Paul Chow et al., “The Design of a SRAM-Based Field-Programmable Gate Array—Part II: Circuit Design and layout,” in <i>IEEE Transactions on Very Large Scale Integration (VLSI) Systems</i> , vol. 7, no. 3, pp. 321-330, Sept. 1999                                |
| 1025        | U.S. Patent No. 5,905,385 (“Sharpe-Geisler”)                                                                                                                                                                                                                                                                   |
| 1026        | Excerpt of Tom M. Apostol, <i>Calculus, Volume I</i> (Wiley 1967) (“Apostol”)                                                                                                                                                                                                                                  |
| 1027        | <i>Principles and Structures of FPGAs</i> (Hideharu Amano, ed., Springer 2018) (“Amano”)                                                                                                                                                                                                                       |
| 1028        | Jason Cong et al., “Technology Mapping For $k/m$ -Macrocell Based FPGAs,” in <i>Proceedings of the 2000 ACM/SIGDA 8th Int’l Symposium on Field Programmable Gate Arrays</i> , 51-59 (Feb. 2000) (“Cong”)                                                                                                       |
| 1029        | Plaintiff’s Preliminary Infringement Contentions in <i>Iida v. Intel Corp.</i> , No. 6:22-cv-00662-ADA (W.D. Tex.)                                                                                                                                                                                             |

| <b>Exhibit No.</b> | <b>Description</b>                                                                                                                                                                                                                                                                                       |
|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1030               | Complaint in <i>Masahiro Iida v. Intel Corporation and Intel K.K.</i> , filed July 4, 2022, in the Department of Intellectual Property Rights, Tokyo District Court, certified translation, and declaration of translator                                                                                |
| 1031               | Intel’s Opening Claim Construction Brief in <i>Iida v. Intel Corp.</i> , No. 6:22-cv-00662-ADA (W.D. Tex.)                                                                                                                                                                                               |
| 1032               | Iida’s Responsive Claim Construction Brief in <i>Iida v. Intel Corp.</i> , No. 6:22-cv-00662-ADA (W.D. Tex.)                                                                                                                                                                                             |
| 1033               | Declaration of Dr. Marwan Hassoun in support of Professor Masahiro Iida’s Responsive Claim Construction Brief in <i>Iida v. Intel Corp.</i> , No. 6:22-cv-00662-ADA (W.D. Tex.)                                                                                                                          |
| 1034               | Intel’s Reply Claim Construction Brief in <i>Iida v. Intel Corp.</i> , No. 6:22-cv-00662-ADA (W.D. Tex.)                                                                                                                                                                                                 |
| 1035               | Iida’s Sur-Reply Claim Construction Brief in <i>Iida v. Intel Corp.</i> , No. 6:22-cv-00662-ADA (W.D. Tex.)                                                                                                                                                                                              |
| 1036               | Joint Claim Construction Statement in <i>Iida v. Intel Corp.</i> , No. 6:22-cv-00662-ADA (W.D. Tex.)                                                                                                                                                                                                     |
| 1040               | Japanese Patent No. 3580785B2, certified translation, and declaration of translator                                                                                                                                                                                                                      |
| 1041               | Iida’s Brief filed March 20, 2023, in <i>Masahiro Iida v. Intel Corporation and Intel K.K.</i> , filed July 4, 2022, in the Department of Intellectual Property Rights, Tokyo District Court, certified translation, and declaration of translator, certified translation, and declaration of translator |
| 1043               | Andrew S. Tanenbaum, <i>Structured Computer Organization</i> (2d ed. 1984) (“Tanenbaum”)                                                                                                                                                                                                                 |
| 1044               | Neil H.E. Weste & Kamran Eshraghian, <i>Principles of CMOS VLSI Design: A Systems Perspective</i> (1985) (“Weste”)                                                                                                                                                                                       |

14. While I have organized the information presented in this declaration into sections and/or divisions, my opinions are supported by the information in the declaration in its entirety.

15. Unless otherwise noted, all emphases in any quoted material has been added.

### **III. MY UNDERSTANDING OF PATENT LAW**

16. I am not an attorney. In developing my opinions, I discussed various relevant legal principles with Petitioner's attorneys. My understanding in this respect is as follows.

17. I understand that "*inter partes* review" ("IPR") is a proceeding before the United States Patent and Trademark Office ("PTO") for evaluating the patentability of an issued patent's claims based on prior art patents and printed publications.

18. I understand that, in this proceeding, Petitioner has the burden of proving that the challenged claims of the '737 patent are unpatentable by a preponderance of the evidence. I understand that "preponderance of the evidence" means that a fact or conclusion is more likely true than not true.

19. I understand that, in IPR proceedings, claim terms in a patent are given their plain-and-ordinary meaning as understood by a POSA in the context of the entire patent and the prosecution history pertaining to the patent. If the

specification and/or prosecution history provides a special definition for a claim term that differs from the meaning the term would otherwise possess, that special definition controls. I have applied these standards in preparing the opinions in this declaration.

20. I understand that determining whether a particular patent or printed publication constitutes prior art to a challenged patent claim can require determining the effective filing date (also known as the priority date) to which the challenged claim is entitled. I understand that for a patent claim to be entitled to the benefit of the filing date of an earlier application to which the patent claims priority, the earlier application must have described the claimed invention in sufficient detail to convey with reasonable clarity to a POSA that the inventor had possession of the claimed invention as of the earlier application's filing date. I understand that a disclosure that merely renders the claimed invention obvious is not sufficient written description for the claim to be entitled to the benefit of the filing date of the application containing that disclosure.

21. I understand that for an invention claimed in a patent to be patentable, it must be, among other things, novel (i.e., not anticipated) and not obvious from the prior art. My understanding of these two legal standards is set forth below.

**A. Anticipation**

22. I understand that, for a patent claim to be “anticipated” by the prior art (and therefore not novel), each and every limitation of the claim must be found, expressly or inherently, in a single prior art reference. I understand that a claim limitation is disclosed for the purpose of anticipation if a POSA would have understood the reference to disclose the limitation based on inferences that a POSA would reasonably be expected to draw from the explicit teachings in the reference when read in light of the POSA’s knowledge and experience.

23. I understand that a claim limitation is inherent in a prior art reference if that limitation is necessarily present when practicing the teachings of the reference, regardless of whether a person of ordinary skill recognized the presence of that limitation in the prior art.

**B. Obviousness**

24. I understand that a patent claim may be unpatentable if it would have been obvious in view of a single prior art reference or a combination of prior art references.

25. I understand that a patent claim is obvious if the differences between the subject matter of the claim and the prior art are such that the subject matter as a whole would have been obvious to a person of ordinary skill in the relevant field at

the time the invention was made. Specifically, I understand that the obviousness question involves a consideration of:

- the scope and content of the prior art;
- the differences between the prior art and the claims at issue;
- the knowledge of a person of ordinary skill in the pertinent art; and
- if present, objective factors indicative of non-obviousness, sometimes referred to as “secondary considerations.” To my knowledge, the Patent Owner has not asserted any such secondary considerations with respect to the ’737 patent.

26. I understand that in order for a claimed invention to be considered obvious, a POSA must have had a reason for combining teachings from multiple prior art references (or for modifying a single prior art reference or implementing its teachings in a given manner, in the case of obviousness in view of a single reference) in the fashion proposed.

27. I further understand that in determining whether a prior art reference would have been combined with other prior art or with other information within the knowledge of a POSA, the following are examples of approaches and rationales that may be considered:

- combining prior art elements according to known methods to yield predictable results;

- simple substitution of one known element for another to obtain predictable results;
- use of a known technique to improve similar devices in the same way;
- applying a known technique to a known device ready for improvement to yield predictable results;
- applying a technique or approach that would have been “obvious to try,” i.e., choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success;
- known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations would have been predictable to one of ordinary skill in the art; and
- some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference, to implement it in a given way, or to combine prior art reference teachings to arrive at the claimed invention. I understand that this teaching, suggestion or motivation may come from a prior art reference or from the knowledge or common sense of one of ordinary skill in the art.

28. I understand that for a single reference or a combination of references to render the claimed invention obvious, a POSA must have been able to arrive at

the claimed invention by modifying, implementing, and/or combining the teachings of the applied references.

#### **IV. PERSON OF ORDINARY SKILL IN THE ART**

29. I have been informed and understand that for purposes of assessing whether prior art references disclose every element of a patent claim (thus “anticipating” the claim) and/or would have rendered the claim obvious, the patent and the prior art references must be assessed from the perspective of a POSA, based on the understanding of that person at the time of the patent claim’s priority date. I have been informed and understand that a POSA is presumed to be aware of all pertinent prior art and the conventional wisdom in the art and is a person of ordinary creativity. I have applied this standard throughout my declaration.

30. The ’737 patent involves technology in the field of PLDs, specifically the use of look-up tables with internally configurable “LUT units” to implement logic functions in PLDs. I have been asked to provide my opinions as to the state of the art in these areas by June 2001. I use this timeframe because the face of the ’737 patent indicates an earliest effective filing date of June 29, 2001. I also understand that the Patent Owner has identified June 29, 2001, as his alleged priority date for the ’737 patent in patent infringement litigation against Petitioner. *See Ex.1029, 5.* Whenever I offer an opinion in this declaration about the knowledge of a POSA, the manner in which a POSA would have understood the

claims of the '737 patent or its description, the manner in which a POSA would have understood the prior art, or what a POSA would have been led to do based on the prior art, I am referencing the 2001 timeframe, even if I do not say so specifically in each case. Demonstrating that the '737 patent's claims were anticipated or would have been obvious in 2001 demonstrates that they remained so any time after 2001, because a POSA as of a later date would possess the same or more knowledge.

31. I understand that the Patent Owner might attempt to prove that the alleged invention recited in the challenged claims was conceived at some time prior to the earliest effective filing date on the face of the patent. At the time of this declaration, I am unaware of any evidence produced by the Patent Owner to establish any earlier conception date.

32. I understand there are multiple factors relevant to determining the level of ordinary skill in the pertinent art, including (1) the levels of education and experience of persons working in the field at the time of the invention, (2) the sophistication of the technology, (3) the types of problems encountered in the field, and (4) the prior art solutions to those problems.

33. In my opinion, a POSA would have had a bachelor's degree in electrical engineering, computer engineering, or an equivalent degree, and one year of professional experience relating to programmable logic devices or field

programmable gate arrays. More education could substitute for experience, and vice versa. This person would have been capable of understanding and applying the teachings of the '737 patent and would have had a working knowledge of programmable logic devices pertinent to the '737 patent, including prior art describing look-up table designs. This definition is consistent with the definition Patent Holder has offered in *Iida v. Intel Corp.*, No. 6:22-cv-00662-ADA (W.D. Tex.) (the "District Court Case"). Ex.1033, ¶32

34. By 2001, I held a Ph.D. in Computer Science and was employed as the Senior Architect for Clearwater Networks (formerly XStream Logic, Inc.). I also served as a Visiting Scholar at the University of Texas, Austin, where I directed graduate students in research into high performance computer architecture. In May 2001, I co-founded FlowStorm, Inc. (later ConSentry Networks), where I was responsible for defining and guiding the overall chip architecture for a new packet processor. Prior to that I received a bachelor's degree in Electrical Engineering and Computer Science. By 2001, I had more than 18 years of academic and industry experience in computer science, including the experience listed in Section I above. Therefore, I was a person of more than ordinary skill in the art during the relevant timeframe. During my career and specifically in the 2001 timeframe, I worked with many people who fit the characteristics of a POSA, and I am familiar with their level of skill. When

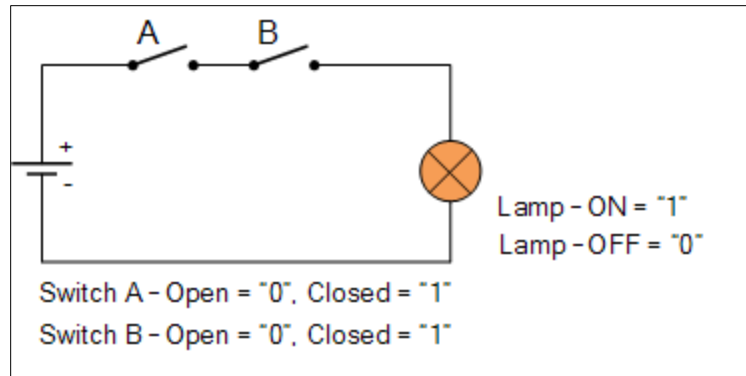
developing the opinions set forth in this declaration, I assumed the perspective of a POSA, as set forth above.

## **V. BACKGROUND OF PROGRAMMABLE LOGIC DEVICES AND LOOK-UP TABLES**

35. Before addressing the Grounds, I will provide a brief overview of the relevant technology.

### **A. Digital Circuits**

36. To understand look-up tables and the role they play in PLDs, it is helpful to start with a few basic principles of digital circuits. Digital circuits are the foundation of modern computing. Unlike analog circuits, which process signals that vary continuously over a voltage range, digital circuits are designed to process signals that are normally present at discrete voltages. The majority of digital circuits handle two different voltages. One of these voltages could be labeled “HIGH,” “on,” “1,” or “TRUE,” and the other voltage could be labeled “LOW,” “off,” “0,” or “FALSE.” In the simple example below, two switches are illustrated where the state of the switch as being either open or closed can be represented by binary inputs A and B, and the state of the Lamp as being on or off can be represented by a binary output (whether a lamp is on or off):



When A and B are HIGH (1), the two switches are closed, current flows through the circuit, and the lamp turns on. Conversely, if either A or B, or both are LOW (0), current does not flow through the circuit and the lamp does not turn on.

37. Digital circuits are frequently represented in a “truth table,” which shows the relationship between that circuit’s inputs and outputs. Amano, 3. The circuit above, for example, has a truth table with four rows, one for every combination of its two inputs:

| Inputs   |          | Output |
|----------|----------|--------|
| Switch A | Switch B | Lamp   |
| 0        | 0        | 0      |
| 0        | 1        | 0      |
| 1        | 0        | 0      |
| 1        | 1        | 1      |

For a binary circuit with  $K$  inputs, there would be  $2^K$  rows in a truth table if each possible input pattern is represented. *Id.* For example, a two-input circuit would have a complete truth table with  $2^2$ , or 4 rows, and a three-input circuit would have a complete truth table with  $2^3$ , or 8 rows, and so on. These truth tables describe the output for every possible input pattern the circuit can implement.

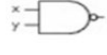
38. To construct digital circuits, designers use basic building blocks called “logic gates.” Logic gates implement fundamental Boolean logic functions on binary inputs—AND, OR, and NOT—and combinations of those functions. Amano, 3-4. A logic gate can be defined by its truth table. Illustrated below are six logic gates and their associated symbols and truth tables:

AND



| x | y | z |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

NAND



| x | y | z |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

OR



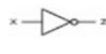
| x | y | z |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

NOR



| x | y | z |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

NOT



| x | z |
|---|---|
| 0 | 1 |
| 1 | 0 |

EXOR



| x | y | z |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Amano, 5. Note that the truth table indicates only the logical function that a circuit implements, not the way that the circuit is implemented in hardware. In fact, there are many different ways to implement a particular logic gate using physical transistors.

39. To design complex digital logic circuits, engineers can combine many logic gates to form a more complicated logic function. Amano, 4-5, 33-34. Even complex logic functions can be represented in a truth table, such as shown below for a three-input majority-vote function:

| A | B | C | f |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

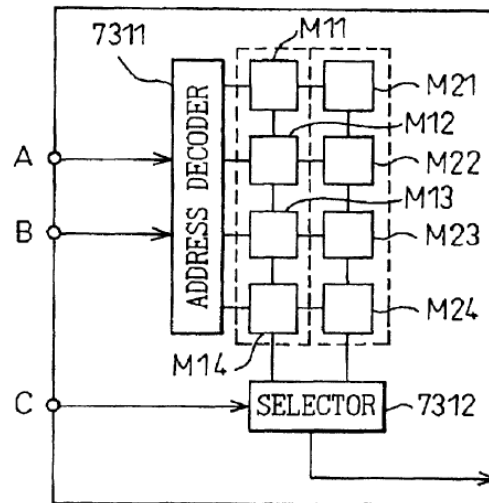
(a) Truth table

Amano, 34 & FIG.2.9. As the truth table shows, the function (f) is true (1) if, and only if, a majority of its inputs (A, B, and C) are true (1). The majority-vote truth table shows the resulting output for every possible input pattern. Thus, the truth table completely describes this more complex logic function. Also, just like simple logic gates, a complex logic function can be implemented using a variety of hardware configurations. *See* Singh, 281-82.

## B. Look-Up Tables

40. Both simple and complex logic functions can be implemented in hardware using look-up tables (often shortened to “LUTs”). A look-up table is a “digital memory” that can implement a logic function by storing the values for that function’s truth table in its memory. Brown-FPGA, xv, 51; Singh, 282; Bhat, 18; Fujii, ¶5. A digital memory contains an array of memory cells, each of which can store a single binary value (i.e., a 0 or a 1, or “bit”). These binary values can be read (i.e., output) from the memory using an “address,” which specifies the location (row and column) of the bit value(s) that should be output. *See, e.g.,* Bhat, 18.

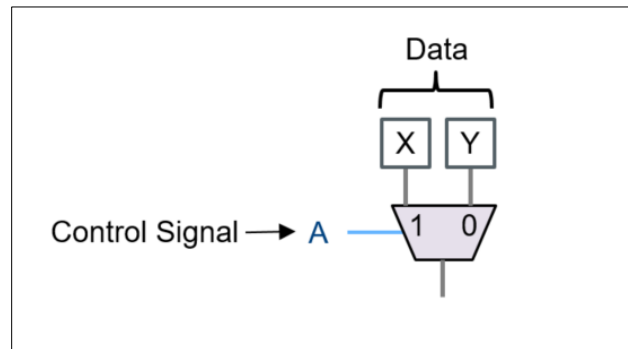
41. For example, consider the following digital memory described in the '737 patent:



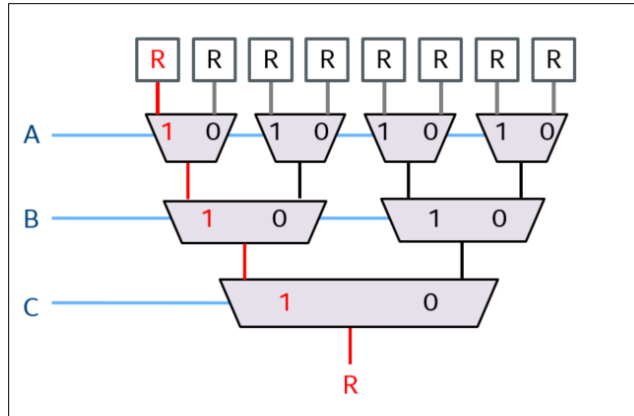
'737-patent, FIG.7. This memory contains a four-by-two array of eight bits (M11-M14 and M21-M24) that are addressed using three binary values (A, B, and C). Together, the values for A and B identify one of four rows using the “address decoder.” Then, the value for C identifies one of two columns using the “selector.” In this way, a specific combination of values for A, B and C identifies exactly one of the eight memory cells in the array.

42. Other configurations can be used to address a digital memory. In particular, a memory array of a given size can have a differing number of rows and columns. For example, an eight-bit memory could be implemented as shown above with four rows and two columns, but could also be implemented as eight rows and a single column or a single row and eight columns. In a memory array, rows are selected by an address decoder and columns are selected using a selector, or

multiplexer. If there are many columns in the array, the specific column being addressed can be selected using a tree of multiplexers. *See, e.g.,* Fujii, FIG.1; Cong, FIG.7; Amano, 37. A multiplexer is a circuit component (often depicted as a trapezoid) that selects one output from multiple data inputs based on binary control signals:

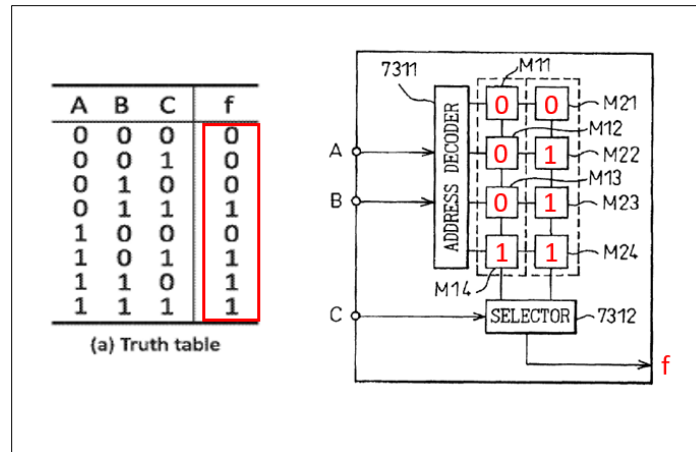


*See also* Trimberger, 21. In this example of a 2-1 multiplexer, when the control signal A is HIGH (1), the multiplexer outputs X, and when the control signal A is LOW (0), the multiplexer outputs Y. A multiplexer capable of selecting more than two inputs can be implemented by stacking multiplexers into a tree. *See, e.g.,* Fujii, FIG.1; Cong, FIG.7; Amano, 37. For example, a tree of seven 2-to-1 multiplexers can be used to implement an 8-1 multiplexer, which could be used to select one of eight columns in a memory array:



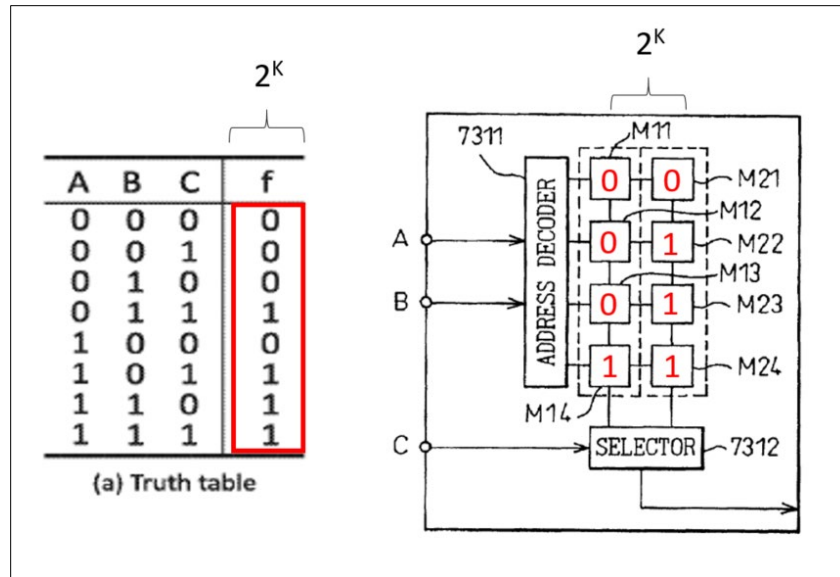
In this example, the memory array is arranged as a single row with eight columns, so no row decoders are needed. Each level of multiplexers is controlled by a control signal (A, B, or C). The values of these input signals identify the single memory bit that should be output. In this example, if A, B, and C are HIGH (1), the left-most memory bit is output (shown in red).

43. Look-up tables implement logic circuits by storing the values of the circuit's associated truth tables in the memory cells. *See, e.g., Brown-FPGA, xv, 51; Amano, 33-41.* The memory of the look-up table stores each possible output for a circuit, and the input signals are used to address that memory. As an example, consider how the memory at paragraph 41 above could be used to implement a majority-vote function described above in paragraph 39:



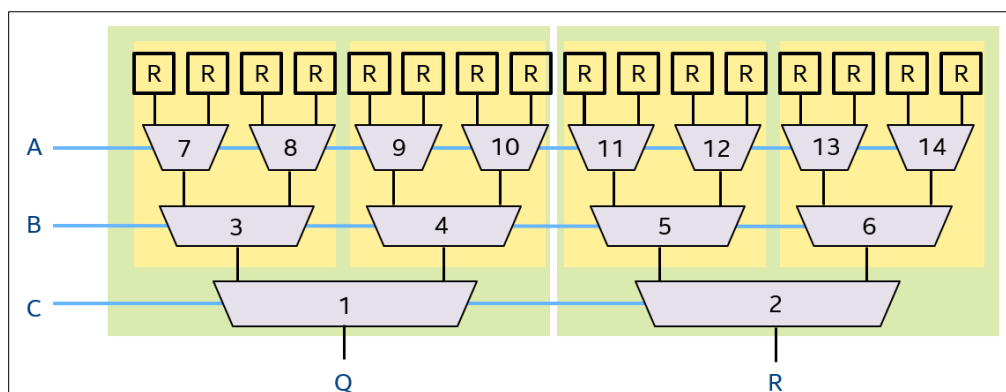
The outputs for the truth table (column f) are programmed into the memory array of the look-up table (from left to right, top to bottom). Then, the values for A, B, and C are used to “look up” the correct location in the memory when the device is used. In this way, the look-up table implements the majority-vote function’s relationship between inputs and outputs by storing the values of its truth table.

44. In general, a look-up table with at  $2^K$  bits can implement any  $K$ -input 1-output logic function. Every possible  $K$ -input logic function can be completely represented with a truth table of exactly  $2^K$  rows. *See supra*, ¶37. Thus, with  $2^K$  memory bits, a look-up table can store the values of one entire output column of a  $K$ -input truth table:



Thus, such a look-up table can implement any arbitrary logic function of  $K$  inputs and one output. *E.g.*, Brown-Rose, 52; Brown-FPGA, xv, 51; Fujii, ¶5; Sharpe-Geisler, 1:10-48, FIGS. 1-3. By changing the values that are stored in the memory of that look-up table, the function can be changed and thus an arbitrary function of  $K$ -inputs and one output can be implemented.

45. Another feature of look-up tables is that they often can be viewed as a collection of smaller look-up tables. For example, consider this look-up table with three inputs and two outputs:



46. This single device may be viewed as being comprised of four smaller look-up tables with the components shown inside the yellow boxes, each having *two* inputs (A and B), with the outputs from the two yellow look-up tables on the left proceeding to output selector 1 (i.e., an output selector outside the look-up table), and the outputs from the two yellow look-up tables on the right proceeding to output selector 2 (also outside the look-up table). Alternatively, this device may be viewed as having two (rather than four) smaller look-up tables with the components shown in the green boxes, each having *three* inputs (A, B, and C), with *no* selectors connected to outputs Q and R from the look-up tables. Thus, whether or not the smaller or the larger look-up tables within the device are considered to be distinct components impacts whether the overall device contains selectors that lie outside the component look-up tables.

47. As of the effective filing date of the '737 patent, a POSA would have understood that there were trade-offs involved in choosing a look-up table of a particular size or with a particular number of inputs or outputs. For example, using a high number of inputs (more than three or four) might be beneficial for certain user functions resulting in an increase in device speed and better utilization, but might be detrimental for other user functions resulting in a decrease in device speed and lower utilization. Trimberger, 27, 66; Brown-FPGA, 87-115; He, Abstract (collecting studies).

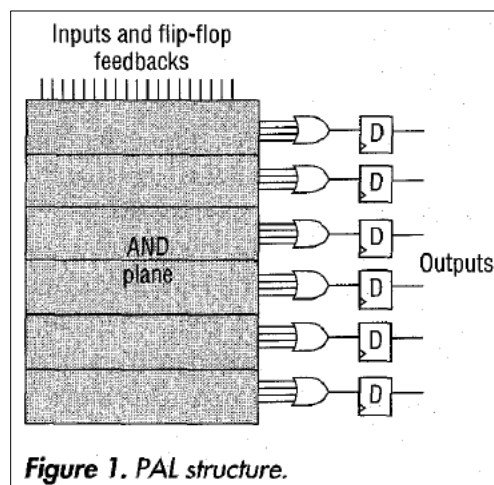
48. In addition, as of the effective filing date of the '737 patent, a POSA would have understood that there were advantages to providing multiple outputs for a look-up table that could be divided into smaller tables, as doing so would permit decreasing the total number of logic blocks for certain use cases and make up for an increase in the area required for routing structure structures due to the increased number of pins. *See, e.g.,* Brown-FPGA, 101, 103.

### **C. Overview of Programmable Logic Devices**

49. Unlike application-specific integrated circuits (“ASICs”), which are typically hardwired to perform a single function, programmable logic devices (“PLDs”) can typically be configured (and reconfigured) by customers to perform a wide variety of desired functions. *See* Brown-FPGA, 1-2. A customer can buy a PLD, such as a field programmable gate array (“FPGA”), off the shelf and configure that device to implement a logic circuit. FPGAs are *field programmable* for all sorts of functions.

50. Over time, PLDs have improved to achieve greater logic density (i.e., amount of logic per circuit area), faster speed, reduced power consumption, and increased flexibility. While the terminology sometimes varies, PLDs generally can be divided into a few categories: simple programmable logic devices (“simple PLDs”), complex programmable logic devices (“complex PLDs”), and field programmable gate arrays (“FPGAs”). Brown-FPGA, 3; *see also* Fujii, ¶¶2-4.

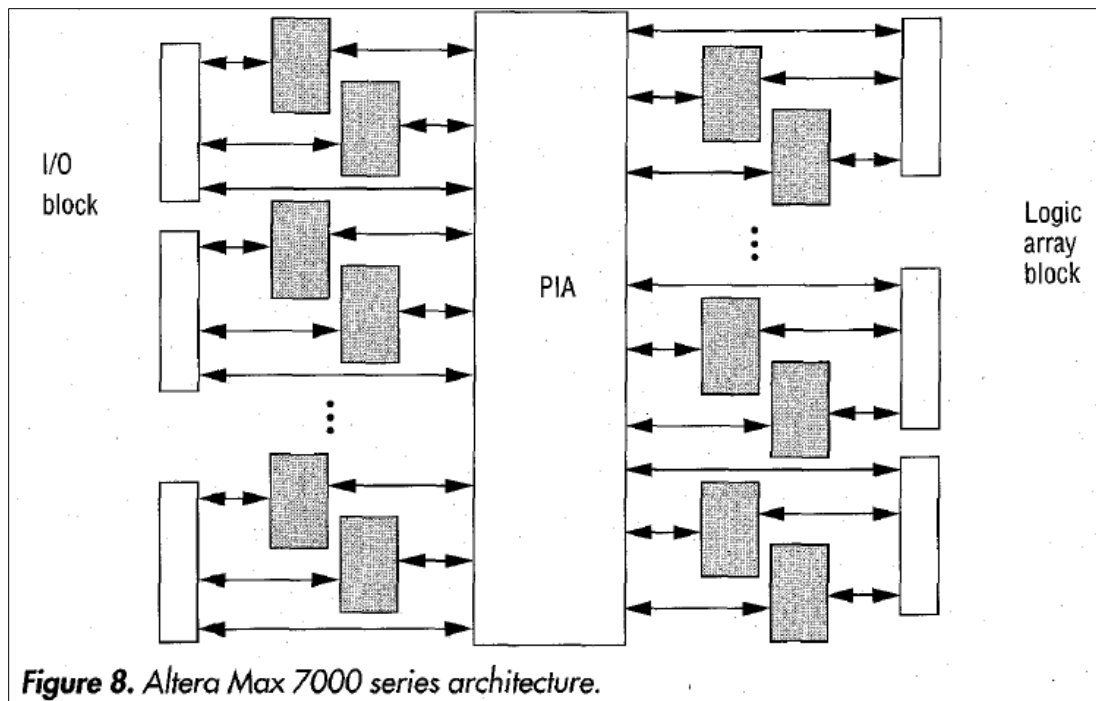
51. Simple PLDs were among the first field-programmable devices. For example, in the early 1970s, a simple PLD called the Programmable Logic Array (“PLA”) was developed. *See, e.g.*, Brown-Rose, 42-43. It featured two stages of programmable logic, a plane of AND gates and a plane of OR gates, and was capable of handling relatively large numbers of data inputs and outputs. Brown-Rose, 42-43; *see also, e.g.*, Brown-FPGA, 3. Because of this structure, PLAs were particularly well suited for sum-of-products applications. Brown-Rose, 42-43. At the time, drawbacks of PLAs were that they were expensive to manufacture and offered somewhat poor speed-performance. Brown-Rose, 43. Subsequently, another simple PLD, the Programmable Array Logic (“PAL”) device, was designed. Brown-Rose, 43. PALs fixed the OR plane, limiting flexibility but increasing speed. Brown-Rose, 43.



Brown-Rose, FIG.1.

52. To address some of the loss in flexibility, devices were developed that varied the structure of the PAL. Brown-Rose, 43-44. Still, simple PLDs were limited in their logic capacity. Brown-FPGA, 3. This is because the programmable logic plane grows quickly in size relative to an increase in the number of inputs for a large class of desired logic functions. Simple PLDs also consumed significant power, limiting their size to the order of product terms. *See* Trimberger, 15.

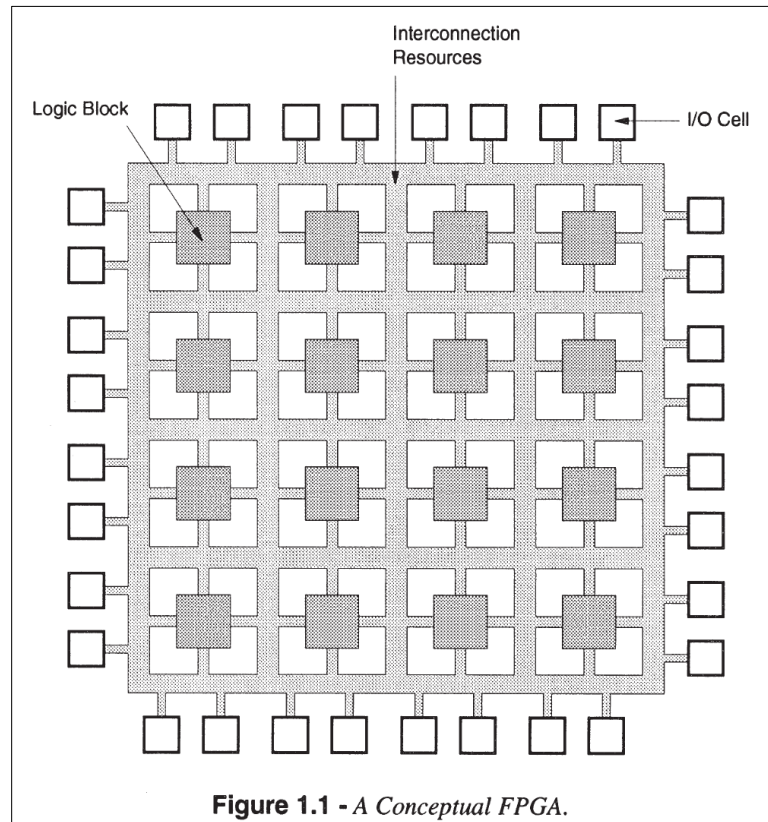
53. Complex PLDs were designed to address the limits on logic capacity by effectively integrating multiple simple PLDs onto the same chip. For example, Altera's MAX 7000 connected an array of Logic Array Blocks ("LABs"), a simple PLD-like structure, using programmable interconnect wires connecting every LAB to every other LAB. Brown-FPGA, 30-33, Brown-Rose 47-48, FIGS. 8-10. Like most (but not all) devices referred to as complex PLDs, the MAX 7000 relies on AND-planes coupled to OR-planes to perform the user's logical operations.



Brown-Rose, FIG.8. By stringing simple PLDs together, complex PLDs could realize reasonably complex designs, and by using cell-to-cell connectivity, complex PLDs provided relatively high speed-performance. However, power consumption and signal delays limited the size of some complex PLDs. *See, e.g.,* Trimberger, 79.

54. FPGAs, first introduced in the mid-1980s, long before the '737 patent's purported priority date, have a similar overall architecture to complex PLDs. *See* Trimberger, 17-18 (explaining how its definition of "FPGA" overlapped with another definition of complex PLD). FPGAs, by definition, consist of a two-dimensional array of uncommitted elements (logic blocks) that can be

interconnected. *See* Brown-FPGA, 4-6. A general depiction of FPGA architecture is provided below:



Brown-FPGA, 5 FIG.1.1.

55. Logic is implemented in an FPGA by portioning the desired logic function into small logic chunks, then assigning those chunks to logic blocks. *See, e.g.,* Brown-FPGA, 6. The signals are then routed via wires to the relevant logic blocks and the logic blocks are coupled together to complete the broader function. This requires programmable connections among the interconnect wires and between the interconnect and the logic blocks. Otherwise, the FPGA could not be

programmable to perform different functions involving more than one logic block. Brown-FPGA, 6.

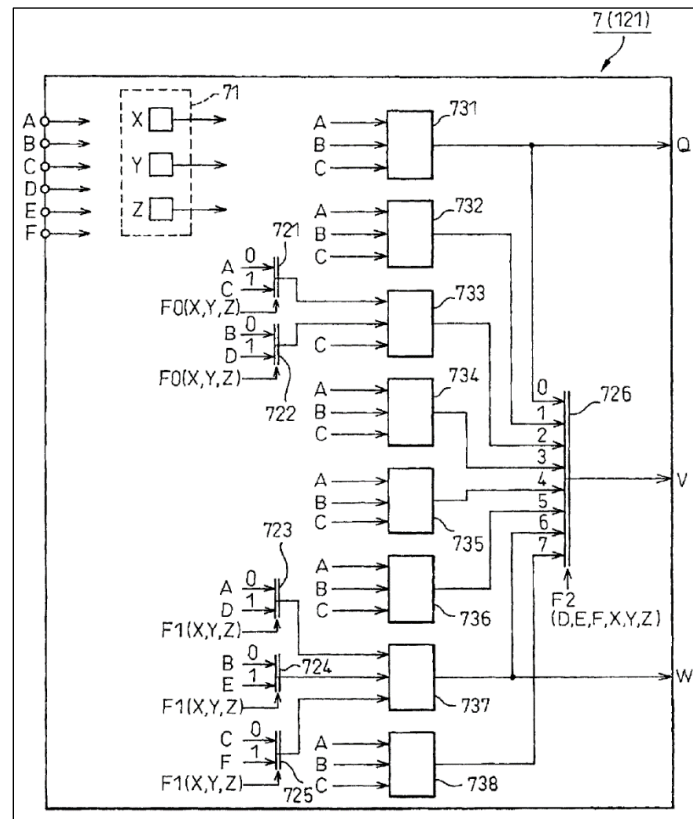
56. Unlike complex PLDs, FPGAs typically use look-up tables to implement a user's logic in their logic blocks. *See, e.g.*, Brown-FPGA, 5. Often, “implementing logic in a lookup table produces a smaller, faster FPGA because the lookup table efficiently combines versatile logic implementation with the programming overhead of the memory cells.” Trimmerger, 27 (discussing design choices for block size and structure).

## **VI. THE '737 PATENT**

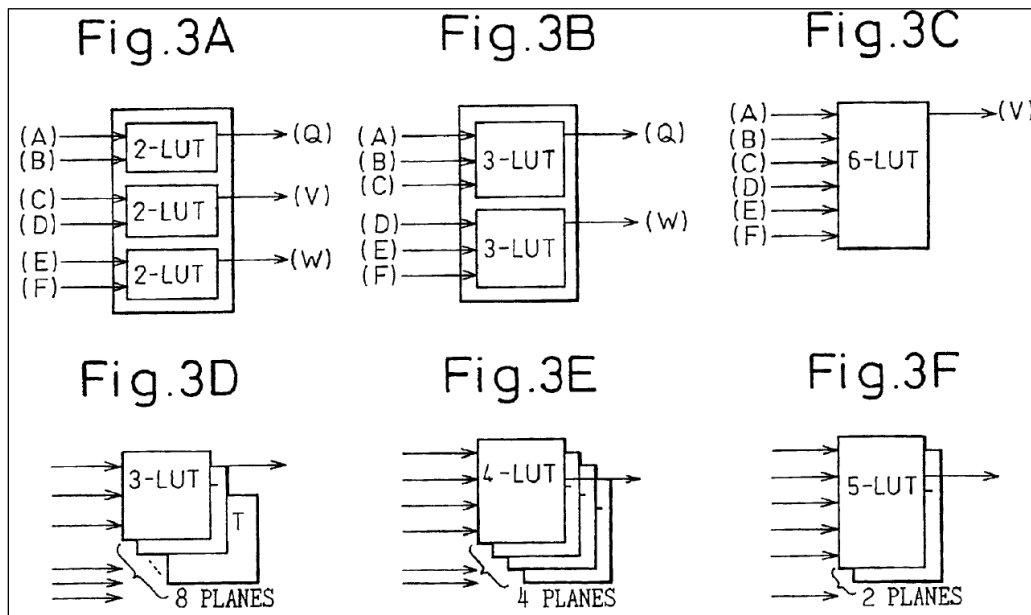
### **A. Preferred Embodiments**

57. The '737 patent relates to look-up tables (“LUTs”) used to implement logic circuits in programmable logic circuit devices, also known PLDs, which include FPGAs. *See, e.g.*, '737-patent, Abstract, 1:13-28. The '737 patent acknowledges that the “[b]asic architecture of such a[n] FPGA has been previously disclosed” ('737-patent, 1:28-30), and therefore, it omits details of FPGA architecture, such as details about routing wires, switch circuits, connection blocks, and input/output (“I/O”) blocks. '737-patent, 3:54-4:3 (“The present invention relates mainly to the configuration of LUT **121**” in FIG.1, rather than to FPGA **100**). The patent also admits that a “[m]ulti-context and cluster architecture of plural LUTs in FPGA” was also already known in the art. '737-patent, 1:30-32.

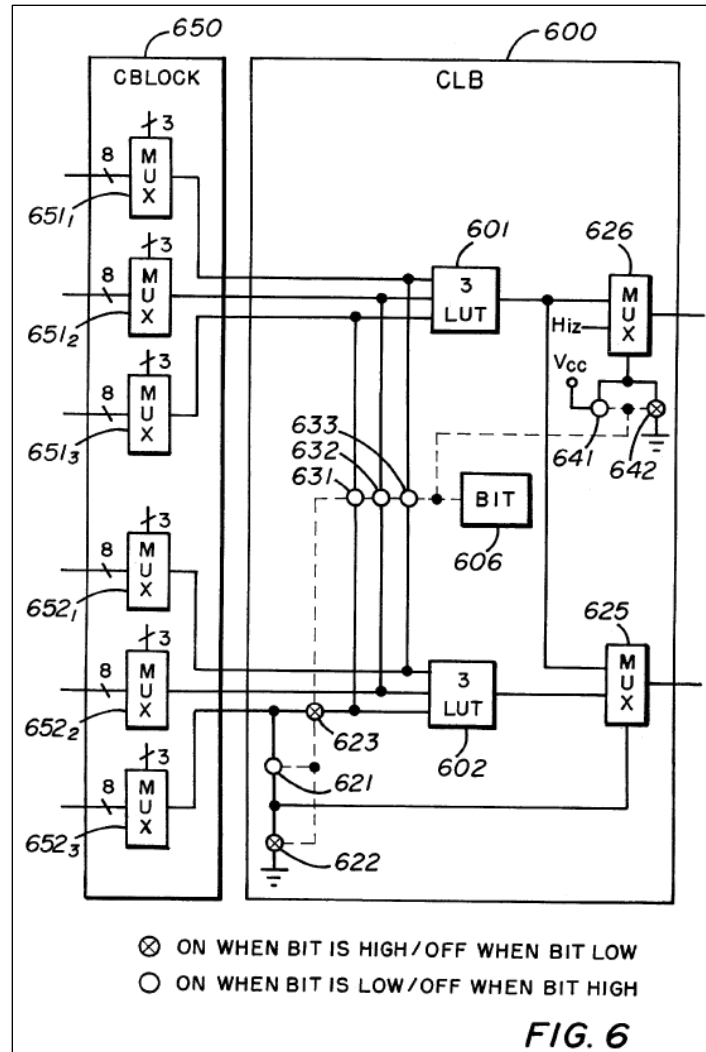
58. The '737 patent purports to address the problem of how, “in a circuit having input bit number less than M, one LUT needs to be assigned to this circuit, leading to [an] unnecessary increase of implementation area (logic and routing circuitry area; circuit area).” ’737-patent, 1:59-63. The patent purports to solve this alleged problem by disclosing a “6-input 3-output LUT” that can, “by varying the values (X, Y, Z) of mode changing memory (71), the 6-input 3-output LUT can be set to different internal configurations as shown in FIGS. 3A to 3F.” ’737-patent, 4:12-17. For example, FIG.6 depicts one of the preferred embodiments of the alleged invention of the ’737 patent:



'737-patent, FIG.6, 2:64-67, 3:11-16, 3:46-48, 5:24-61, 6:15-7:21. This figure shows a “6-input 3-output LUT 7 (121)” (i.e., the large box encompassing the other components) that contains “a mode changing memory 71, selectors 721~726, and 3-input 1-output LUT units (memory array) 731~738.” '737-patent, 5:24-29. Five of the six selectors (721 through 725) are “for selecting input signal[s]” to the “LUT units.” '737-patent, 5:26-61, 6:15-7:21, FIG.6. One selector (726) is “for selecting output signal[s]” from the “LUT units.” '737-patent, 5:55-60. “[B]y varying the values (X, Y, Z) of mode changing memory (71),” which control selectors 721 through 726, “the 6-input 3-output LUT can be set to different internal configurations as shown in FIGS. 3A to 3F,” reproduced below. '737-patent, 4:15-17, 4:18-47, 6:15-7:21.



59. Such techniques were not new. For example, Sharpe-Geisler (cited at '737-patent, 1:30-32), had explained that it was known that (1) a look-up table with  $M$  inputs may be composed of two smaller look-up tables of  $M-1$  inputs each and (2) two look-up tables of  $M$  inputs each may be combined to form a look-up table of  $M+1$  inputs. Sharpe-Geisler, 1:65-2:16; *see* Rose, 1021 (“[A]ny  $K$ -LUT can be implemented by two  $(K-1)$ -LUT’s connected by a two-to-one multiplexer, where the  $K$ th input is the input to the multiplexer. If the outputs of the smaller LUT’s are made accessible, then the entire block can be used as either two  $(K-1)$ -LUT’s or one  $K$ -LUT.”). Figure 6 of Sharpe-Geisler, reproduced below, illustrates an embodiment in which the “3 input LUTs 601 and 602” can be combined to form a 4-input look-up table or, alternatively, to provide their own separate outputs. Sharpe-Geisler, 3:16-24, FIG.6. The two alternative configurations, two 3-input look-up tables or a single 4-input look-up table, are controlled by a “single memory cell bit 606.” Sharpe-Geisler, 3:25-27, 4:56-5:35, FIG.6.



60. Similarly, Fujii, discussed below, had disclosed a K-input programmable logic element PL1 that was divided into “plural memory circuit groups,” each of which implemented a logic circuit of fewer than K inputs. This allowed PL1 to implement multiple functions and leave less memory unused. Fujii, ¶¶6, 9-10, 12, 14, 50, 78, 99, 102. PL1 therefore was “capable of efficiently realizing a desired logic function by using fewer logic elements and less memory.” Fujii, ¶¶9, 102.

61. Cliff likewise described “adding” elements “to a look-up table[’s] circuitry for allowing the look up table to be effectively partitioned into smaller look up tables” so that “adders and counters can be implemented more efficiently and with less waste of look up table resources.” Cliff, 3:4-35, 5:63-6:9, 31:64-35:26 (fourth embodiment).

62. Unlike the ’737 patent, the prior art details *how* look-up tables are partitioned into smaller look-up tables, thereby increasing their flexibility and efficiency.

## **B. Prosecution History**

63. I have reviewed the ’737 patent’s specification, the ’737 patent’s prosecution history, and the prosecution history of the Japanese parent application, Japanese Application No. 2001-199644, filed on June 29, 2001, to which the ’737 patent claims priority.<sup>1</sup> *See* ’737-patent, 1; Ex.1002 (’737-patent file history); Ex.1040 (Japanese patent); Ex.1010 (Japanese patent file history).

64. The Examiner’s Reasons for Allowance for the ’737 patent stated that, “although the prior art of record discloses [a] control circuit having a memory for controlling a plurality of selectors for selecting LUTs’ input signals,” the prior art

---

<sup>1</sup> I understand that Japanese Application No. 2001-199644 issued as JP Patent No. 3580785B2 on October 27, 2004.

of record failed to teach an “internal configuration control circuit controlling and defining an internal configuration of the plurality of LUT units.” Ex.1002, 196.

65. None of the prior art references relied upon as invalidating grounds in this Petition were before the Examiner during prosecution, nor were similar disclosures discussed. ’737-patent, cover at (56); Ex.1002.

### **C. The Challenged Claims**

66. The ’737 patent has three independent claims. I have reproduced independent claims 1, 7, and 13 below with bracketed letters added for purposes of cross reference.

1. [1PRE] A look up table of M inputs and N outputs, comprising:

[1A] a plurality of LUT units; and

[1B] an internal configuration control circuit controlling an internal configuration of said plurality of LUT units,

[1C] wherein said internal configuration control circuit comprises a plurality of selectors selecting I/O signals of said plurality of LUT units, and

[1D] a selector control circuit having a memory, controlling said plurality of selectors in accordance with data stored in said memory, and defining the internal configuration of said plurality of LUT units.

7. [7PRE] A programmable logic circuit device comprising:

[7A] a plurality of logic blocks.[7B] a plurality of routing wires connected to each of said logic blocks.[7C] a plurality of switch circuits provided at an intersection of each of said routing wires.[7D] a plurality of connection blocks provided between an I/O line of each of said logic blocks and each of said routing wires; and

[7E] an I/O block performing an input/output operation with external equipment,

[7F] wherein each of said logic blocks has a look up table of M inputs and N outputs, comprising:

[7G] a plurality of LUT units; and

[7H] an internal configuration control circuit controlling an internal configuration of said plurality of LUT units,

[7I] wherein said internal configuration control circuit comprises a plurality of selectors selecting I/O signals of said plurality of LUT units, and

[7J] a selector control circuit having a memory, controlling said plurality of selectors in accordance with data stored in said memory, and defining the internal configuration of said plurality of LUT units.

13. [13PRE] A method of configuring a look up table of M inputs and N outputs, comprising:

[13A] providing a plurality of LUT units, and

[13B] selectively controlling I/O signals of said plurality of LUT units to set a predetermined mode of an internal configuration.

67. I discuss the dependent claims in connection with the relevant grounds below. I also list all the claims in an appendix at the end of this declaration.

| Ground Number and References |                                                                  | Claims           |
|------------------------------|------------------------------------------------------------------|------------------|
| 1                            | Fujii in view of Betz                                            | 1-5, 7-11, 13-16 |
| 2                            | Fujii in view of Betz and the general knowledge of a POSA        | 6, 12            |
| 3                            | Fujii in view of Betz and Chow                                   | 1-5, 7-11, 13-16 |
| 4                            | Fujii in view of Betz, the general knowledge of a POSA, and Chow | 6, 12            |

## VII. CLAIM INTERPRETATION

68. It is my opinion that for purposes of this proceeding the claim terms need not be expressly construed to resolve the prior art issues presented in this Petition, as the challenged claims are unpatentable under either party's proposed constructions.

69. Nothing in this section should be construed as expressing any opinion as to whether the claims constitute patentable subject matter under 35 U.S.C. §101, or whether they satisfy the definiteness, enablement, best mode, or written description requirements of §112.

70. I understand that Patent Owner and Petitioner have briefed two claim-construction issues in the District Court Case. The parties' positions are below. *See* Exs.1031-1036.

| <b>Claim Term</b>                                                                                               | <b>Patent Owner's Proposal</b>    | <b>Petitioner's Proposal</b>                                                                                                                                                                                                      |
|-----------------------------------------------------------------------------------------------------------------|-----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>"look up table of M inputs and N outputs"</p> <p>(claims 1, 7, 13, and all claims that depend therefrom)</p> | <p>Plain-and-ordinary meaning</p> | <p>"component that is composed of memories and is capable of implementing an arbitrary logic circuit of M inputs on at least one of N outputs"<sup>2</sup></p>                                                                    |
| <p>"LUT units"</p> <p>(all claims)</p>                                                                          | <p>Plain-and-ordinary meaning</p> | <p>Indefinite</p> <p>Otherwise, if not indefinite, each "LUT unit" is a "component that is composed of memories and is capable of implementing an arbitrary logic circuit of its inputs with at least one output"<sup>3</sup></p> |

---

<sup>2</sup> I understand that, in response to Patent Owner's arguments in the District Court Case, Petitioner agreed to drop language from its proposed construction that would have required the "*look up table of M inputs and N outputs*" to be part of a programmable logic circuit device. Ex.1034, 2 n.1.

<sup>3</sup> *See supra* note 2.

**A. “look up table of M inputs and N outputs”**

71. The '737 patent defines “*look up table of M inputs and N outputs*” in two parts. First, it defines a “*look up table*,” stating that “[a] look up table (LUT) as a major component of a programmable logic circuit device (PLD or FPGA) ***is composed of memories, and is capable of implementing an arbitrary logic circuit.***” ’737-patent, 1:25-28.<sup>4</sup> Second, the patent defines what it means for a look-up table to have “*M inputs and N outputs*,” stating:

In a programmable logic circuit device, the number of bits M of input signal for the logic circuit that can be implemented corresponds to the number of bits of the address of memory composing the LUT, and the number of bits N of output signal corresponds to the number of output bits of the memory. Thus, a logic circuit that can be implemented in one LUT ***is an arbitrary logic circuit of M inputs and N outputs.***

Such a LUT ***is*** denoted in the present specification as M-input N-output LUT.

’737-patent, 1:46-54.

72. Although Patent Owner contends that this term does not require construction, the parties agree that the plain meaning of a “*look up table of M inputs and N outputs*” is a component that is composed of memories and is capable of implementing an arbitrary logic circuit of M inputs on at least one of N outputs.

---

<sup>4</sup> Emphasis added unless otherwise noted.

See Ex.1032, 3-4, 7-11; Ex.1035, 1-2. The parties also agree that a look-up table having  $M$  inputs requires  $2^M$  memory bits to be capable of implementing any arbitrary  $M$ -input logic function. See Ex.1032, 3-4 (recognizing an  $M$ -input look-up table requires  $2^M$  memory bits), 7-11; Ex.1035, 1-2; *see also* Ex.1033, ¶¶21, 23 (“A look up table is a ‘digital memory’ that can implement a logic function by storing the values for that function’s truth table in its memory cells” and “a LUT having  $k$  inputs requires  $2^k$  memory bits.”). Finally, the parties agree that, while the claimed “look up table of  $M$  inputs and  $N$  outputs” may have more than one output, it need be capable of performing an arbitrary function of  $M$  inputs on **only one** of those outputs. Ex.1035, 2. This is consistent with the disclosure in the ’737 patent of two preferred embodiments of a “6-input 3-output” look-up table, each having only 64 memory bits (i.e., each of the eight “LUT units” 731-738 in FIG.6 has 8 memory bits, and each of the four “LUT units” 831-834 in FIG.9 has 16 memory bits). ’737-patent, 3:46-53, 5:24-32, 5:62-6:14, 7:21-31, 7:45-8:21, FIGS.6-12. These memory bits are sufficient to implement any function of 6 inputs on one output ( $2^6 \times 1 = 64$  bits required), but they cannot implement any function of 6 inputs on three outputs. To implement any function of 6 inputs on a second output independent of the first output, the look-up table would need an additional 64 ( $2^6$ ) bits of memory ( $2^6 \times 2 = 128$  bits required). To implement any function of 6

inputs on all three outputs independently, the look-up table would need an additional 128 ( $2^6 \times 2$ ) bits of memory ( $2^6 \times 3 = 192$  bits required).

73. In summary, a “*look up table of M inputs and N outputs*” is a “component that is composed of memories and is capable of implementing an arbitrary logic circuit of M inputs on at least one of N outputs.”

**B. “LUT units”**

74. My opinions on validity do not depend on the interpretation of “LUT units.” I understand that, in the District Court Case, Petitioner has argued that “*LUT units*” are smaller look-up tables into which a larger “*look up table*” may be divided.<sup>5</sup> See Ex.1031, 13-15. Patent Owner contends that the plain-and-ordinary meaning of “LUT unit” is “an individual LUT that cannot be fractured or divided *into smaller LUT units*.” Ex.1035, 3 (emphasis in original). As discussed more fully below, *see infra* [1A], it is my opinion that the Challenged Claims would have been obvious under either parties’ proposed construction.

---

<sup>5</sup> Petitioner also has argued that, under the plain meaning of “LUT units,” the claims are indefinite because a single implementation can both infringe and not infringe the claims simultaneously. Ex.1031, 13-19; Ex.1034, 4-10.

## **VIII. GROUND 1**

75. After reviewing Fujii, Betz, and claims 1-5, 7-11, and 13-16 of the '737 patent, it is my opinion that each of these claims would have been obvious to a POSA in view of Fujii in combination with Betz (Ground 1). The basis for my opinion and the details of my analysis are below.

### **A. Overview of Fujii and Betz**

#### **1. Fujii (Ex.1005)<sup>6</sup>**

76. Fujii is a Japanese Unexamined Patent Application Publication that was published on May 6, 1997. I have been informed that Fujii meets the requirements to be prior art to the '737 patent.

77. Fujii explains that it “relates to a programmable logic element, which is a unit for implementing logic functions in a programmable [Large Scale Integration (LSI)] whose logic functions can be programmed by a user.” Fujii, ¶1. It further explains that FPGAs are a conventional type of programmable LSI, Fujii,

---

<sup>6</sup> Exhibit 1005 is a compilation containing the English-language translation of Fujii, which is relied upon here, an affidavit as required by 37 C.F.R. §42.63(b) (in the form of a declaration as permitted by 37 C.F.R. §42.2), and a copy of the original publication in Japanese.

¶3, FIG.22, and look-up tables are the most common form of “programmable logic element,” Fujii, ¶5.

78. In particular, Fujii relates to the same problem as the '737 patent, that implementing a small function in a larger LUT means that some memory circuits go unused. Fujii, ¶¶5-6 (“[I]f we assign to a K-input LUT a logic function whose actual number of inputs  $i$  is less than K, only  $2^i$  bits of memory are used to store the program that implements the above logic function, and therefore,  $(2^K - 2^i)$  bits of memory remain in the LUT as memory not used for the above program (invalid memory).”); '737-patent, 1:59-63 (“[I]n a conventional FPGA, even if circuit division results in a circuit having input bit number less than M, one LUT needs to be assigned to this circuit, leading to unnecessary increase of implementation area.”), 4:18-25.

79. Fujii addresses this problem by providing a K-input programmable logic element divided into “plural memory circuit groups.” Fujii, claim 1, ¶¶12, 14, 54, 80. Each memory circuit group implements an arbitrary logic circuit of less than K inputs, allowing the programmable logic element to implement a smaller logic function (“less than K” inputs) so that a single programmable logic element implements “a plurality of logic functions,” leaving less memory unused. Fujii, ¶¶6, 9-10, 50 (“When a two-input logic function is assigned to a conventional four-input programmable logic element, the remaining memory circuits other than the

4-bit memory circuit required for programming are unused, which was one of the causes of lowering the degree of integration. In the programmable logic element PL1, however, the remaining memory circuits can also be effectively used for assignment of other logic functions.”), 78 (similar for PL2), 99 (similar for PL3), 102. PL1 thus is capable of realizing a “desired logic function” using “fewer logic elements and less memory” (Fujii, Abstract, ¶¶9, 102), which in turn will increase utilization and reduce power consumption.

80. As discussed in more detail below, Fujii switches between operations (i.e., use modes that can implement certain-sized functions using the plural memory circuit groups, *see infra* [1B]) of programmable logic element PL1 using “control signals” A and B. *See, e.g.*, Fujii, ¶¶11-12, 36-39, 42, 45, 51; FIGS.1, 6.

## **2. Betz (Ex.1006)**

81. Betz is a reference book on high-performance FPGA architectures and related CAD tools written by Vaughn Betz, Jonathan Rose, and Alexander Marquardt and published in 1999. I have been informed by counsel that Betz meets the requirements to be prior art to the '737 patent.

82. At the priority date, the second author of the Betz reference, Jonathan Rose, was a well-known researcher in the field of programmable hardware and the book was co-authored by PhD and MS students he advised. The Betz reference

provides an overview of FPGAs, including logic block and routing architecture.

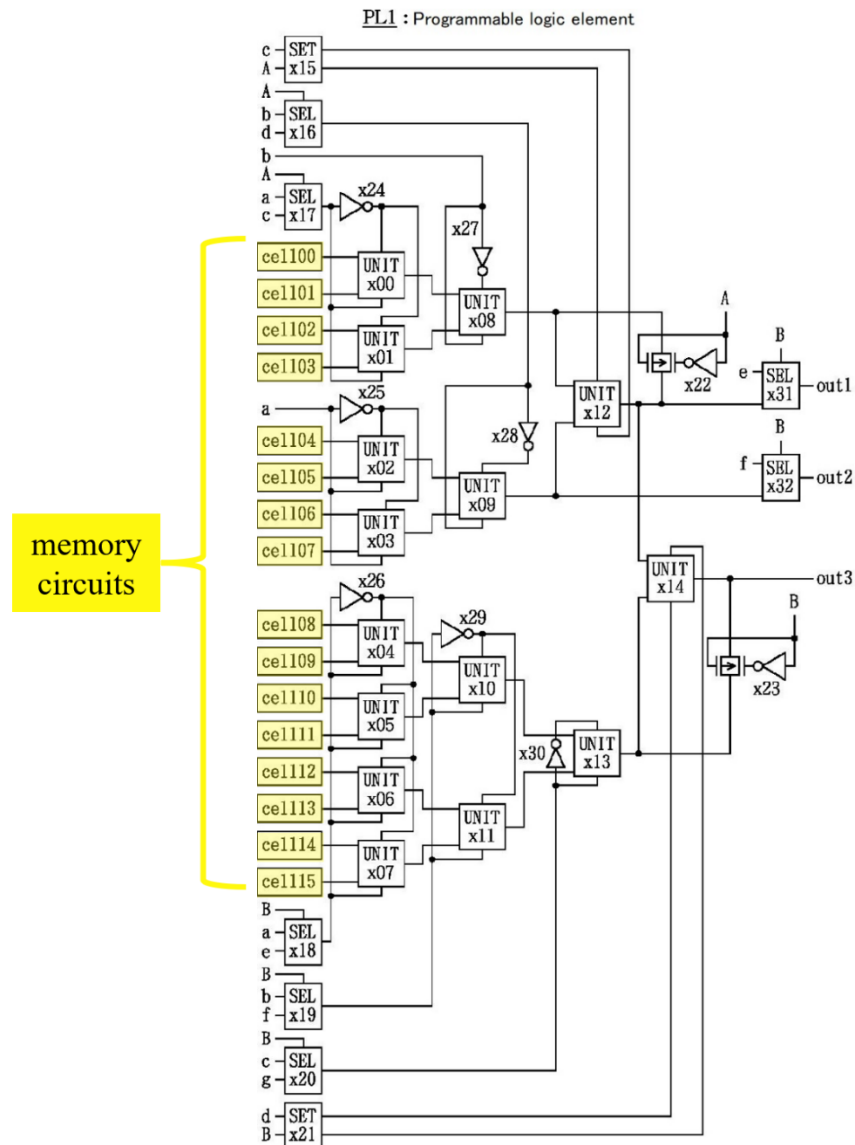
*See* Betz, 2-7, 11-18, 105-149, 207-17.

## **B. Independent Claim 1**

### **1. [1PRE] “A look up table of M inputs and N outputs, comprising”**

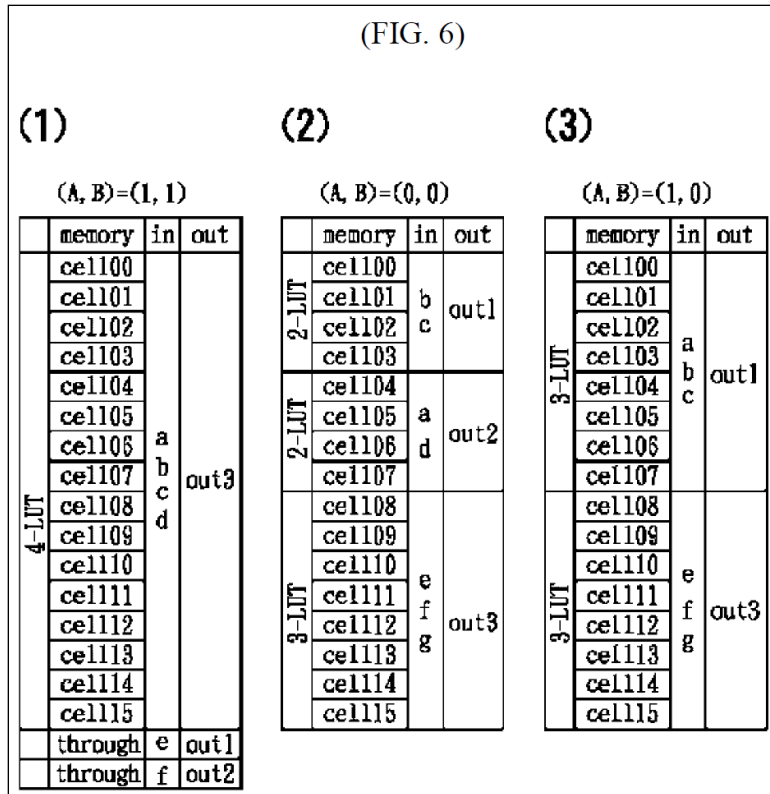
83. Fujii discloses “[a] look up table of *M* inputs and *N* outputs” under the parties’ agreed-upon interpretation of that phrase: a “component that is composed of memories and is capable of implementing an arbitrary logic circuit of *M* inputs on at least one of *N* outputs.” *See supra* §VII.A. In particular, Fujii discloses PL1. Fujii, FIG.1 (depicting PL1); *see also* Fujii, FIGS.8 (depicting PL2, which is substantially similar in structure and function); 15 (depicting PL3, which is substantially similar in structure and function).

84. PL1 is a “component that is composed of memories” because it has “16 memory circuits.” Fujii ¶¶12, 14. Fujii’s FIG.1 (below) depicts PL1 and illustrates these 16 memory circuits (yellow):



Fujii, FIG.1 (annotated).

85. These memory circuits also are shown in FIG.6. Fujii, ¶¶36-38 (“In FIG. 6...cell00 to cell15 indicate memory circuits to be used.”).

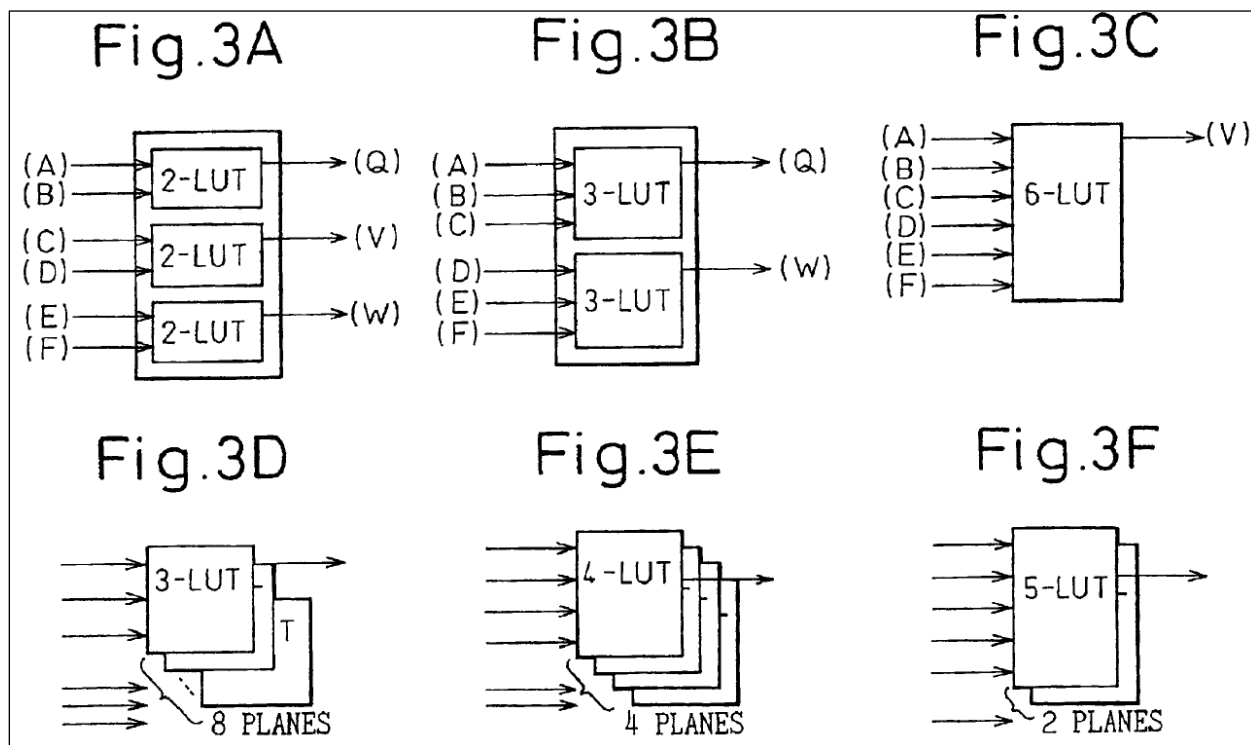


86. Fujii's PL1 is "capable of implementing an arbitrary logic circuit of M inputs on at least one of N outputs," as required by the parties' interpretation. PL1 has "16 [i.e.,  $2^4$ ] memory circuits," Fujii, ¶¶12, 14, allowing it to be "capable of implementing an arbitrary logic function of" up to and no more than 4 "inputs" (i.e., the use mode shown in FIG.6(1) implements an arbitrary logic function of a, b, c, and d). Fujii, ¶¶5-6, 14, 38, FIG.6(1). Thus, PL1 is a look-up table of 4 inputs. Notably, neither the claim language nor the parties' agreed-upon interpretation precludes the "*look up table*" from having more inputs than the M inputs required to implement an arbitrary logic circuit. So the fact that PL1 has additional inputs (e, f, and g) that are not used in the use mode shown in FIG.6(1) is of no import

under either the parties' agreed-upon interpretation or the claim language, as Patent Owner has recognized in the Japanese Case. *See* Ex.1041 (“[T]hroughout the entire claims, there is no language stipulating that ‘input...signals of said plurality of LUT units’ must match the ‘M inputs’ inputted into the ‘lookup table’ as a whole....[I]t is fine if all M inputs are the ‘input...signals of said plurality of LUT units’ of the ‘input and output signals of said plurality of LUT units’, and it is also fine if they are not.”).

87. Moreover, PL1 also has three “*outputs*”, out1, out2, and out3 that output the values of the selected memory bits according to the use modes shown in FIG.6. Fujii, ¶¶34-35, 37-38, 40-44, 47-48, FIGS.1, 6; *see also* ’737-patent, 1:55-58, 6:54-60, 7:17-20. Neither the claim language itself nor the parties’ agreed-upon interpretation precludes the “*look up table*” from having more outputs than required to implement an arbitrary logic circuit of M inputs on one output. The fact that PL1 cannot implement an arbitrary logic circuit of four inputs on each of its outputs (it performs such a function only on out3, *see* FIG.6(1)) is consistent with the ’737 patent’s preferred embodiments (FIGS.6, 9), which each have 3 outputs (Q, V, W) but are capable of implementing an arbitrary logic circuit of 6 inputs on only a single output. *Compare* ’737-patent, 3:46-53, 5:24-32, 5:62-6:14, 6:24-50, 7:21-31, 7:45-8:21, FIGS.6-12 *with* Fujii, ¶5; *see also supra* §VII.A. As shown in

FIG.3 (below), the preferred embodiments can only perform an arbitrary function of 6 inputs on a single output (FIG.3C):



See '737-patent, 4:39-40, 6:54-60. Thus, Fujii's PL1 is "capable of implementing an arbitrary logic circuit of M inputs [4 inputs] on at least one of its N outputs [3 outputs]."

88. In summary, Fujii discloses a component ["programmable logic element PL1"] that is composed of memories ["16 memory circuits"] and is capable of implementing an arbitrary logic circuit of M inputs ["out3 corresponds to the output terminal of a four-input programmable logic element"] on at least one of N outputs ["PL1 has...up to three output terminals"]. Fujii, ¶¶5, 12-14, 40.

Thus, Fujii's PL1 is "[a] look up table of *M* inputs and *N* outputs" under the parties' agreed-upon interpretation of that phrase.

89. Alternatively, if the claim is interpreted such that the "*look up table of *M* inputs and *N* outputs*" need not be capable of implementing an arbitrary logic circuit of *M* inputs, PL1 also could be considered a look-up table of up to 7 inputs (a, b, c, d, e, f, and g) and three outputs (out1, out2, and out3). Fujii, ¶38, FIGS.1, 6. So even under this interpretation, PL1 is still "[a] look up table of *M* inputs and *N* outputs," as recited by claim 1.

## **2. [1A] "a plurality of LUT units; and"**

90. Fujii discloses this limitation, because Fujii teaches that the programmable logic element PL1 has "plural memory circuit groups." *E.g.*, Fujii, claim 1, ¶¶12, 14. For example, programmable logic element PL1 has "three memory circuit groups each composed of 8, 4, and 4 memory circuits." Fujii, ¶12. Under either party's construction, these three memory circuit groups are a "plurality of LUT units." *See also* Fujii, ¶54, FIG.8 (substantially similar "memory circuit groups" in PL2); Fujii, ¶80, FIG.15 (substantially similar "memory circuit groups" in PL3).

91. Fujii discloses "plural memory circuit groups" within PL1's  $2^K$  memory circuits:

[A] second operation of selecting one output signal from the output signals of the  $2^{X_i}$  memory circuits and outputting the selected output signal as an output signal of the programmable logic element, *by dividing the  $2^K$  memory circuits into plural memory circuit groups*, where the sequence from  $i=1$  to  $i=n$  of the sequence  $\{2^{X_i}\}$  ( $2^{X_i} \leq 2^{X_{i+1}}$ ) is equal to  $2^{[K]}$ ,<sup>7</sup> the memory circuit group is composed of  $2^{X_i}$  ( $i=1, 2, \dots, n$ ) *memory circuits*, and  $X_i$ <sup>8</sup> *input signals that do not overlap each other among the input signals are used as selection signals*.

Fujii, ¶14, ¶¶12-13. Each memory circuit group has  $X_i$  input signals and  $2^{X_i}$  memory circuits and, therefore, has sufficient memory to implement an arbitrary logic circuit of  $X_i$  inputs on at least one output. Fujii, ¶5. Moreover, Fujii describes that the  $X_i$  input signals are “appl[ied]” to their memory circuit group. Fujii, ¶¶12, 54, 80.

92. Thus, Fujii discloses “*a plurality of LUT units*” under Petitioner’s construction of “*LUT units*”: it discloses components [“plural memory circuit

---

<sup>7</sup> The text “equal to  $2^K$ ” in this quote is a typographical error that should read “equal to  $2^K$ ”.

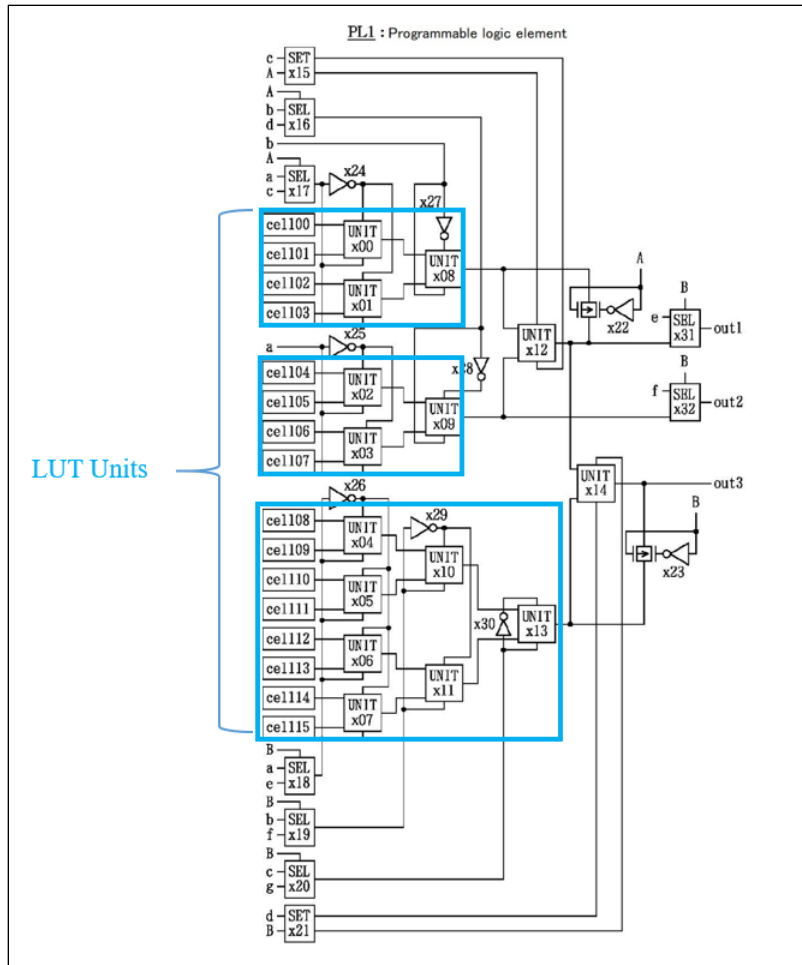
<sup>8</sup> The text “ $X_i$ ” in this quote and used elsewhere in Fujii is a notation that indicates that “ $X$ ” is a variable indicating the number of input signals applied to each memory circuit group, and the letter “ $i$ ” is an index (which can also be written as a subscript) used to distinguish the plurality of memory circuit groups.

groups”] that are each composed of memories [“ $2^{X_i}$  ( $i=1, 2, \dots, n$ ) memory circuits”] and are capable of implementing an arbitrary logic circuit of their inputs [“ $X_i$  input signals” for “selecting one output signal from the output signals of the  $2^{X_i}$  memory circuits”] with at least one output [“outputting the selected output signal as an output signal of the programmable logic element”].

93. For PL1, “the 16 storage elements are divided into  $4+4+8$  (divided into  $2^2+2^2+2^3$ ), and the sequence  $\{2^{[X]_i}\}$ <sup>9</sup> in this case is  $\{2^2, 2^2, 2^3\}$ .” Fujii, ¶¶45, 51. PL1 therefore has three memory circuit groups, namely a top “2-LUT” with four memory circuits, a middle “2-LUT” with four memory circuits, and a bottom “3-LUT” with eight memory circuits. See Fujii, FIG.6(2). These three “*LUT units*” are shown in annotated FIG.1 below:

---

<sup>9</sup> The lowercase “x” in this quote is a typographical error that should be a capital “X”.

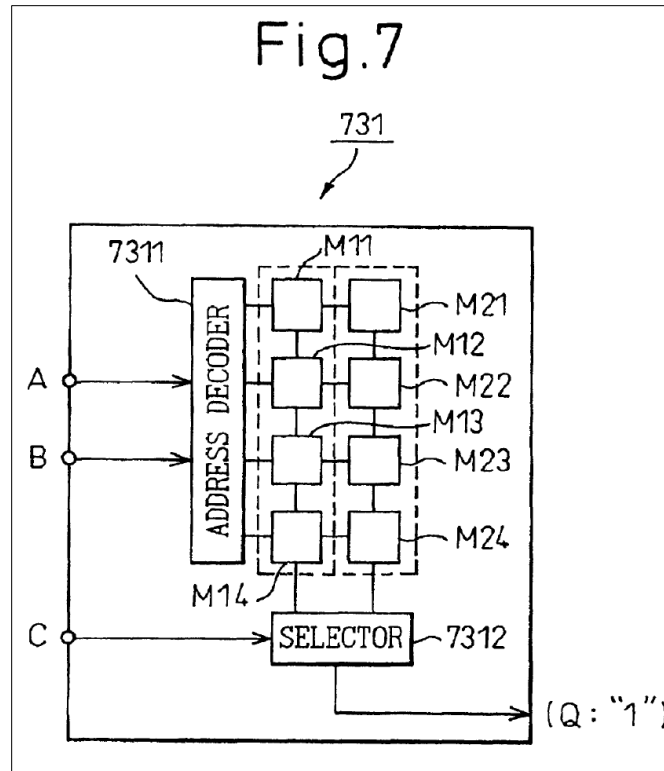


Fujii, FIG.1 (annotated); FIG.6(2). Although not entirely within the blue boxes (drawn as rectangles in full FIG.1 for convenience), the “*LUT units*” include inverters x24-30. These “*LUT units*” constitute a “*plurality of LUT units*” under Petitioner’s proposed construction.

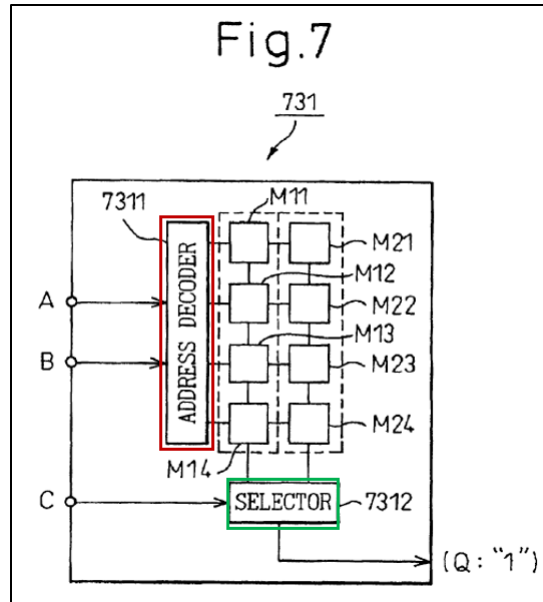
94. Fujii’s “three memory circuit groups” (Fujii, ¶12) also constitute “a *plurality of LUT units*” under Patent Owner’s proposed construction.

95. As discussed above (§VII.B), Patent Owner interprets the term “*LUT unit*” to mean “an individual LUT that cannot be fractured or divided into smaller

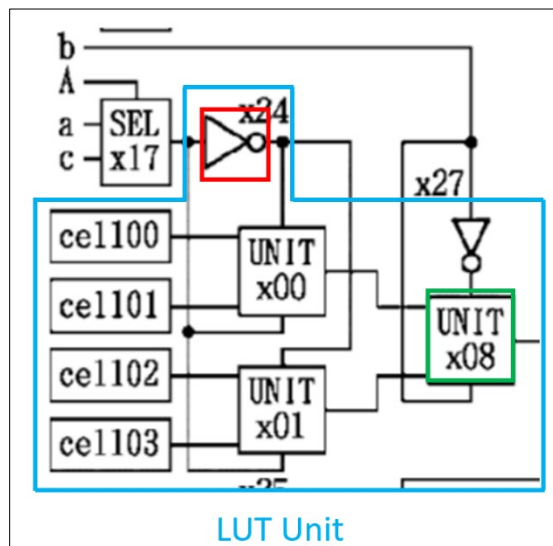
units.” Ex.1032, 14. Patent Owner also has argued that “whether a given look up table of M inputs and N outputs comprises a plurality of LUT units is not a matter of how simplified block diagrams of look up tables are *viewed*, but rather how look up tables are actually *implemented*.” Ex.1032, 14 (emphasis in original). I also understand that Patent Owner contends that the look-up tables shown in FIGS.7, 8, 10 and 11 of the ’737 patent are “exemplary” of the “*LUT units*” recited in the claims. See Ex.1032, 13-14 (“Figures 7 and 8 of the ’737 patent each show block diagrams of ***exemplary 3-input 1-output LUT units*** that can be used in the 6-input 3-output look up table depicted in Fig. 6.”), (“Figures 10 and 11 of the ’737 patent each show block diagrams of ***exemplary 3-input 2-output LUT units*** that can be used in the 6-input 3-output look up table depicted in Fig. 9.”), 17 (“The ’737 patent is abundantly clear to a person of ordinary skill in the art—a LUT unit is described as a ‘memory array’ and ***illustrated clearly in a variety of embodiments (e.g., Figures 6-11)***.”). It follows that Patent Owner believes the “*LUT units*” in FIGS.6-11 “cannot be fractured or divided into smaller units.” Ex.1032, 14. FIG.7 from the ’737 patent is shown below:



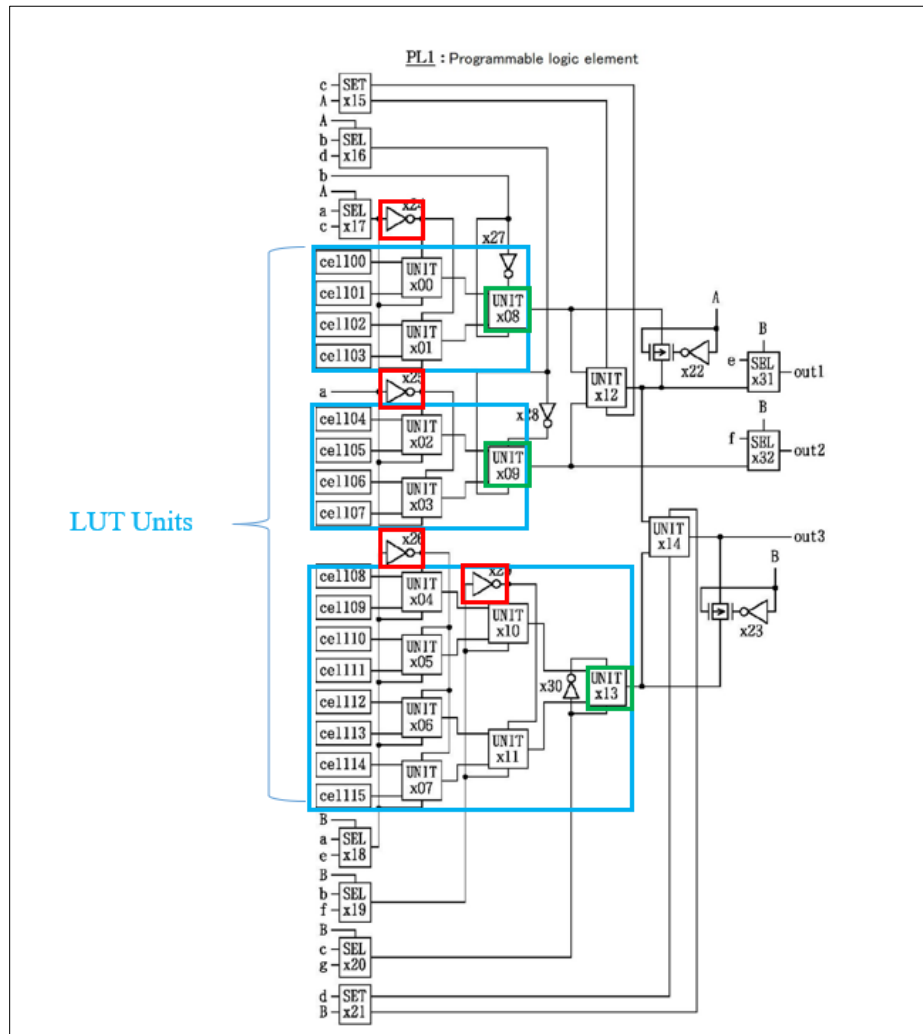
96. Even though in my opinion the phrase “cannot be fractured or divided into smaller units” is ambiguous and imprecise, I have applied an understanding, based on the Patent Owner statements, that it is intended to describe the sharing of circuitry between two smaller LUT units. On this basis, the “exemplary” “*LUT units*” in the ’737 patent’s figures and PL1’s three memory circuit groups in Fujii both have this property. For example, the LUT unit in FIG.7 uses one address decoder (7311, red) to “decode[] 2 bit input signal A, B” and “designate” two memory units, one from M11-14 and one from M21-24. ’737-patent, 6:1-4. Then, a selector (7312, green) “selects one” of those two units “to output” data (’737-patent, 6:4-6):



Similarly, the top 2-LUT “*LUT unit*” in Fujii uses a single inverter x24 (red), shared between selectors x00 and x01, in selecting two memory circuits. Fujii, ¶¶25, 60. Then, a selector x08 (green) selects between those two memory circuits to output data:



97. Fujii, FIG.1 (annotated and excerpted), ¶31. Likewise, the middle 2-LUT “*LUT unit*” and bottom 3-LUT “*LUT unit*” also use shared inverters (red) in selecting two memory circuits, which are then selected by a selector (green):



98. Fujii, FIG.1 (annotated), ¶¶24-27, 30-32, 34. Accordingly, to the extent the '737 patent's preferred embodiments of “*LUT units*” “cannot be fractured or divided into smaller units,” the same is true of Fujii’s three memory circuit groups.

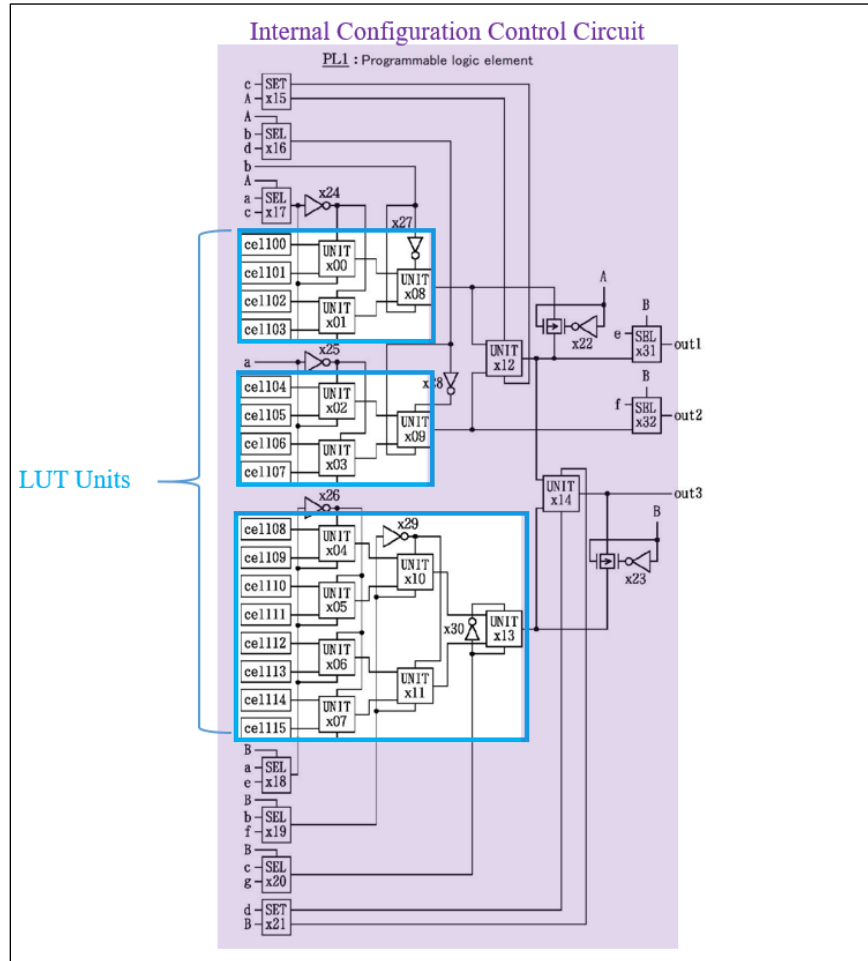
99. PL1 therefore comprises “three memory circuit groups each composed of 8, 4, and 4 memory circuits, selecting 3 input signals, 2 input signals, and 2 input signals” (Fujii, ¶12), which are “*a plurality of LUT units*” under either Petitioner’s or Patent Owner’s proposed construction of “*LUT units*[.]”

**3. [1B] “an internal configuration control circuit controlling an internal configuration of said plurality of LUT units,”**

100. Fujii discloses “*an internal configuration control circuit controlling an internal configuration of said plurality of LUT units*” because it discloses a circuit that allows programmable logic element PL1 to “switch[] between three operations” or “use mode[s].” See Fujii, ¶¶12, 14, 38, 51; see also Fujii, ¶¶54-55, 71-79, FIG.8 (substantially similar circuits and “operations” in PL2); Fujii, ¶¶80, 94-97, FIG.15 (substantially similar circuits and “operations” in PL3). This mode switching constitutes “*controlling an internal configuration of said plurality of LUT units*,” making the circuit “*an internal configuration control circuit*[.]”

101. **First**, Fujii discloses a circuit. In particular, it discloses control signals, selectors, control circuits, CMOS transfer gates, and the wires connecting those components to each other and/or other components within the programmable logic element PL1. Fujii, ¶12 (“The programmable logic element PL1 switches between the following operations by two control signals[.]”), Fujii, ¶15 (noting PL1 includes selectors x12, x14, “selectors x16 to x20, x31, x32, CMOS transfer

gates x22, x23, [and] control circuits x15, x21”). A POSA would have understood these components to constitute a circuit, as shown in purple below:



Fujii, FIG.1 (annotated).<sup>10</sup>

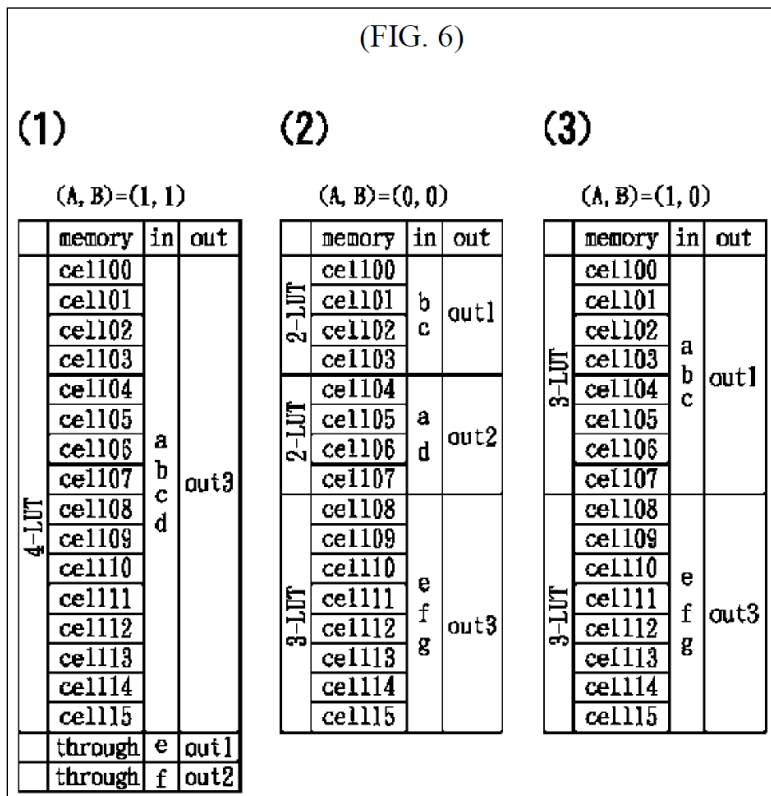
102. **Second**, this circuit represents “an internal configuration control” circuit for “controlling an internal configuration of said plurality of LUT units.”

The circuit shaded in purple above allows programmable logic device PL1 to

<sup>10</sup> Inverters x24-x30 are part of the “LUT units[.]” See *supra* [1A].

“switch[] between three operations” or “use mode[s],” as shown in FIG.6 below.

Fujii, ¶¶12, 14, 38, 51.

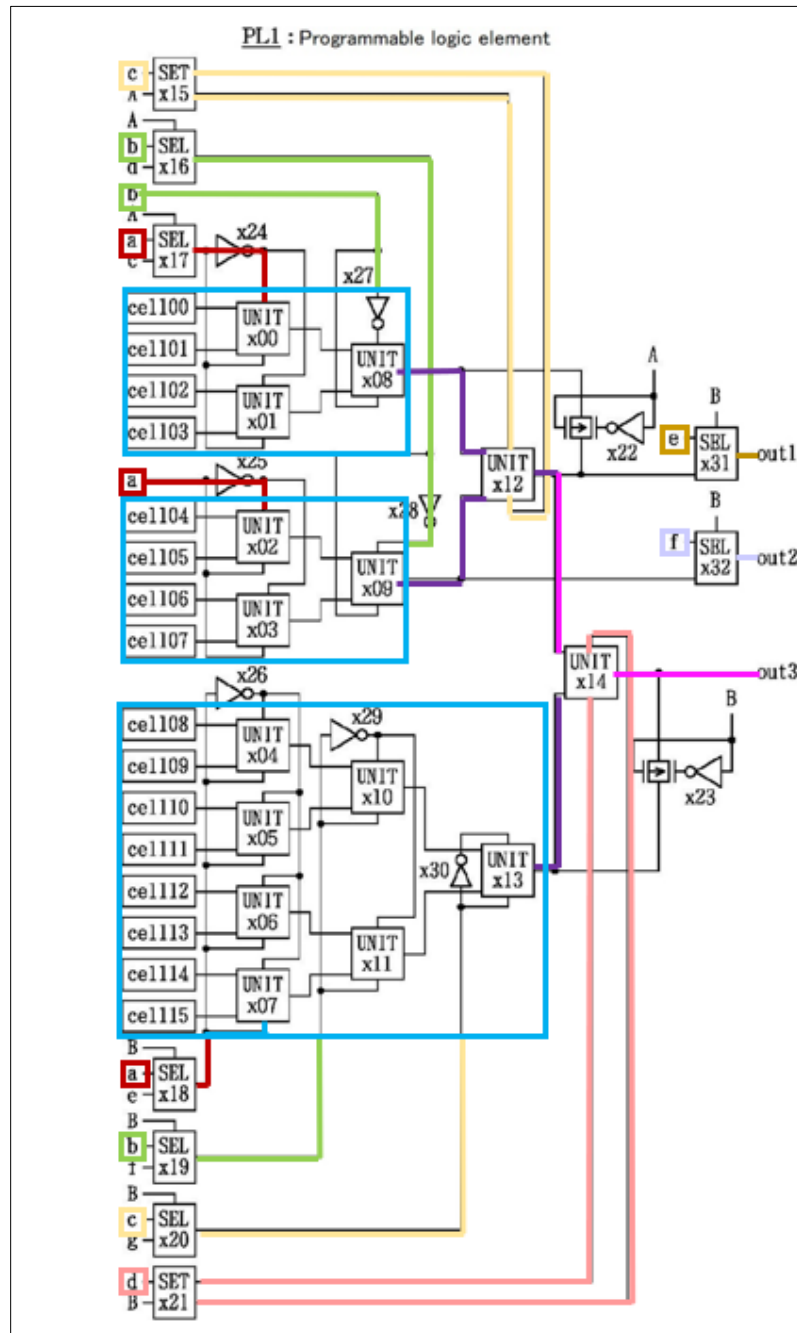


Fujii, FIG.6. In particular, programmable logic element PL1 has three use modes.

Fujii, ¶¶36-51 (describing how “PL1 is a programmable logic element that switches between three operations” or “use mode[s],” as shown in FIG.6).

103. In a **first use mode**, shown in FIG.6(1), control signals A and B are both 1, thereby setting PL1 to use mode 6(1). Based on the operation of selectors, invertors, CMOS transfer gates, and other components, the programmable logic device PL1 operates as a 4-input look-up table. Fujii ¶40, ¶¶37-41, FIGS.1, 6(1).

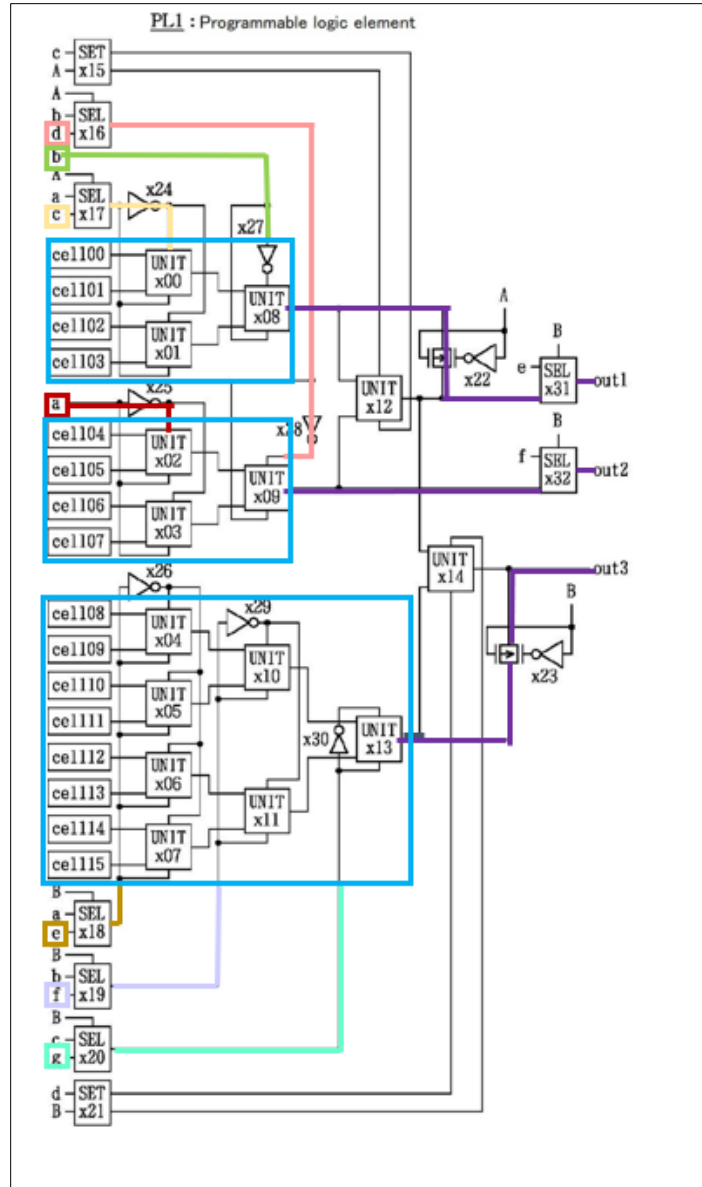
To do so, the three memory circuit groups (“*LUT units*”) are used collectively to implement a single 4-input, 1-output look-up table on out3 as shown below:



Fujii, FIG.1 (annotated). Selectors x16-x20, controlled by control signal A or B, ensure that the same input signals (a and b) are “appl[ied]” to “each memory

circuit group.” Cf. Fujii, ¶¶12, 24-28, 39, FIG.1 (annotated), see FIG.6(1). Then, in accordance with control signals A and B and input signals c and d, selectors x12 and x14 select the outputs of the “*LUT units*” (purple lines) such that “out3 corresponds to the output terminal of a four-input programmable logic element having the signals a, b, c, and d as input signals.” Fujii, ¶¶40, 12, 37-39, FIG.1 (annotated), see FIG.6(1). “[T]he input terminals e and f are connected to the output terminals out1 and out2 by the selectors x31 and x32.” Fujii, ¶41.

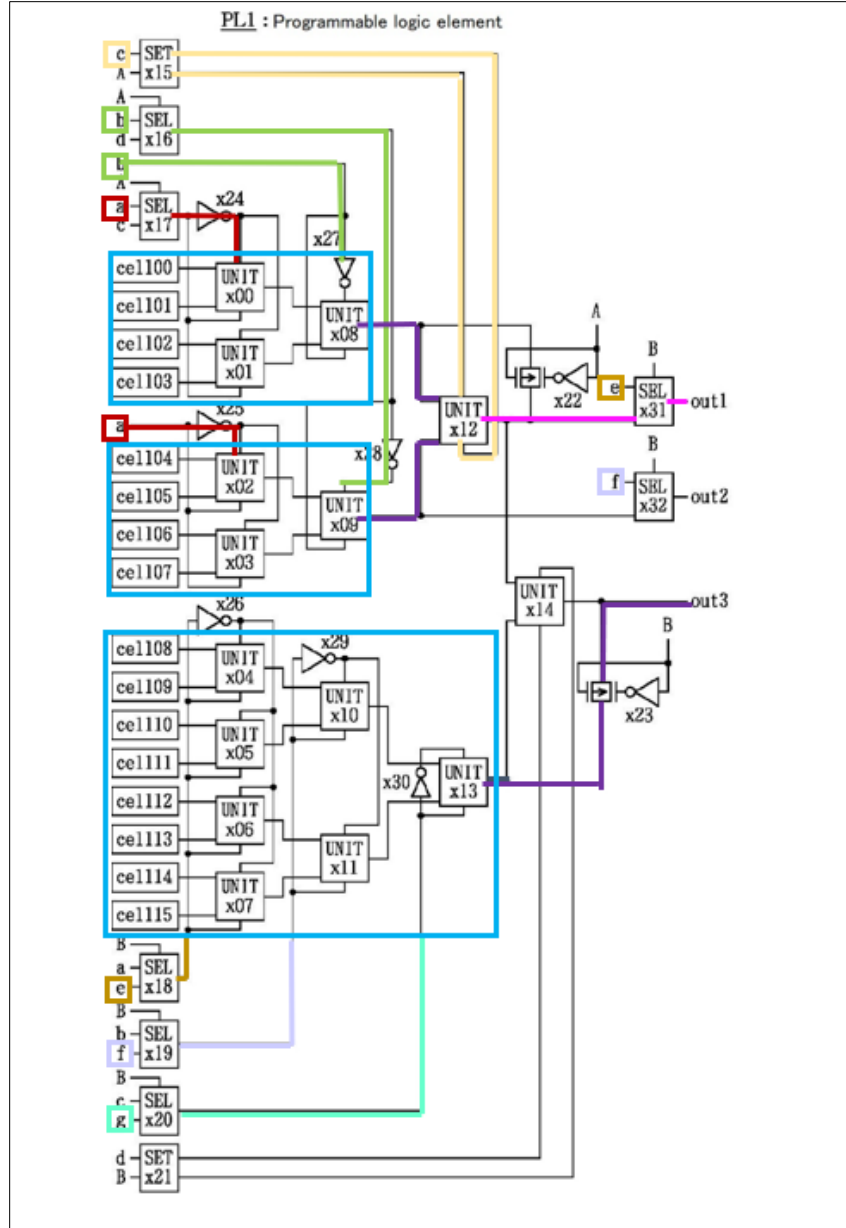
104. In a **second use mode**, shown in FIG.6(2), control signals A and B are both 0, thereby setting PL1 to use mode 6(2). Based on the operation of selectors, CMOS transfer gates, and other components, PL1 operates as two 2-input look-up tables and one 3-input look-up table. Fujii, ¶¶36-37, 45-48, FIGS.1, 6(2). To do so, the three memory circuit groups (“*LUT units*”) are used separately as shown below:



Fujii, FIG.1 (annotated). Selectors x16-x20, controlled by control signal A or B, ensure that different input signals (a-g) are “appl[ied]” to “each memory circuit group.” See Fujii, ¶¶12, 24-28, 45-46, FIG.1 (annotated), see FIG.6(2). Then, in accordance with control signals A and B, selectors x31 and x32 select the outputs of the top and middle 2-LUTs, and the output of the bottom 3-LUT is passed through CMOS transfer gate x23 (purple lines). As a result, “the output terminal

out1 corresponds to the output of a two-input programmable logic element...; the output terminal out2 corresponds to the output terminal of a two-input programmable logic element...; and the output terminal out3 corresponds to the output of a three-input programmable logic element[.]” Fujii, ¶¶48, 12, 47, FIG.1 (annotated), FIG.6(2).

105. In a **third use mode**, shown in FIG.6(3), control signal A is 1 and control signal B is 0, thereby setting PL1 to use mode 6(3). Based on the operation of selectors, CMOS transfer gates, and other components, PL1 operates as two 3-input look-up tables. Fujii, ¶¶36-37, 42-44, FIGS.1, 6(3). That is, the top and middle 2-LUT memory circuit groups (“*LUT units*”) are used to implement a single 3-LUT as shown below:



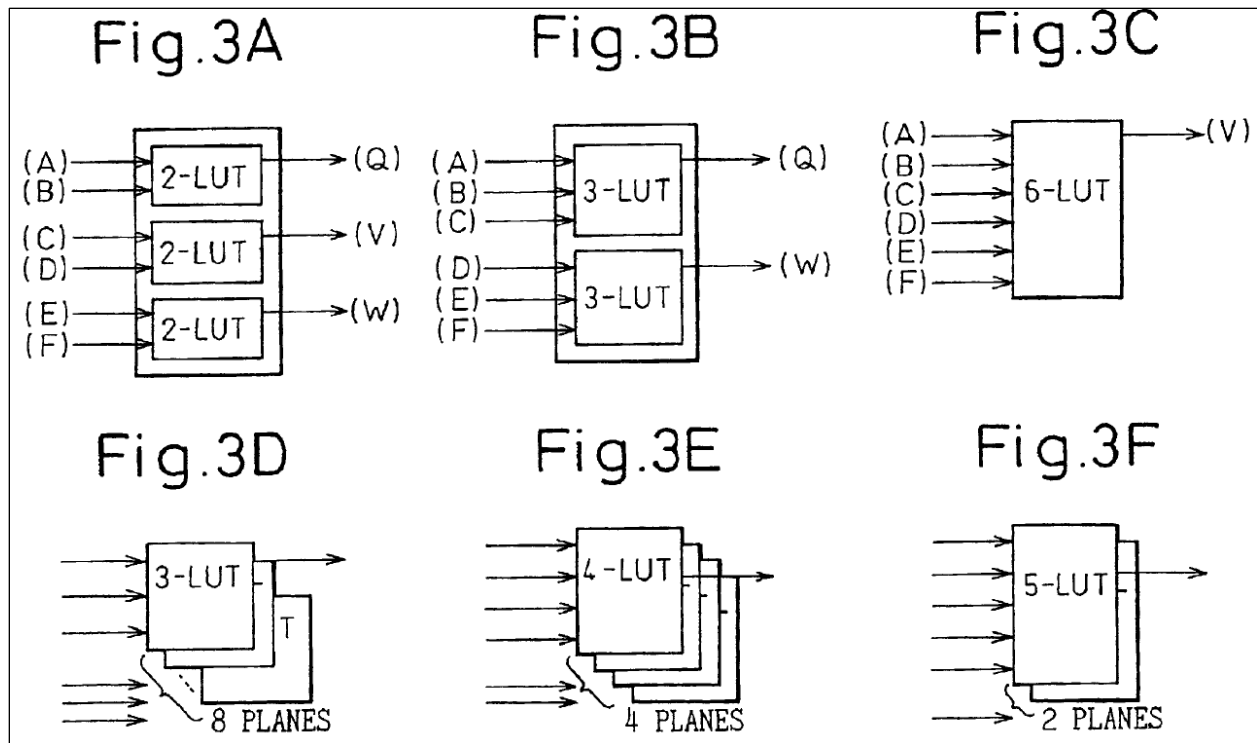
Fujii, FIG.1 (annotated). Selectors x16 and x17, controlled by control signal A, ensure that the same input signals (a-b) are “appl[ied]” to the top and middle 2-LUTs. Fujii, ¶¶12, 24-28, 42, FIG.6(3). In accordance with control signals A and B and input signal c, selector x12 selects the outputs of the top and middle 2-LUTs, and the output of the bottom 3-LUT is passed through CMOS transfer gate x23

(purple lines). As a result, “out1 acts as the output terminal of the [three]<sup>11</sup>-input programmable logic element,” Fujii, ¶43, and “out3 acts as the output terminal of the three-input programmable logic element,” Fujii, ¶44. *See also* Fujii, ¶¶12, 47, FIGS.1, 6(2).

106. This mode-switching is not materially different from the “*controlling an internal configuration of said plurality of LUT units*” described in the ’737 patent. The patent describes “internal configurations” as “configuration modes” that are switched by varying values of a memory. *See* ’737-patent, 7:17-20 (“[A]ccording to the present embodiment [shown in FIG.6], the 6-input 3-output LUT, for example, may be used in various modes of *internal configuration* by varying the values X, Y, Z of the mode changing memory.”), 6:15-19 (“The relation of the 3 bit data X, Y, Z, designated by the mode changing memory 71 in the 6-input 3-output LUT shown in FIG.6 as described above[,] and the corresponding *internal configurations (configuration modes)* . . . with reference to FIGS. 3A~3F.”). Indeed, Figures 3A through 3F show “different modes of internal configuration of [a] 6-input 3-output LUT”:

---

<sup>11</sup> “[F]our” is a typographical error that should be “three[.]” *See* Fujii, ¶43 (indicating input signals “a, b, and c” are applied to the top 3-LUT).



In the same way, Fujii's three modes are internal configurations of the “*LUT units*” that are switched by varying values of control signals as shown in FIG.6.

107. Thus, Fujii discloses an “*internal configuration control circuit* [the control signals, selectors, control circuits, CMOS transfer gates, and wires connecting those components to each other and/or other components within PL1] *controlling an internal configuration of said plurality of LUT units* [by switching between the three use modes shown in FIG.6].”

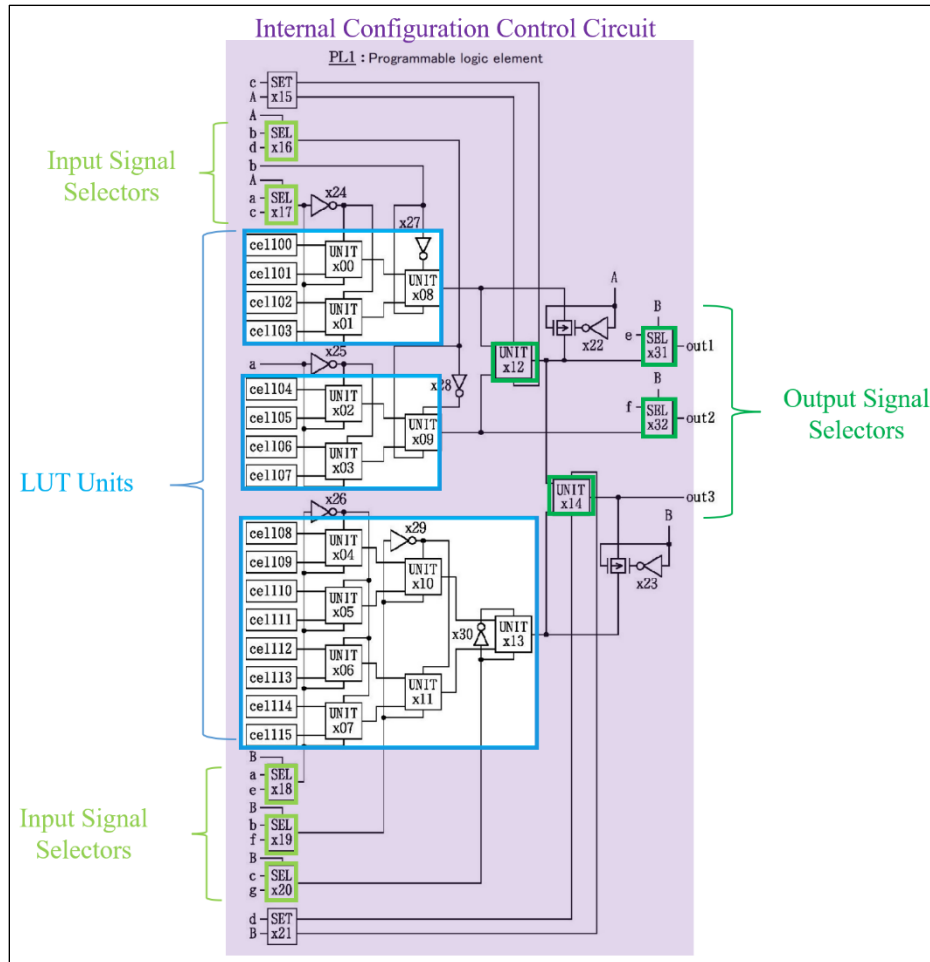
**4. [1C] “wherein said internal configuration control circuit comprises a plurality of selectors selecting I/O signals of said plurality of LUT units, and”**

108. Fujii discloses this limitation because it discloses selectors x16-x20 that select input signals of the “*plurality of LUT units*” in PL1 and selectors x12,

x14, x31, and x32 that select output signals of the “*plurality of LUT units*” in PL1.

Fujii, ¶¶59-64, 67-68, FIG.8 (substantially similar selectors in PL2); Fujii, ¶¶86-87, 90-92, FIG.15 (substantially similar selectors in PL3). These selectors are part of “*said internal configuration control circuit.*” See *supra* [1B].

109. Annotated FIG.1 below shows each of the input signal selectors (light green) and output signal selectors (green) “*selecting I/O signals of said plurality of LUT units*”:



110. On the input side, based on the value of control signal A, selector x16 selects between inputs b and d to selector x09, which is part of the middle 2-LUT. Fujii, ¶24, FIGS.1, 6(2). Similarly, based on the value of control signal A, selector x17 selects between inputs a and c to selectors x00 and x01, which are part of the top 2-LUT. Fujii, ¶25, FIGS.1, 6(2). In the same way, based on the value of control signal B, selectors x18, x19, and x20 select between inputs to the bottom 3-LUT. Fujii, ¶¶26-28, FIGS. 1, 6. The top 2-LUT, middle 2-LUT, and bottom 3-LUT are all “*LUT units*,” *see supra* [1A]. Each of these inputs is an input signal, and, thus, an “*I/O signal*.” Fujii, ¶¶37-38, FIG.6, Explanation of References.

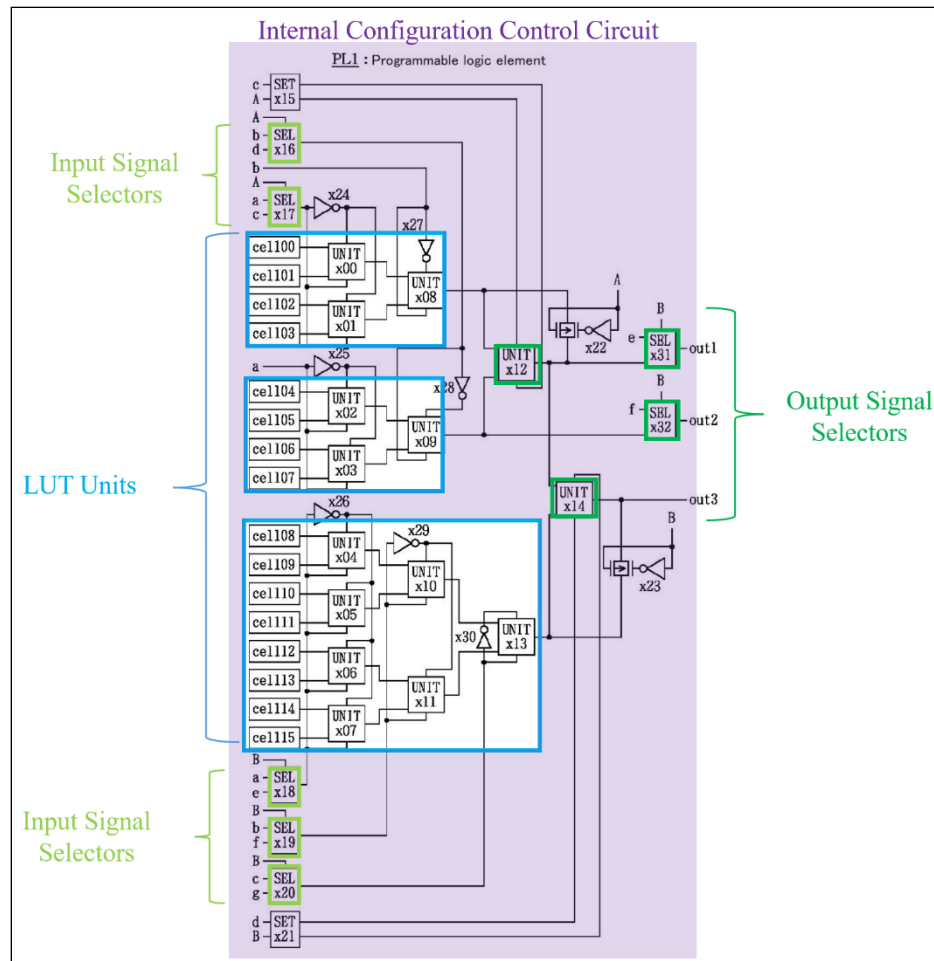
111. On the output side, when selector x12 is active, i.e., when the value of control signal A is 1, such that “selector x12” has not been rendered “non-conductive regardless of the selection signal” (Fujii, ¶19), it selects between the outputs of selectors x08 and x09, each of which is part of a “*LUT unit*” (the top 2-LUT and middle 2-LUT, respectively). Fujii, ¶¶39-40, 42-44, FIGS.1, 6(1), 6(3). Similarly, when selector x14 is active, i.e., when the value of control signal B is 1, such that “selector x14” has not been rendered “non-conductive regardless of the selection signal,” it selects between the output of selector x12 and the output of selector x13 (the output of the bottom 3-LUT), the latter of which is an output of a “*LUT unit*.” Fujii, ¶¶39-40, FIGS.1, 6(1). Also, in Mode 6(2), selector x31 selects between input signal e and the output of selector x08, the latter of which is an

output from a “*LUT unit*” (i.e., the top 2-LUT). Fujii, ¶¶45-48, FIGS.1, 6(2).

Likewise, in Mode 6(2), based on the value of control signal B, selector x32 selects between input signal f and the output from selector x09, the latter of which is an output from a “*LUT unit*” (i.e., the middle 2-LUT). Fujii, ¶¶45-48, FIGS.1, 6(2).

Each output from a “*LUT unit*” is a signal and, thus, an “*I/O signal*.”

112. Annotated FIG.1 below shows each of the input signal selectors (light green) and output signal selectors (green):



113. As described above in [1B], these selectors are part of the “*internal configuration control circuit controlling an internal configuration of said plurality of LUT units.*” See *supra* [1B]. Thus, Fujii’s “*internal configuration control circuit comprises a plurality of selectors [selectors x12, x14, x16-x20, x31, x32] selecting I/O signals [e.g., selector x16 selects between inputs b and d and selector x12 selects between outputs of the top and middle 2-LUTs] of said plurality of LUT units[.]*”

5. [1D] [wherein said internal configuration control circuit comprises . . . ] “a selector control circuit having a memory, controlling said plurality of selectors in accordance with data stored in said memory, and defining the internal configuration of said plurality of LUT units.”

114. Fujii in combination with Betz (hereinafter, “Fujii-Betz”) renders this element obvious.

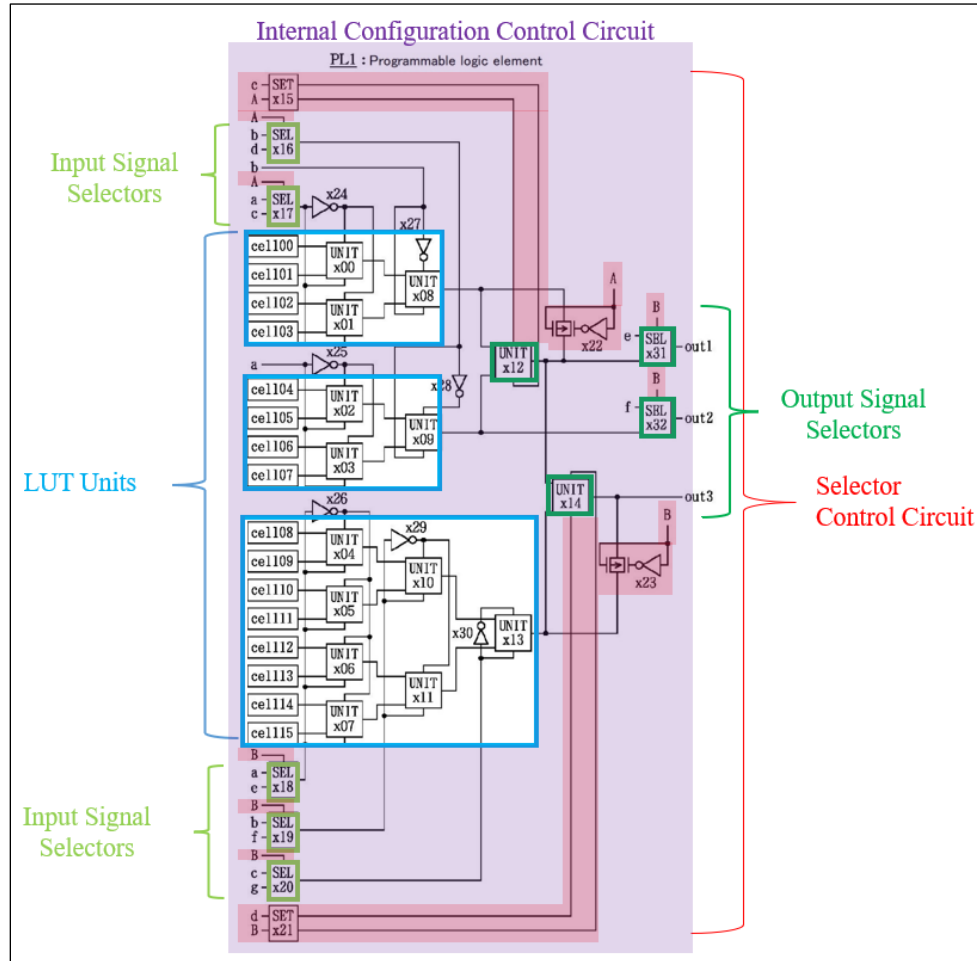
115. **First**, Fujii discloses “a selector control circuit..., controlling said plurality of selectors..., and defining the internal configuration of said plurality of LUT units” because it discloses control signals A and B, control circuits x15 and x21, CMOS transfer gates x22 and x23, and the wires connecting those control signals and components to each other and/or other components in the internal configuration control circuit, as discussed above in [1B]. Fujii, ¶¶12, 15, 36-48, 51, FIG.1; see also Fujii, ¶¶54-55, 71-79, FIG.8 (substantially similar circuits in PL2); Fujii, ¶¶80, 94-97, FIG.15 (substantially similar circuits in PL3). As discussed

above, these components in the “selector control circuit” are part of “said internal configuration control circuit” (see *supra* [1B]) that “define[s] an internal configuration of said plurality of LUT units,” because these components (as part of the internal configuration control circuit) define PL1’s use modes.

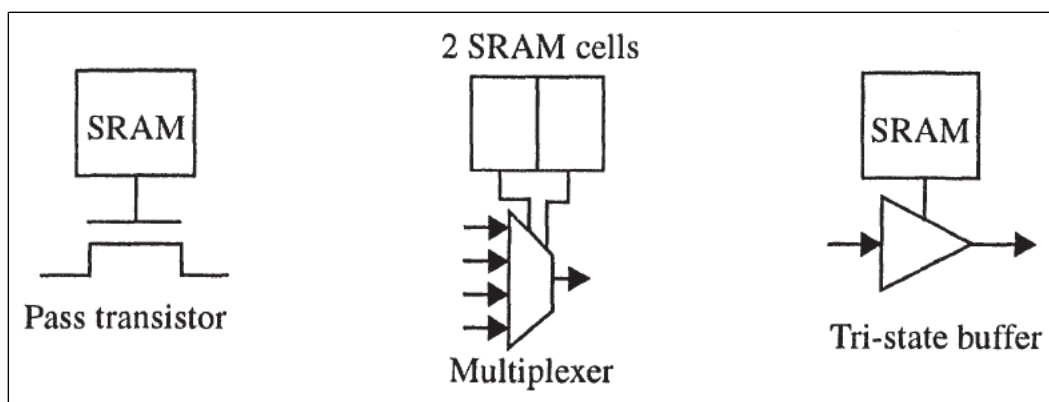
116. Control signals A and B, in combination with the other circuitry listed above, define the use modes of the programmable logic element PL1 by controlling selectors. *See supra* [1B], [1C]. For example, depending on the values of control signals A and B, control circuits x15 and x21 in conjunction with transfer gates x22 and x23 determine whether selector x12 and selector x14 operate as part of selection circuits and are connected to outputs (out1 and out3, respectively) and select outputs of the LUT units. Fujii, ¶¶19, 30, 34-35, 39, 42, 45, 47, FIGS.1, 6. Moreover, depending on the values for control signals A and B, selectors x16 through x20 select inputs to the “*LUT units*” and selectors x31 and x32 select outputs from the “*LUT units*.” *See supra* [1C].

117. These components form “*a selector control circuit*” that controls “*said plurality of selectors*” and “*defin[es] the internal configuration of said plurality of LUT units*,” because they establish what use modes PL1 operates in, and those modes are controlled by selectors that are controlled by the values of control signals A and B. *See supra* [1B] (describing how PL1 switches use modes using these components), [1C] (describing control of selectors). Annotated FIG.1

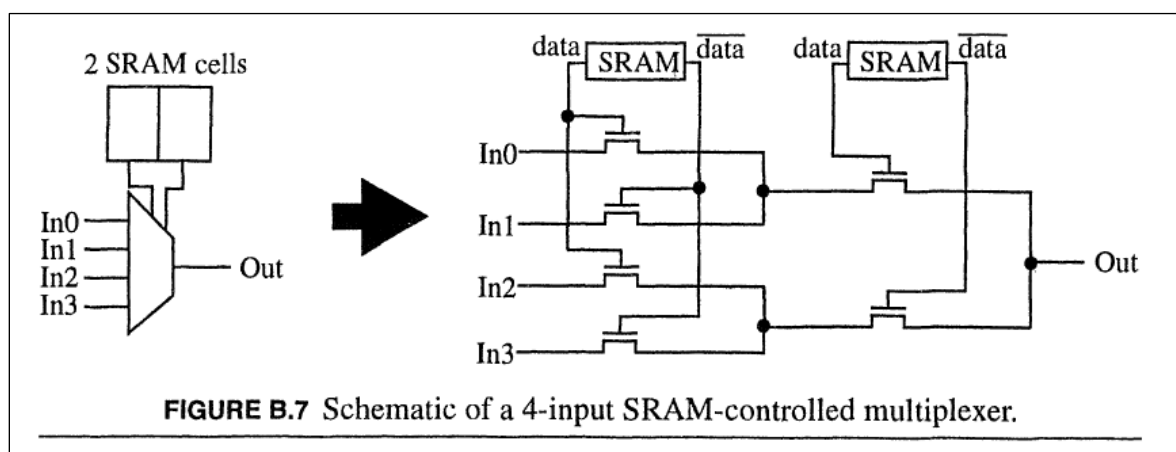
below shows the “*selector control circuit*” (in red) of programmable logic element PL1:



118. **Second**, Betz discloses “*having a memory*” and “*controlling [a] plurality of selectors in accordance with data stored in said memory[.]*” In particular, Betz’s FIG.2.2 shows three types of switches controlled in accordance with data stored in memory, e.g., controlling a 4-to-1 multiplexer in accordance with data stored in 2 SRAM cells:



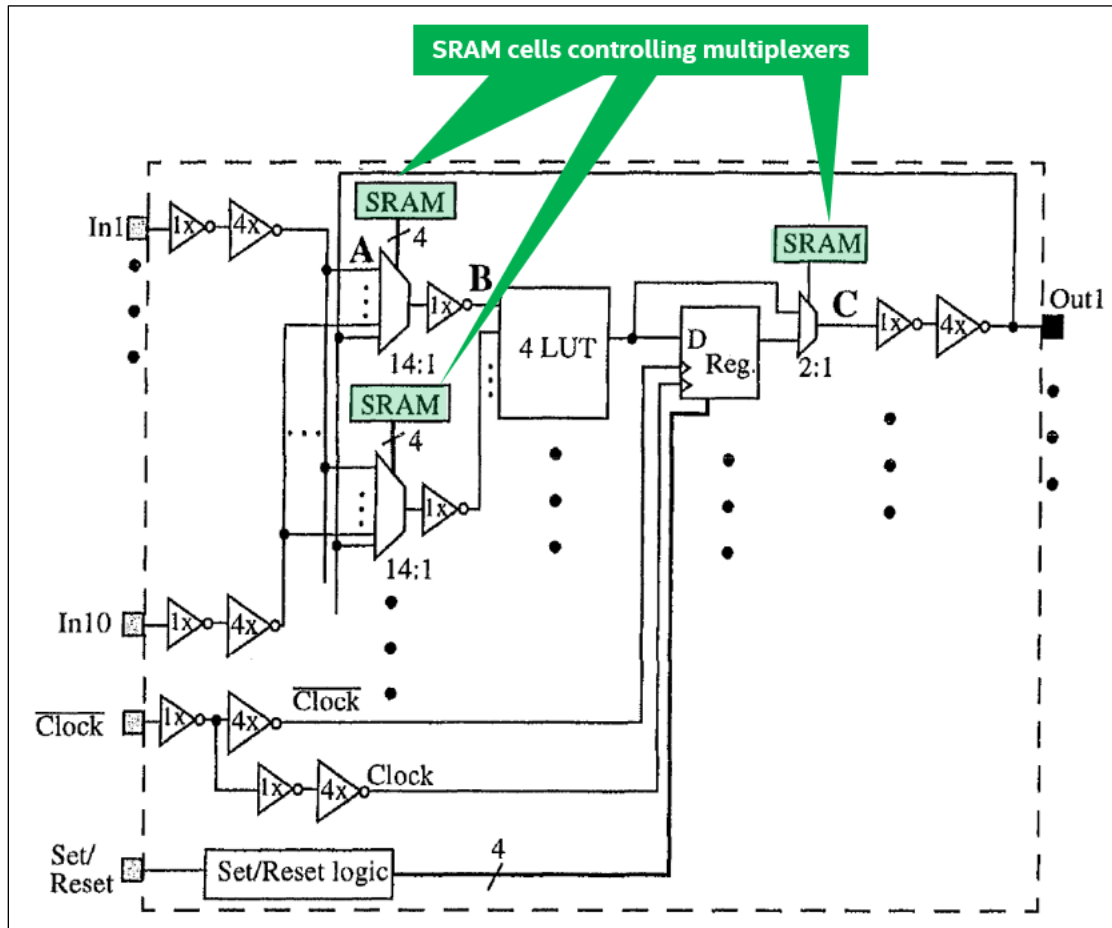
See also Betz, 12, 209, FIG.B.2 (showing pass transistor); Trimberger, 19-21 & FIG.2.3.2.c (showing a 2-to-1 multiplexer controlled by configuration memory in SRAM-based architecture); Brown-FPGA pp. 15-16, Fig. 2.2 (showing SRAM-based transmission gate and multiplexer). Betz's FIG.B7 shows that each SRAM cell contains "data" in memory that is used to control the switches:



See also Betz, 208, 213, FIG.B.1.

119. Betz also teaches that it was known to implement SRAM-based switches to control selection of inputs and outputs of look-up tables within a logic block. Betz, 214, FIG.B.8. For example, FIG.B.8 has a "logic block" with four "4-

input look-up table[s]“ (one labeled “4 LUT” below) and sixteen 14-to-1 multiplexers at the inputs of the 4-LUTs and four 2-to-1 multiplexers at the outputs of each “4-LUT.” Betz, 214-216.



Betz, 214, FIG.B.8 (annotated). Each SRAM cell (green) to the left of the illustrated “4 LUT” uses 4 data bits to control a 14-to-1 multiplexer, which selects inputs to the “4 LUT.” Betz, 214, FIG.B.8. The SRAM cell (green) to the right of the illustrated “4 LUT” uses 1 data bit to control a 2-to-1 multiplexer, which selects an output from the “4 LUT.” Betz, 214, FIG.B.8. Each SRAM cell is located within Betz’s logic block. Betz, FIG.B.8.

120. Betz therefore teaches “*having a memory*” [SRAM cells within the logic block] and “*controlling [a] plurality of selectors* [the 14-to-1 multiplexers at the inputs to the “4 LUTs” and the 2-to-1 multiplexers at the output of the “4 LUTs”] *in accordance with data stored in said memory* [the data stored in each SRAM cell].”

121. A POSA would have been motivated to modify Fujii’s “selector control circuit” that controls a “plurality of selectors” and “defin[es] the internal configuration of said plurality of LUT units” to “control said plurality of selectors in accordance with data stored in said memory” as taught in Betz. Indeed, Betz provides express motivation to modify Fujii by teaching that the use of SRAM cells to control programmable switches is “the most popular” of the “three different approaches to making FPGAs programmable[.]” Betz, 11-13; see also Trimberger, 15-21 (“SRAM-programmable FPGAs have become very popular.”); Brown-FPGA, 15-16 & FIG.2.2, 21 (“[S]tatic RAM programming technology [was] used in FPGAs produced by several companies” as of 1992.).

122. Moreover, Fujii does not specify implementation details regarding the source of control signals A and B, which are used to control the “*plurality of selectors*,” *see supra* [1B], [1C]. Thus, a POSA would have been motivated to consult other references, including Betz, to fill in those details. In particular, a POSA would have consulted Betz because it “provides background information

about FPGA architecture” (Betz, 11), and PL1 is used in FPGA architecture (Fujii, ¶¶3, 5, 6, 102). Moreover, Betz provides implementation details for SRAM-controlled switches, and control signals A and B are used to control similar switches, i.e., selectors (multiplexers), transfer gates (switches), and control circuits (switches). *Compare* Fujii, FIG.2, ¶16 (“FIG. 2 is a circuit diagram showing a concrete example of selector x16[.]”) *with* Weste, 17, FIG.1.10(c) (showing same CMOS 2-input “multiplexer”); *compare* Fujii, FIG.1 (transfer gates x22, x23) *with* Weste, 8, FIG.1.3 (same complementary “switch”); *compare* Fujii, FIG.3, ¶18 (control circuit x15) *with* Weste, 14-15 (“compound gate is formed by using...switch structures”). Thus, a POSA would have consulted Betz for details about the source of control signals A and B, i.e., that one or both of those signals could be stored in data in SRAM cells.

123. Separately, there were well-known advantages in using SRAM technology because “it provides an FPGA that can be re-configured (in-circuit) very quickly and it can be produced using a standard CMOS process technology.” Brown-FPGA, 16. This would have also motivated the use of SRAM technology in the manner described above.

124. These reasons, separately and collectively, would have motivated a POSA to modify Fujii in light of the teachings in Betz.

125. Moreover, a POSA would have had a reasonable expectation of success in modifying Fujii in this way because, as discussed above, Betz teaches the use of SRAM-based switches to control selection of inputs and outputs of look-up tables within a logic block similar to Fujii's PL1, which already uses switches in that same way. Betz, 214, FIG.B.8. Indeed, Betz teaches that SRAM-controlled switches were "the most popular technology" at the time for FPGAs. Betz, 11-12. Also, PL1 uses CMOS process technology (Fujii, ¶¶15, 20, 22, 33-35, 39, 42, 47, p.10 (describing FIG.4)), and SRAM programming technology "can be produced using a standard CMOS process technology" (Brown-FPGA, 16).

126. Modifying Fujii in this way amounts to use of a known technique (Betz's SRAM-based switches to control selection of inputs and outputs of look-up tables) to improve a similar device (PL1, which addresses the same problem of controlling selection of inputs and outputs of look-up tables) in the same way ("provid[ing] an FPGA that can be re-configured (in-circuit) very quickly," *see* Brown-FPGA, 16).

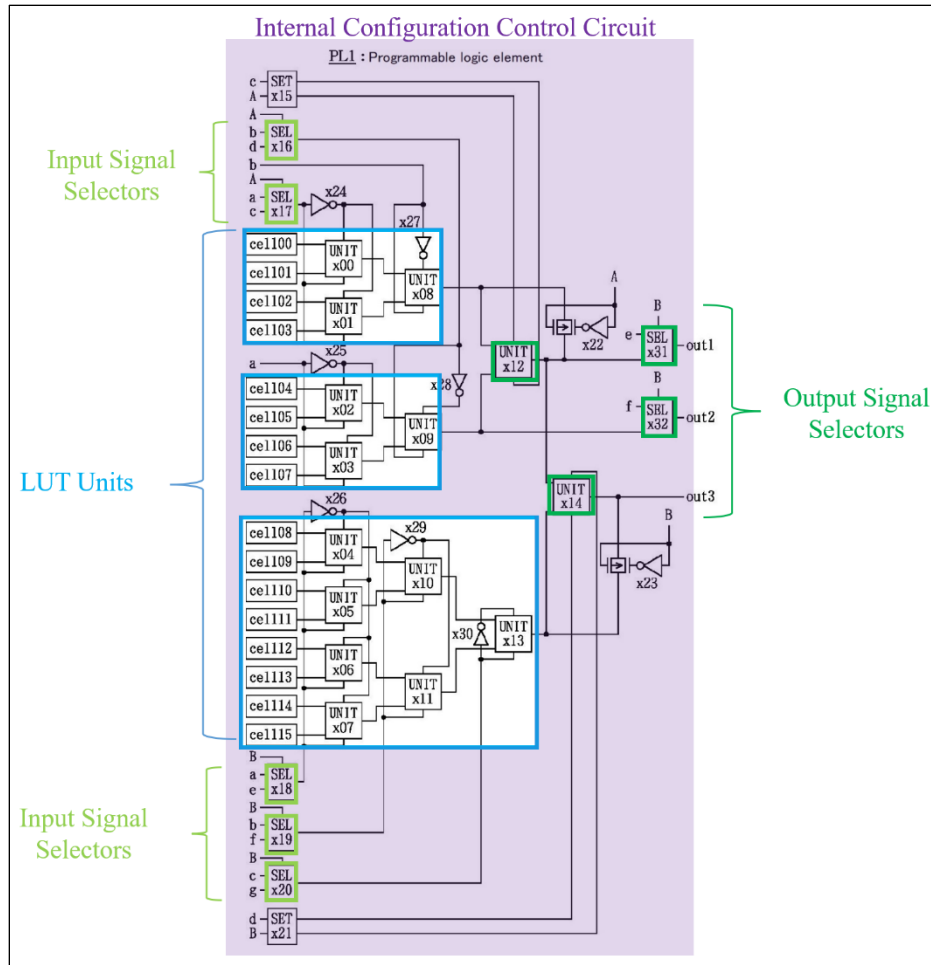
127. Fujii-Betz therefore teaches this element because, when the values of one or both of control signals A and B in Fujii's internal configuration control circuit (*see* [1B], [1C]) are stored in SRAM cells as taught in Betz, PL1 comprises "*a selector control circuit having a memory, controlling said plurality of selectors*

*in accordance with data stored in said memory, and defining the internal configuration of said plurality of LUT units.”*

**C. Claim 2**

- 1. [2A] “The look up table as claimed in claim 1, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and”**

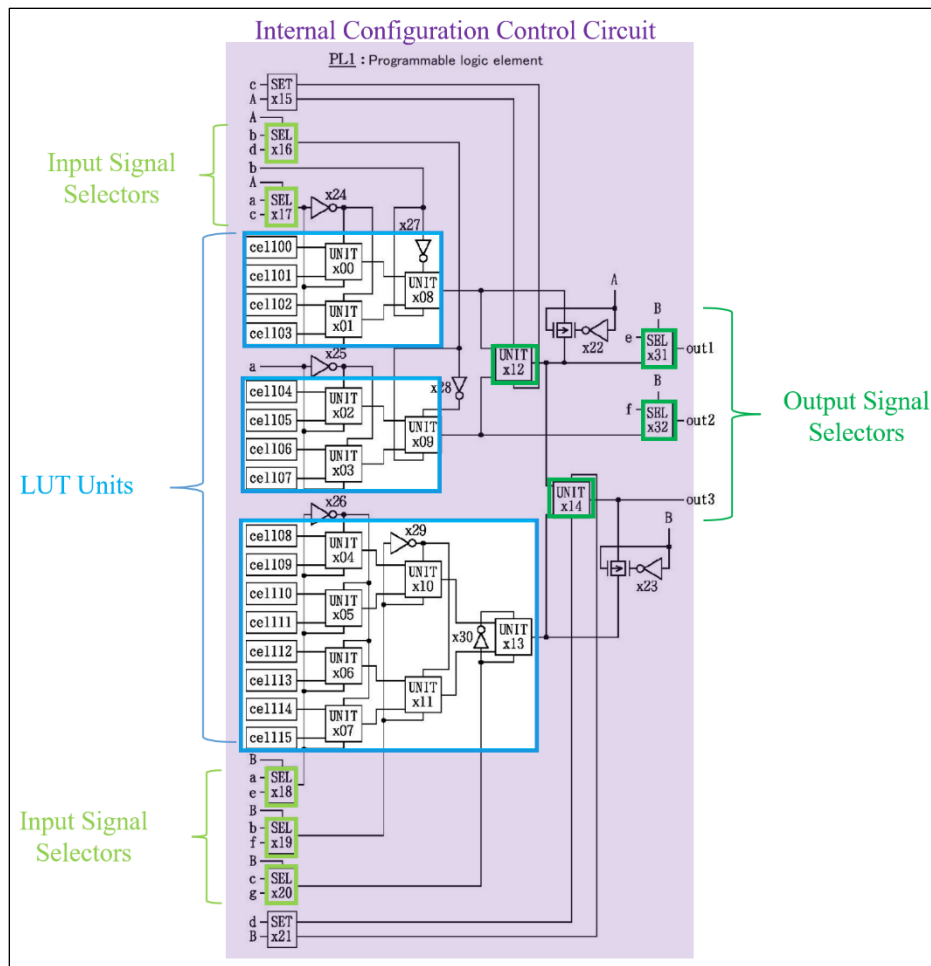
128. Fujii discloses this limitation because, as discussed above ([1C]) and shown in FIG.1 below, selectors x16-x20 are each provided at an input of at least one of the “*LUT units*” in PL1 to select an input signal. Annotated FIG.1 below shows each “*input signal selector*” (light green):



See also Fujii, FIGS.8 (substantially similar for PL2), 15 (substantially similar for PL3).

2. [2B] [wherein said plurality of selectors include: . . .] “an output signal selector provided at an output of said LUT units selecting an output signal,”

129. Fujii discloses this limitation because, as discussed above ([1C]) and shown in annotated FIG.1 below, selectors x12, x14, x31, and x32 are each provided at an output of the “LUT units” in PL1 to select an output signal. Annotated FIG.1 below shows each “output signal selector” (green):



See also Fujii, FIGS.8 (substantially similar for PL2), 15 (substantially similar for PL3).

3. [2C] “said input signal selector and said output signal selector being controlled in accordance with the data stored in said memory.”

130. Fujii-Betz renders this limitation obvious. As discussed above ([1D]), Betz teaches controlling input selectors and output selectors with data stored in memory composed of SRAM cells. As explained ([1D]), a POSA would have found it obvious to modify Fujii to store control signal A, control signal B, or both

as data in memory located within PL1 to control selectors, including an input signal selector and output signal selector as identified in [2A]-[2B]. Therefore, Fujii-Betz discloses “*said input signal selector and said output signal selector being controlled in accordance with the data stored in said memory.*” See also *supra* [1D] (substantially similar for PL2 and PL3).

#### **D. Claim 3**

1. **[3A] “The look up table as claimed in claim 1, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and”**

131. Fujii discloses this limitation. As discussed above ([2A]), selectors x16-x20 in PL1 are each “*an input signal selector provided at an input of at least one of said LUT units to select an input signal.*” See also *supra* [2A] (substantially similar for PL2 and PL3).

2. **[3B] [wherein said plurality of selectors include: . . .] “an output signal selector provided at an output of said LUT units selecting an output signal,”**

132. Fujii discloses this limitation. As discussed above ([2B]), selectors x12, x14, x31, and x32 are each “*an output signal selector provided at an output of said LUT units selecting an output signal.*” See also *supra* [2B] (substantially similar for PL2 and PL3).

**3. [3C] “said input signal selector and said output signal selector being controlled in accordance with the input signal.”**

133. Fujii-Betz renders this limitation obvious. Fujii discloses controlling output signal selectors x12 and x14 in accordance with input signals c and d, respectively. *See, e.g.*, Fujii, ¶¶18-19, 29-30, 39, 42, p.10 (labelling “a, b, c, d, e, f, and g...input signals”). Input signal c and control signal A are input into control circuit x15, which determines whether selector x12 is rendered non-conductive or is used as a selection circuit. Fujii, ¶¶15, 18-19, 29, 39, 42, 47. Similarly, input signal d and control signal B are input into control circuit x21, which determines whether selector x14 is rendered non-conductive or is used as a selection circuit. Fujii, ¶¶15, 18-19, 30, 39, 42, 47.

134. It would have been obvious to modify Fujii’s PL1 to use one of input signals a-g to control one or more input signal selectors discussed above. *See supra* [2A], [3A]. This would have been a simple substitution of one known element (one of Fujii’s input signals) for another known element (control signal A or B), each of which controls both input and output selectors, as discussed above ([1B], [1C], [2A]-[2B]) to obtain predictable results (the use of a signal entering PL1 to control input selectors). Moreover, a skilled artisan would have had a reasonable expectation of success in making this modification given the disclosures in Fujii

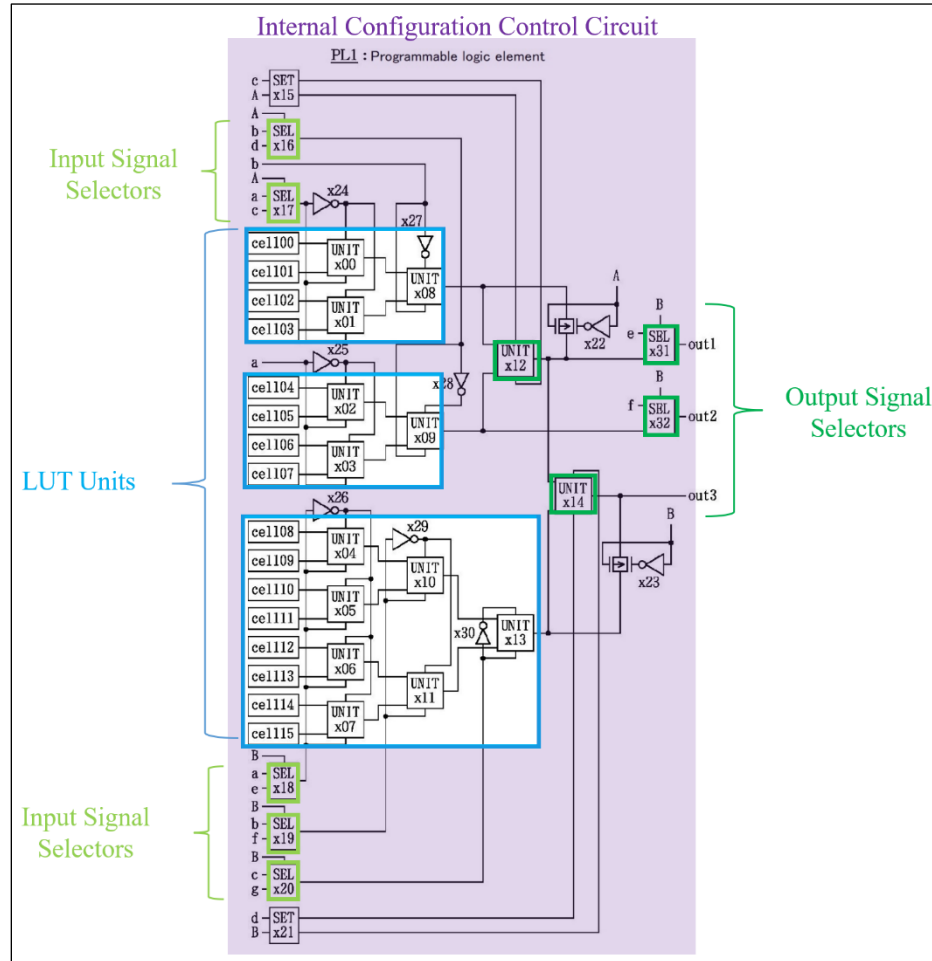
and Betz that, in part, show control of output signal selectors in accordance with input signals.

135. Thus, Fujii-Betz renders obvious “*said input signal selector* [input selectors x16-x20] *and said output signal selector* [output selectors x12 and x14] *being controlled in accordance with the input signal* [e.g., input signal c or d].”

136. Alternatively, control signals A and B are “*input signals*” that control input and output selectors as discussed above. *See supra* [1B]-[1C], [2A]-[2B]. Fujii discloses that PL1 “*inputs* at least one control signal” (Fujii, ¶14; *see also* Fujii, claim 1, ¶12), making those control signals “input signals.” Notably, this interpretation of “*input signals*” to include control signals is consistent with the disclosed embodiments of the ’737 patent, which show input selectors that are controlled exclusively by control signals (selectors 721-725 being controlled by signals X, Y and Z in FIG.6 and selectors 821-825 being controlled by signals X, Y and Z in FIG. 9).

137. Further, as discussed above ([2A]-[2C], [3A]-[3B]), and as shown in annotated FIG.1 below, control signal A controls input signal selectors x16-x17 and output signal selector x12 (via control circuit x15), while control signal B controls input signal selectors x18-x20 and output signal selectors x14 (via control circuit x21) and x31-x32. Thus, Fujii discloses “*said input signal selector* [any one of selectors x16-x20] *and said output signal selector* [any one of selectors x12,

x14, x31, and x32] *being controlled in accordance with the input signal* [control signals A or B or input signals c or d].”



Fujii, FIG.1 (annotated). Further, as discussed above ([1D]), when the value of exactly one of control signals A or B is set using data stored in SRAM cells (*see* Betz), the selectors controlled by that control signal (A or B) also are controlled “*in accordance with data stored in said memory,*” thereby satisfying the limitations in both [1D] and [3C]. *See supra* [1D].

138. Thus, Fujii teaches “said input signal selector and said output signal selector being controlled in accordance with the input signal.”

**E. Claim 4: “The look up table as claimed in claim 1, wherein said look up table of M inputs and N outputs is a 6-input 3-output look up table.”**

139. This limitation would have been obvious in light of the teachings of Fujii. Based on Fujii’s express teachings, it would have been obvious to modify PL1 to constitute a “*6-input 3-output look up table*.”

140. Fujii’s teachings expressly apply to a programmable logic element with a variable number of  $K$  input signals:

That is, the programmable logic element PL1 wherein more than  $K$  *input signals* are input, [the element] *is provided with  $2^K$  memory circuits*, outputs at least one output signal, inputs at least one control signal, and has at least one selection circuit, characterized in that the following operations are switched by the above control signal:

(i) a first operation of *using the  $K$  input signals* as the selection signal of the selection circuit, *selecting one output signal from the output signals from  $2^K$  memory circuits*, and outputting the selected output signal as an output signal of the programmable logic element; and

(ii) a second operation of selecting *one output signal from the output signals of the  $2^{X_i}$  memory circuits and outputting the selected output signal* as an output signal of the programmable logic element, by dividing the  $2^K$  memory circuits into plural memory circuit groups,

where the sequence from  $i=1$  to  $i=n$  of the sequence  $\{2^{X_i}\}$  ( $2^{X_i} \leq 2^{X_{i+1}}$ ) is equal to  $2^{[K]}$ , the memory circuit group is composed of  $2^{X_i}$  ( $i=1, 2, \dots, n$ ) *memory circuits*, and  $X_i$  input signals that do not overlap each other among the input signals are used as selection signals.

Fujii, ¶14 (brackets in original), ¶¶5-6, claim 1. In one operation mode, PL1 can implement an arbitrary logic circuit of  $K$  inputs. Fujii, ¶14. Fujii also teaches that the number of  $K$  inputs “is often 3-5” (Fujii, ¶5), necessarily contemplating  $K=6$  as a less often number only one away from those “often” used.

141. Given these teachings, a POSA would have been motivated to modify PL1 to be capable of implementing an arbitrary logic circuit of six inputs ( $K=6$  instead of  $K=4$ ) by increasing the number of memory circuits in PL1 to  $2^6$  (64) memory circuits (instead of  $2^4$ ). It was well known that wider look-up tables (i.e., look-up tables capable of implementing an arbitrary logic circuit of more inputs) provided advantages over smaller look-up tables for certain applications. *See, e.g.*, Trimberger, 66 (explaining that, compared to “[n]arrower” 4-input look-up tables, “wider lookup tables would allow faster functions of more variables”). A 6-input look-up table offers speed advantages. He, Abstract (noting 5-, 6-, and 7-input look-up tables are “superior in terms of system speed” compared to 4-LUTs); Brown-FPGA, 108-09 (“As  $D_R$  [i.e., the total routing delay] increases, the cost in delay of each logic block level begins to dominate, and so the blocks with lower values of  $N_L$  [i.e., the number of logic blocks in the critical path] achieve superior

performance. For  $D_R = 2$  the lowest delay is reached at about the 5- or 6-input cases. For  $D_R = 4$ , the best performance is achieved at about 6 or 7 inputs.”); Singh at 285 (“[T]he five- and six-input lookup tables are good choices for  $DR = 2$  and  $DR = 4$  ns, which are realistic values for routing delay.”). To take advantage of these speed benefits, a POSA would have been motivated to modify PL1 to be capable of implementing an arbitrary logic circuit of six inputs (i.e.,  $K=6$ ). This would also be a simple substitution ( $K=6$  for  $K=4$ ) of known elements (the number of input signals and the corresponding number of memory circuits) to obtain predictable results (PL1 capable of implementing an arbitrary logic circuit of 6 inputs). A POSA would have had a reasonable expectation of success in achieving this modification because, as discussed above, Fujii expressly teaches that PL1 may have any number of input signals (more than  $K$ ) and a corresponding number of memory circuits ( $2^K$ ).

142. Moreover, this modified version of PL1 would have exactly three outputs. The disclosed embodiment of PL1 has three output terminals (out1, out2, out3), *see* Fujii ¶¶34-35, 37-38, 40-48, FIGS.1, 6(1)-6(2), and modifying that disclosed embodiment of PL1 to have  $K=6$  (64 memory circuits) without changing the number of outputs would have been routine. Fujii itself shows various programmable logic elements (PL2 and PL3) with different values for  $K$  (4 and 3,

respectively) and the same number of outputs (2). Fujii ¶¶53-55, 79-81, FIGS.8, 13, 15, 20.

143. Additionally, Fujii teaches that PL1 with any number of input signals (more than K) still contains “plural memory circuit groups,” which are “*LUT units*” (see *supra* [1A]). Fujii, ¶14. It also “inputs at least one control signal” and has operations (i.e., modes) that “are switched by [that] control signal[,]” Fujii, ¶14, thereby “*controlling an internal configuration of said plurality of LUT units*” as described above ([1B]). This requires selecting different outputs from the “*LUT units*” to be “an output signal of the programmable logic element,” see Fujii, ¶14(i)-(ii), which requires at least one output signal selector as described above ([1B]-[1D], [2B]). Because PL1 inputs more than K inputs and has “a first operation using the K input signals as the selection signal of [a] selection circuit,” PL1 must have at least one input signal selector to ensure K inputs enter the selection circuit as described above ([1B]-[1D], [2A]). Fujii, ¶14.

144. Fujii therefore renders obvious “said look up table [PL1] of M inputs [K=6] and N outputs [out1, out2, out3] is a 6-input 3-output look up table.”

145. Alternatively, to the extent the claim is construed to mean the “*look up table of M inputs and N outputs*” requires only a minimum of M inputs and N outputs and need not be capable of implementing an arbitrary logic circuit of its inputs with at least one output, PL1 is a look-up table of at least 6 inputs (a-g) and

three outputs (out1, out2, out3). Fujii, ¶¶35-41, FIGS.1, 6(2). Under this alternative construction, PL1 remains “[t]he look up table as claimed in claim 1, wherein said look up table of M inputs and N outputs is a 6-input 3-output look up table,” as recited by claim 4.

**F. Claim 5: “The look up table as claimed in claim 4, wherein said 6-input 3-output look up table comprises eight 3-input 1-output LUT units.”**

146. As described above, Fujii-Betz renders obvious “[t]he look up table as claimed in claim 4.” See *supra* [4]. Claim 5’s additional limitations would have been obvious in light of Fujii. Based on Fujii’s express disclosures, it would have been obvious to implement the 6-input 3-output PL1 discussed above ([4]) using “eight 3-input 1-output LUT units.”

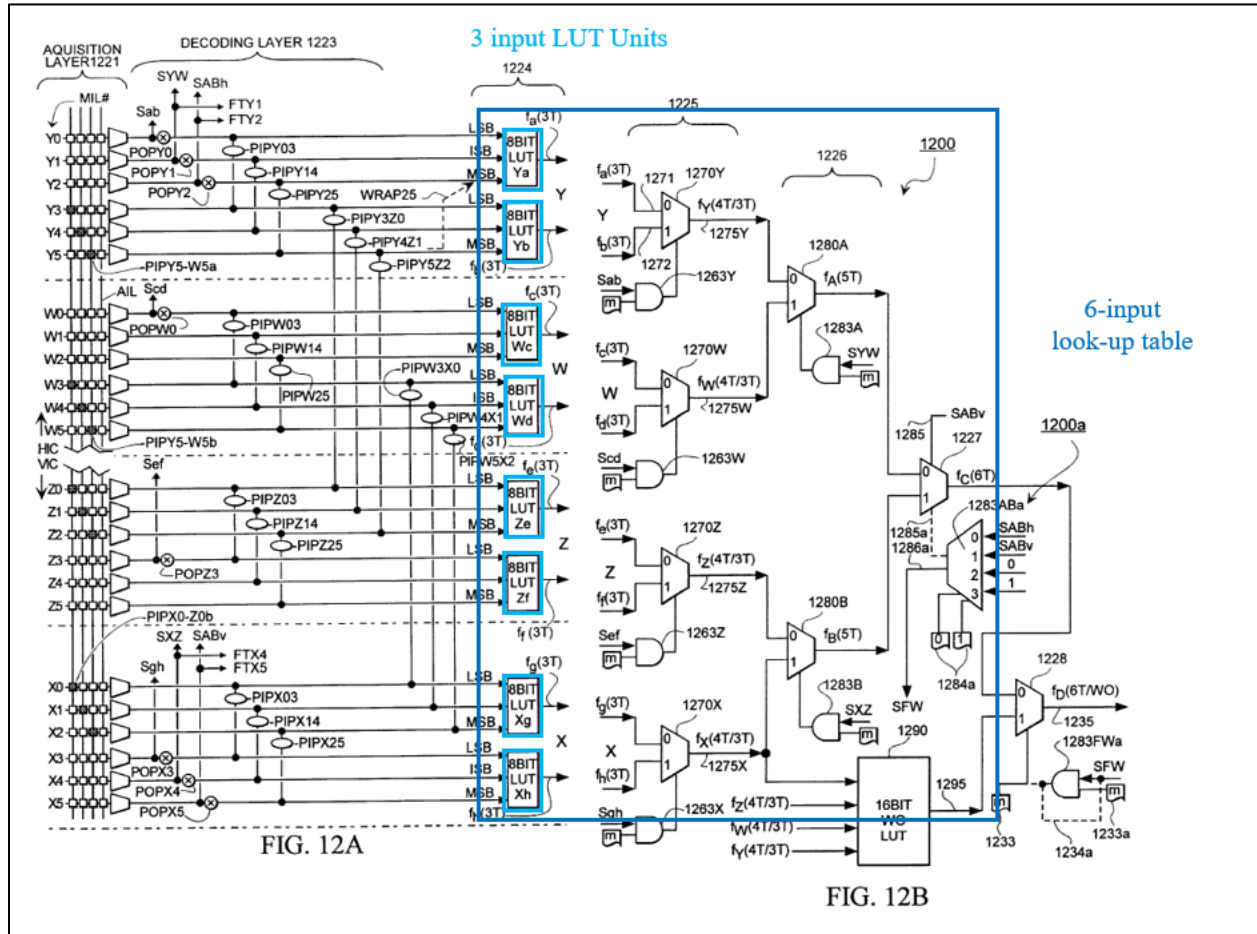
147. As noted (*supra* [4]), Fujii teaches that PL1 may have any number of input signals, i.e., more than “K input signals,” and a corresponding number of memory circuits, i.e., “ $2^K$  memory circuits.” Fujii, ¶¶5, 14, claim 1. Fujii further teaches that those “ $2^K$  memory circuits” may be “divided...into plural memory circuit groups” in various ways with each memory circuit group having “ $2^{X_i}$  memory circuits,” where “X” is the number of input signals to a particular memory circuit group and “i” is an index, ranging from 1 to “n” (the total number of memory circuit groups), identifying a particular memory circuit group. Fujii, ¶14. Fujii expressly contemplates that each memory circuit group may have the same

number of input signals ( $X_i$ ) and, thus, the same number of memory circuits. Fujii, ¶14 (“where the sequence from  $i=1$  to  $i=n$  of the sequence  $\{2^{X_i}\}$  ( $2^{X_i} \leq 2^{X_{i+1}}$ ) is equal to  $2^{[K]}$ ,” meaning “ $2^{X_i}$ ” may be equal to “ $2^{X_{i+1}}$ ”). Indeed, Fujii’s programmable logic element PL2 uses two 3-input, 1-output memory circuit groups. Fujii, ¶100 (“[T]he sequence  $\{2^{X_i}\}$  in this case is  $\{2^3, 2^3\}$ .”), FIGS.8, 13. As explained, the plurality of memory circuit groups, each having  $X_i$  input signals and  $2^{X_i}$  memory circuits, is a “*plurality of LUT units.*” *See supra* [1A], [4].

148. Given this teaching, a POSA would have been motivated to implement the modified PL1 described above ([4]) using eight memory circuit groups (i.e.,  $n=8$ ), each of which implements an arbitrary logic circuit of three inputs (i.e.,  $X_i=3$  for all  $i$ ) on at least one output. This would have been a simple substitution ( $n=8$  and  $X_i=3$  for the undefined “ $n$ ” and “ $X_i$ ” in [4]) of known elements (input signals and corresponding memory circuit groups) to obtain predictable results. Moreover, Fujii expressly discloses an embodiment (PL2) in which  $X_i=3$  for each memory circuit group ( $i=1 \dots n$ ). Fujii, ¶100, FIG.13(1). This would have motivated a POSA to implement the 6-input, 3-output PL1 in the same way, using  $X_i=3$  for each memory circuit group ( $i=1 \dots 8$ ).

149. It was well known and predictable that a 6-input look-up table could be constructed using eight 3-input, 1-output look-up tables. *See* Agrawal, 55:34-41, 57:44-58:18. Agrawal discloses “a plurality of eight” LUTs, each of which is a “3-

input LUT.” Agrawal, 55:34-38. Through two layers of “function synthesis” (1225, 1226) (Agrawal, 56:22-56) and a 2-to-1 multiplexer (1227) (Agrawal, 57:58-58:18), these “3-input LUT[s]” can be combined to implement an arbitrary logic circuit of 6 inputs on one output ( $f_c(6T)$ ) as shown below:



Agrawal, FIGS.12A-12B (combined and annotated), col. 55-58.

150. Moreover, a POSA would have had a reasonable expectation of success in so modifying PL1. Fujii expressly discloses that PL1’s “ $2^K$  memory circuits” may be “divided...into plural memory circuit groups” in various ways with each memory circuit group having “ $2^{X_i}$  memory circuits,” where  $X_i$  might be

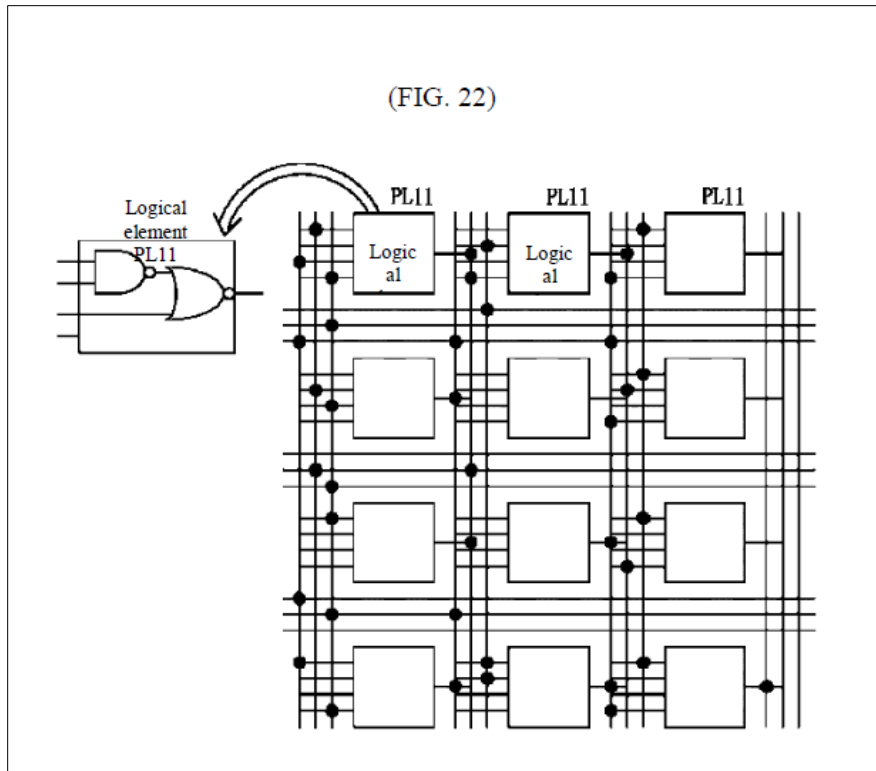
the same for every memory circuit group in PL1. Fujii, ¶14. Fujii also expressly discloses an embodiment (PL2) where  $X_i=3$  for each memory circuit group ( $i=1 \dots n$ ). Fujii, ¶100, FIG.13(1).

151. Fujii therefore renders obvious “[t]he look up table as claimed in claim 4, wherein said 6-input 3-output look up table comprises eight 3-input 1-output LUT units.”

## **G. Independent Claim 7**

### **1. Motivation to Combine the Fujii-Betz PL1 with Betz’s FPGA Architecture**

152. Fujii explains that PL1 is part of “an FPGA in which K-input LUTs are used as programmable logic elements and a plurality of these LUTs are arranged in an array” (Fujii, ¶6) and provides a “configuration example of an FPGA” in FIG. 22.



Fujii, FIG.22, ¶¶1-5, 11, 102. However, Fujii does not describe implementation details about FPGA architecture. A POSA would have been motivated to consult other references, including Betz, to fill in those details. In particular, a POSA would have been motivated to combine the Fujii-Betz PL1 (i.e., PL1 as modified by Betz such that the values of one or both of control signals A and B are stored in SRAM cells (*see supra* [1D])) with Betz’s disclosure of FPGA architecture. *See* Betz, 12-18.

153. Fujii expressly provides such a motivation because the stated “Effect of the Invention” is that “one programmable logic element can function as a programmable logic element with different numbers of inputs,” thereby “allow[ing] arbitrarily non-uniform sizes of programmable logic elements *on the*

**FPGA**, and the desired logic circuit can be achieved with fewer logic elements and fewer memories.” Fujii, ¶102. A POSA would have consulted Betz because it “provides background information about FPGA architecture” (Betz, 11), and PL1 is used in FPGA architecture (Fujii, ¶¶3, 5, 6, 102).

154. Moreover, Betz uses “LUT-based logic blocks” in its FPGA architecture (Betz, 13), similar to the LUT-based logic blocks in Fujii (Fujii, ¶6 (describing “an FPGA in which K-input LUTs are used as programmable logic elements and *a plurality of these LUTs are arranged in an array.*”)). Indeed, Fujii describes an improvement to LUT-based logic blocks:

The present invention, when realizing the desired logic function, is to provide a programmable logic element capable of efficiently realizing a desired logic function by using fewer logic elements and less memory according to the logic function to be realized.

Fujii, ¶9; *see also* Fujii, ¶10, 102.

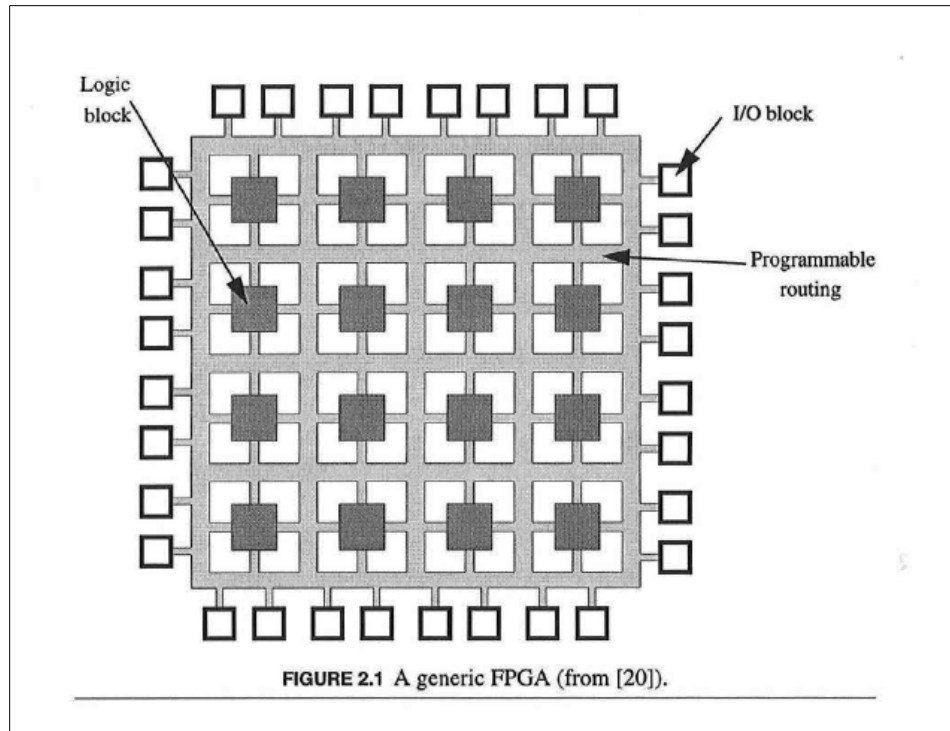
155. A POSA, therefore, would have been motivated to apply Fujii’s known technique (the Fujii-Betz PL1) to improve a known device (Betz’s FPGA architecture) in the same way (providing a programmable logic element capable of using fewer logic elements and less memory). This also amounts to a simple substitution of one known element (the Fujii-Betz PL1) for another (the LUT in Betz’s LUT-based logic block) to obtain predictable results.

156. A POSA would have had a reasonable expectation of success in making this combination. The modified Fujii-Betz PL1 (e.g., [1PRE]-[1D]), would have worked for its intended purpose with Betz's FPGA architecture. And combining the Fujii-Betz PL1 with Betz's FPGA architecture would have led to predictable results, namely "fewer logic elements and less memory according to the logic function to be realized" in that FPGA. Fujii, ¶9.

**2. [7PRE] "A programmable logic circuit device comprising:"**

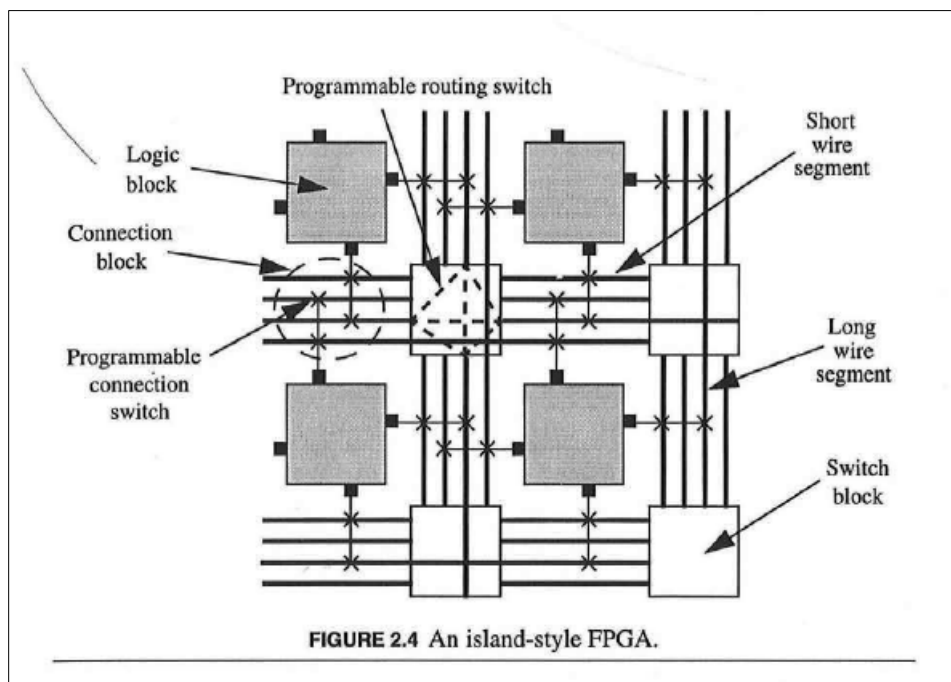
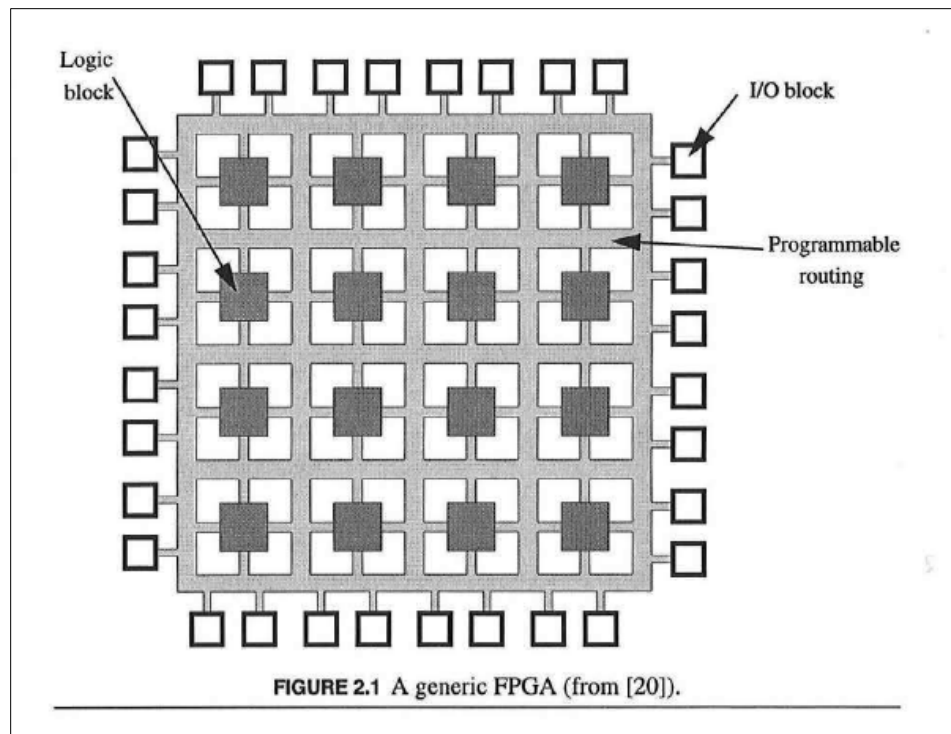
157. Betz discloses this limitation. In particular, Betz discloses a "generic FPGA" that includes "logic blocks, I/O blocks, and programmable routing." Betz, 11-12, FIG.2.1. An FPGA is a type of "*programmable logic circuit device*." '737-patent, 1:13-24 ("Such a *programmable logic circuit device* is known as PLD (Programmable Logic Device) or **FPGA** (Field Programmable Gate Array).").

158. Betz uses "island-style FPGAs" throughout its description of FPGA architecture in §2.1. Betz, 14 ("[W]e will almost exclusively investigate island-style routing architecture," with an exception for "some of the results" in Chapter 5 not relevant here). An example of such an FPGA is shown in FIG.2.1:



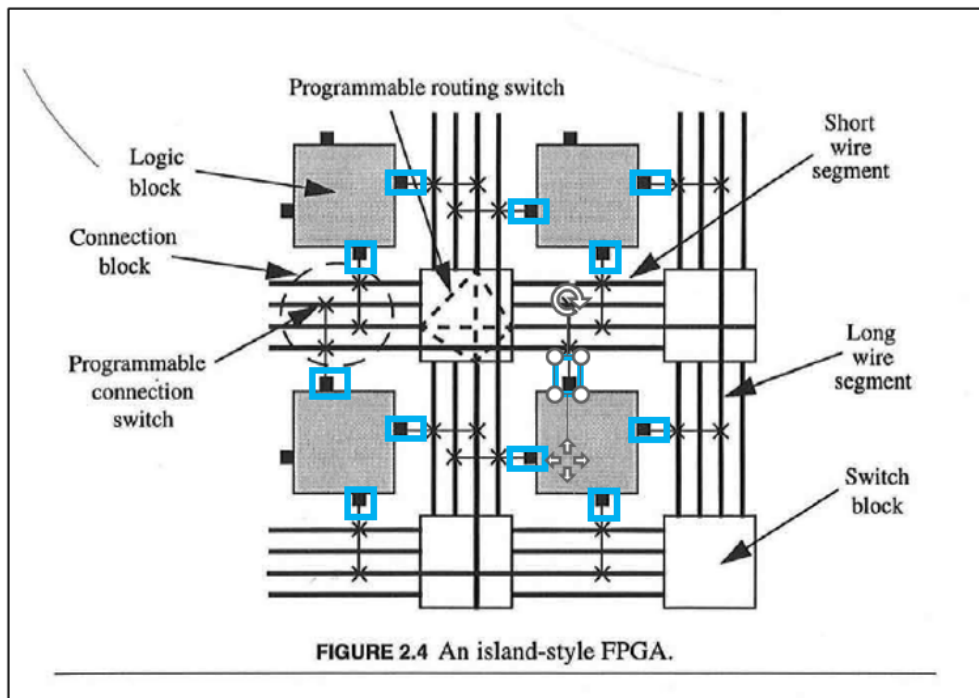
### 3. [7A] “a plurality of logic blocks;”

159. Betz discloses this limitation. Betz discloses a plurality of “LUT-based *logic blocks*,” Betz, 13, which are one of the “fundamental components” of an FPGA, Betz, 11. “A circuit is implemented in an FPGA by programming each of the[se] logic blocks to implement a small portion of the logic required by the circuit.” Betz, 11-15, FIGS. 2.1, 2.4. Betz’s “*plurality of logic blocks*” is shown below in FIG.2.1 and FIG.2.4.



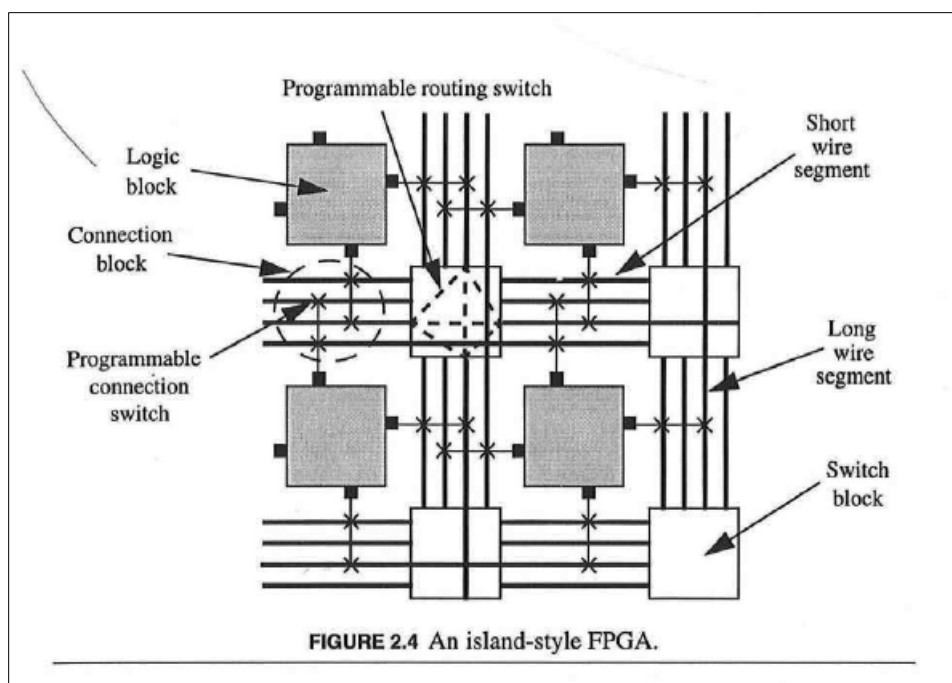
4. [7B] “a plurality of routing wires connected to each of said logic blocks;”

160. Betz discloses this limitation. Betz discloses that its “*logic blocks are surrounded* by routing channels of pre-fabricated *wiring segments on all four sides.*” Betz, 14. In addition, “[a] logic block input or output, which we call a pin, can connect to some or all of the wiring segments in the channel adjacent to it.” Betz, 14-15 (“The number of wires in each channel to which a *logic block pin* can connect is called the connection block flexibility, or  $F_c$ .”). As shown in FIG.2.4 below, Betz’s FPGA thus has “a plurality of routing wires [“short wire segment[s]” and “long wire segment[s]”] connected to each of said logic blocks [via logic block pins (in blue)].” Betz, 14-15, FIGS.2.4, 1.2, 1.3.



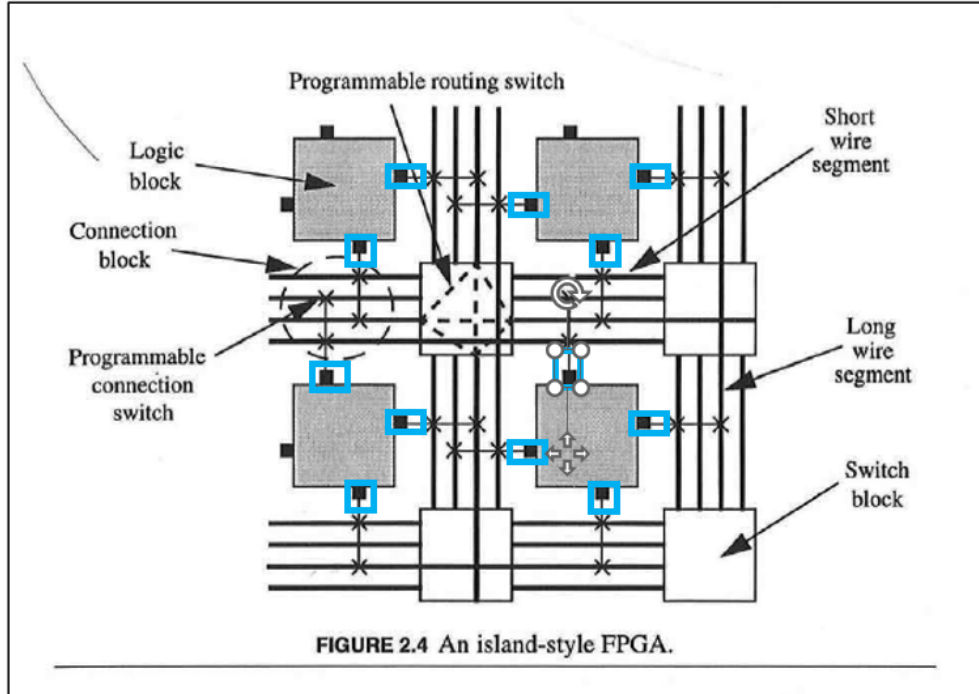
5. [7C] “a plurality of switch circuits provided at an intersection of each of said routing wires;”

161. Betz discloses this limitation. Betz teaches that, “[a]t every intersection of a horizontal channel and a vertical channel, there is a *switch block*....This is simply a set of programmable switches that allow some of the wire segments incident to the switch block to be connected to others; note that for clarity only a few of the programmable switches contained by switch blocks are shown in Figure 2.4.” Betz, 14. The “channel[s]” Betz references are made of the “wiring segments” used for routing. Betz, 14; *see supra* [7B]. As shown in FIG.2.4, Betz thus discloses “a plurality of switch circuits [“switch block[s]” each with a “[p]rogrammable routing switch”] *provided at an intersection of each of said routing wires* [“short wire segment[s]” and “long wire segment[s]”].” Betz, 14-15, FIG.2.4.



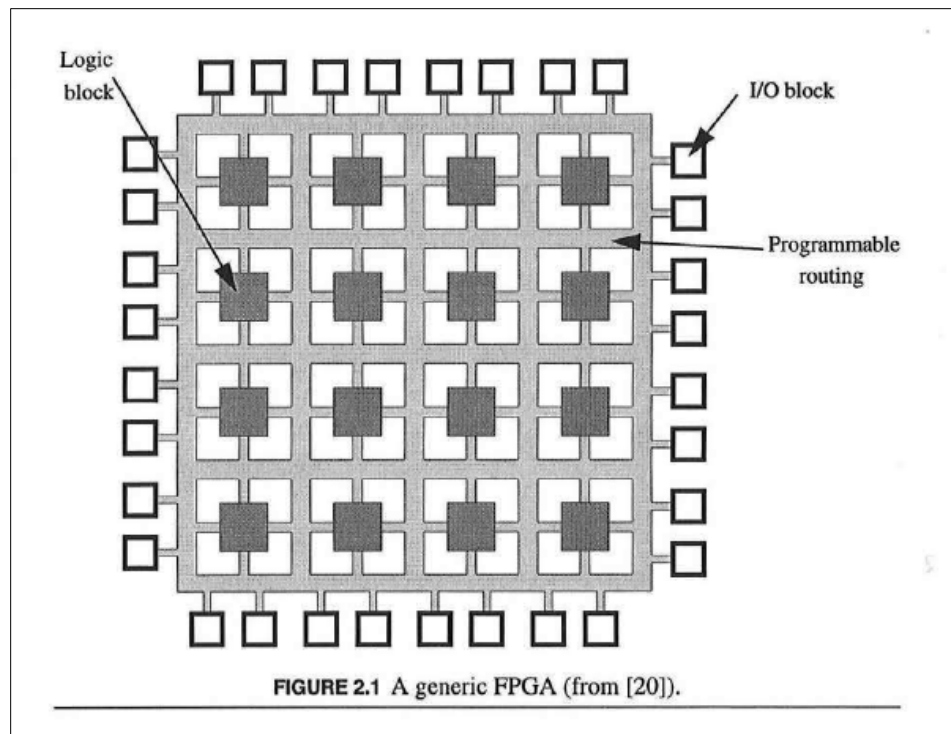
6. [7D] “a plurality of connection blocks provided between an I/O line of each of said logic blocks and each of said routing wires; and”

162. Betz discloses this limitation. In particular, Betz discloses that “[a] *logic block input or output*, which we call a pin, can connect to...*all of the wiring segments* in the channel adjacent to it via a *connection block*...of programmable switches.” Betz, 14. As shown in FIG.2.4, Betz thus discloses “a plurality of *connection blocks* [“[c]onnection block[s]”] *provided between an I/O line of each of said logic blocks* [the logic block pins (in blue)] *and each of said routing wires* [“short wire segment[s]” and “long wire segment[s]”].” Betz, 14-15, FIGS.2.4, 1.2, 1.3.



**7. [7E] “an I/O block performing an input/output operation with external equipment,”**

163. Betz discloses this limitation. In particular, Betz discloses “I/O blocks” that “act as either an input pad or an output pad, as required by the circuit” (Betz, 11), as shown in FIG.2.1 below.



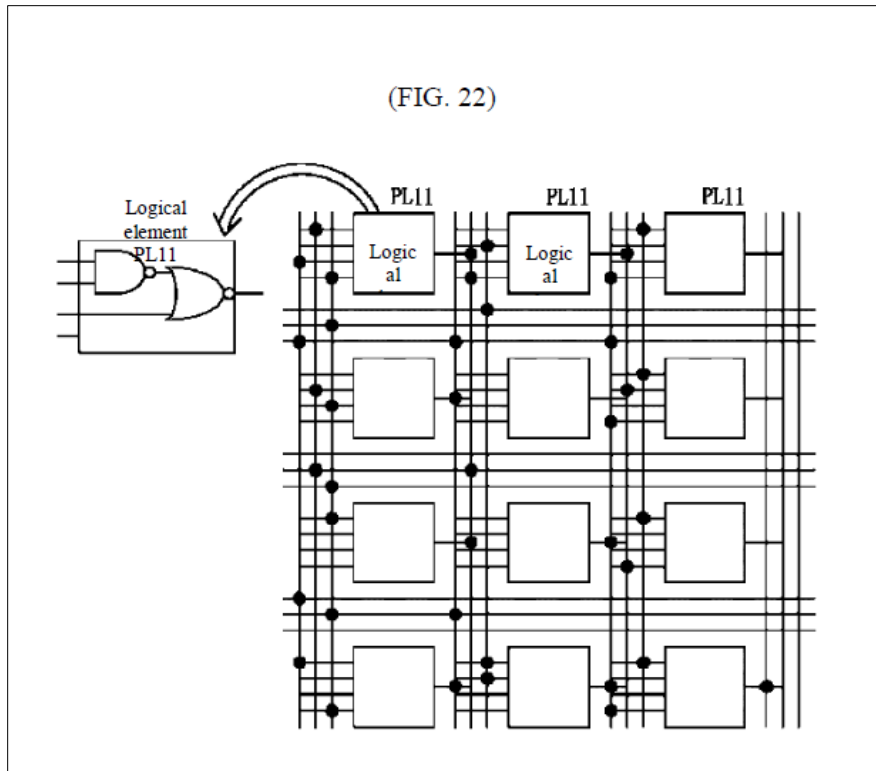
Betz discloses that I/O blocks are connected to pins on the device: “perimeter I/O is assumed (i.e. wire-bonded pads, rather than flip-chip), so I/O blocks can only be placed on the edges of the FPGA.” Betz, 51. Betz also notes that: “Locked I/Os occur frequently in real FPGA designs, since the circuit board on which an FPGA is mounted is often fabricated before the FPGA circuit design is completed. In such cases, the placement tool must keep the circuit I/Os in the locations required by the circuit board design.” Betz, 51. Thus, because the I/O blocks are connected to a pin on the device, they are necessarily used for “*performing an input/output operation with external equipment.*” See also Betz, 2-3 (describing how FPGAs “can be programmed” and “reprogramm[ed],” which necessarily requires the ability to perform input/output operations with external equipment), 208 (describing

programmable SRAM cells); *see also* Trimberger, 15 (“An SRAM-programmable FPGA is programmed by loading *configuration memory cells* from ***an external source.***”); As shown in FIG.2.1, Betz thus discloses “*an I/O block* [“I/O block”] *performing an input/output operation with external equipment.*”

**8. [7F] “wherein each of said logic blocks has a look up table of M inputs and N outputs, comprising:”**

164. As noted, Betz uses “LUT-based logic blocks,” Betz, 13, and a POSA would have been motivated to substitute the Fujii-Betz PL1 (i.e., PL1 as modified by Betz such that the values of one or both of control signals A and B are stored in SRAM cells (*see supra* [1D])) for the LUT(s) in each of those blocks (*see supra* VIII.G.1).

165. Fujii itself uses the same programmable logic element (e.g., PL11) in each logic block of its FPGA, as shown in the “configuration example of an FPGA” in FIG.22. Fujii, ¶¶1-6, 9-10, 102, FIG.22.



As described above in claim 1, the Fujii-Betz PL1 is “a look up table of  $M$  inputs and  $N$  outputs.” A POSA therefore would have been motivated to use the Fujii-Betz PL1 in each of Betz’s logic blocks. Doing so would have been consistent with Fujii’s teaching that “one programmable logic element can function as a programmable logic element with different numbers of inputs,” thereby “allow[ing] arbitrarily non-uniform sizes of programmable logic elements on the FPGA, and the desired logic circuit can be achieved with fewer logic elements and fewer memories.” Fujii, ¶102.

166. Accordingly, “each of said logic blocks [in Betz’s modified FPGA] has a look up table of  $M$  inputs and  $N$  outputs [the Fujii-Betz PL1].”

**9. [7G] “a plurality of LUT units; and”**

167. This limitation is substantially similar to [1A] and is disclosed by Fujii as discussed above. *See supra* [1A].

**10. [7H] “an internal configuration control circuit controlling an internal configuration of said plurality of LUT units,”**

168. This limitation is substantially similar to [1B] and is disclosed by Fujii as discussed above. *See supra* [1C].

**11. [7I] “wherein said internal configuration control circuit comprises a plurality of selectors selecting I/O signals of said plurality of LUT units, and”**

169. This limitation is substantially similar to [1C] and is disclosed by Fujii as discussed above. *See supra* [1C].

**12. [7J] “a selector control circuit having a memory, controlling said plurality of selectors in accordance with data stored in said memory, and defining the internal configuration of said plurality of LUT units.”**

170. This limitation is substantially similar to [1D] and is rendered obvious by Fujii in view of Betz as discussed above. *See supra* [1D].

**H. Claim 8**

**1. [8A] “The programmable logic circuit device as claimed in claim 7, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and”**

171. This limitation is substantially similar to [2A] and is disclosed by Fujii as discussed above. *See supra* [2A].

2. **[8B] “an output signal selector provided at an output of said LUT units selecting an output signal,”**

172. This limitation is substantially similar to [2B] and is disclosed by Fujii as discussed above. *See supra* [2B].

3. **[8C] “said input signal selector and said output signal selector being controlled in accordance with the data stored in said memory.”**

173. This limitation is substantially similar to [2C] and is rendered obvious by Fujii in view of Betz as discussed above. *See supra* [2C].

#### **I. Claim 9**

1. **[9A] “The programmable logic circuit device as claimed in claim 7, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and”**

174. This limitation is substantially similar to [3A] and is disclosed by Fujii as discussed above. *See supra* [3A].

2. **[9B] “an output signal selector provided at an output of said LUT units selecting an output signal,”**

175. This limitation is substantially similar to [3B] and is disclosed by Fujii as discussed above. *See supra* [3B].

3. **[9C] “said input signal selector and said output signal selector being controlled in accordance with the input signal.”**

176. This limitation is substantially similar to [3C] and is rendered obvious by Fujii in view of Betz as discussed above. *See supra* [3C].

**J. Claim 10: “The programmable logic circuit device as claimed in claim 7, wherein said look up table of M inputs and N outputs is a 6-input 3-output look up table.”**

177. Fujii-Betz renders obvious the programmable logic circuit device as claimed in claim 7. *See supra* [7PRE]-[7J]. This limitation is substantially similar to claim 4, and it would have been obvious in light of Fujii. *See supra* [4].

**K. Claim 11: “The programmable logic circuit device as claimed in claim 10, wherein said 6-input 3-output look up table comprises eight 3-input 1-output LUT units.”**

178. Fujii-Betz renders obvious the programmable logic circuit device as claimed in claim 10. *See supra* [7PRE]-[7J], [10]. This limitation is substantially similar to claim 5, and it would have been obvious in light of Fujii as discussed above. *See supra* [5].

**L. Independent Claim 13**

**1. [13PRE] “A method of configuring a look up table of M inputs and N outputs, comprising:”**

179. Fujii discloses this limitation. Fujii’s PL1 is “*a look-up table of M inputs and N outputs.*” *See supra* [1PRE]; Fujii, FIG.1. Moreover, Fujii discloses components that define and control an internal configuration of “*LUT units*” of the look-up table. *See supra* [1A]-[1D]. A POSA would have understood that defining and/or controlling an internal configuration of the “*LUT units*” constitutes “[a] *method of configuring a look up table of M inputs and N outputs.*” *See also supra* [1PRE]-[1D] (substantially similar for PL2 and PL3).

**2. [13A] “providing a plurality of LUT units, and”**

180. As discussed above, Fujii discloses a “*plurality of LUT units*” in PL1. *See supra* claim 1[B]. Likewise, Fujii discloses “*providing a plurality of LUT units.*” *See also supra* [1B] (substantially similar for PL2 and PL3).

**3. [13B] “selectively controlling I/O signals of said plurality of LUT units to set a predetermined mode of an internal configuration.”**

181. Fujii-Betz renders obvious this limitation. Fujii discloses that, “A and B are control signals for determining the *use mode* of the programmable logic circuit PL1.” Fujii, ¶38, FIG.6. As discussed ([1A]-[1D]), control signals A and B “*set*” the use mode of PL1 by controlling selectors and other hardware that, in turn, “*selectively control[] I/O signals of said plurality of LUT units*” to define the internal configuration of “*LUT units*” in each use mode. These use modes are each a “*predetermined mode of an internal configuration[.]*” *See supra* [1B]-[1D]; Fujii, ¶¶12-14, 36-48, FIGS.1, 6. Thus, Fujii-Betz renders obvious “*selectively controlling I/O signals of said plurality of LUT units to set a predetermined mode of an internal configuration.*” *See also supra* [1A]-[1D] (substantially similar for PL2 and PL3).

**M. Claim 14: “The method of configuring a look up table as claimed in claim 13, wherein the I/O signals of said plurality of LUT units**

**are selectively controlled in accordance with data stored in the corresponding look up table.”**

182. Fujii-Betz renders obvious this limitation. As discussed ([1C]-[1D], 13[B]), *“the I/O signals of said plurality of LUT units are selectively controlled”* by selectors that are controlled by control signals A and B to “determin[e] the use mode of the programmable logic circuit PL1.” Fujii, ¶¶36-48, FIGS.1, 6. As explained ([1D]), it would have been obvious to apply the teachings of Betz to modify PL1 to set the values of one or both of control signals A and B using data stored in memory composed of SRAM cells within PL1. *See supra* [1D]. By so doing, *“the I/O signals of said plurality of LUT units are selectively controlled”* by selectors controlled by control signals A and B as shown in FIG.6 *“in accordance with data stored [the values of A and/or B stored in SRAM] in the corresponding look up table [Fujii’s PL1].” See also supra* [1C]-[1D], 13[B] (substantially similar for PL2 and PL3).

**N. Claim 15: “The method of configuring a look up table as claimed in claim 13, wherein an input signal input to at least one of said LUT units and an output signal output from said LUT units are**

**selectively controlled in accordance with data stored in the corresponding look up table.”<sup>12</sup>**

183. Fujii-Betz renders obvious this limitation. As discussed ([1B]-[1C], [2A]-[2B]), Fujii teaches that control signals A and B control input selectors x16-x20, each of which selectively controls “*an input signal input to at least one of said LUT units,*” as well as output selectors x12, x14, x31, and x32, each of which selectively controls “*an output signal output from said LUT units.*” See *supra* [1C]-[1D], [2A]-[2B], [13B]. As discussed ([1C]-[1D], [2A]-[2B], 13[B]), “*an input signal input to at least one of said LUT units and an output signal output from said LUT units are selectively controlled*” by control signals A or B to “determin[e] the use mode of the programmable logic circuit PL1.” Fujii, ¶¶36-48, FIGS.1, 6. As explained ([1D], [2C]), it would have been obvious to apply the teachings of Betz to modify PL1 to set the values of one or both of control signals A and B using data stored in memory composed of SRAM cells within PL1. See *supra* [1D]. By so doing, “*an input signal input to at least one of said LUT units and an output signal output from said LUT units are selectively controlled*” by input selectors x16-20 and output selectors x12, x14, x31, and x32, which in turn are controlled by control signals A or B as shown in FIG.6 “*in accordance with data stored [the*

---

<sup>12</sup> As corrected by the Certificate of Correction entered April 26, 2005. See ’737-patent (last page).

values of A and/or B in SRAM] *in the corresponding look up table* [Fujii's PL1].”

*See also supra* [1B]-[1D], [2A]-[2B], [13B] (substantially similar for PL2 and PL3).

- O. Claim 16: “The method of configuring a look up table as claimed in claim 13, wherein an input signal input to at least one of said LUT units and an output signal output from said LUT units are selectively controlled in accordance with a predetermined function of the input signal.”<sup>13</sup>**

184. Fujii-Betz renders obvious this limitation. *See also supra* [3C] (substantially similar for PL2 and PL3). As discussed above ([14]-[15]), Fujii-Betz teaches that “*an input signal input to at least one of said LUT units and an output signal output from said LUT units are selectively controlled*” by selectors that are controlled by control signals A and B. As noted ([3C]), output selectors x12 and x14 are controlled in accordance with input signals c and d in selecting between output signals. Fujii ¶¶18-19, 29-30, 39, 42, FIG.3. As discussed above ([3C]), it would have been obvious to modify PL1 to use one or more of input signals a-g (e.g., c or d) to control at least one of input selectors x16-x20. By so doing, “*an input signal input to at least one of said LUT units and an output signal output from said LUT units are selectively controlled*” by one of input signals a-g.

---

<sup>13</sup> As corrected by the Certificate of Correction entered April 26, 2005. *See* '737-patent (last page).

185. Alternatively, as explained ([3C]), control signals A and B are each an “*input signal*.”

186. Additionally, a POSA would have understood that one type of “*predetermined function of the input signal*” is the identity function. See Apostol, 50-51 (“Suppose that  $f(x) = x$  for all real  $x$ . This function is often called the *identity function*.”); Tanenbaum, 66, FIG.3-6 (illustrating the “*Identity law*,” including well-known functions “ $1A = A$ ” and “ $0 + A = A$ ”).

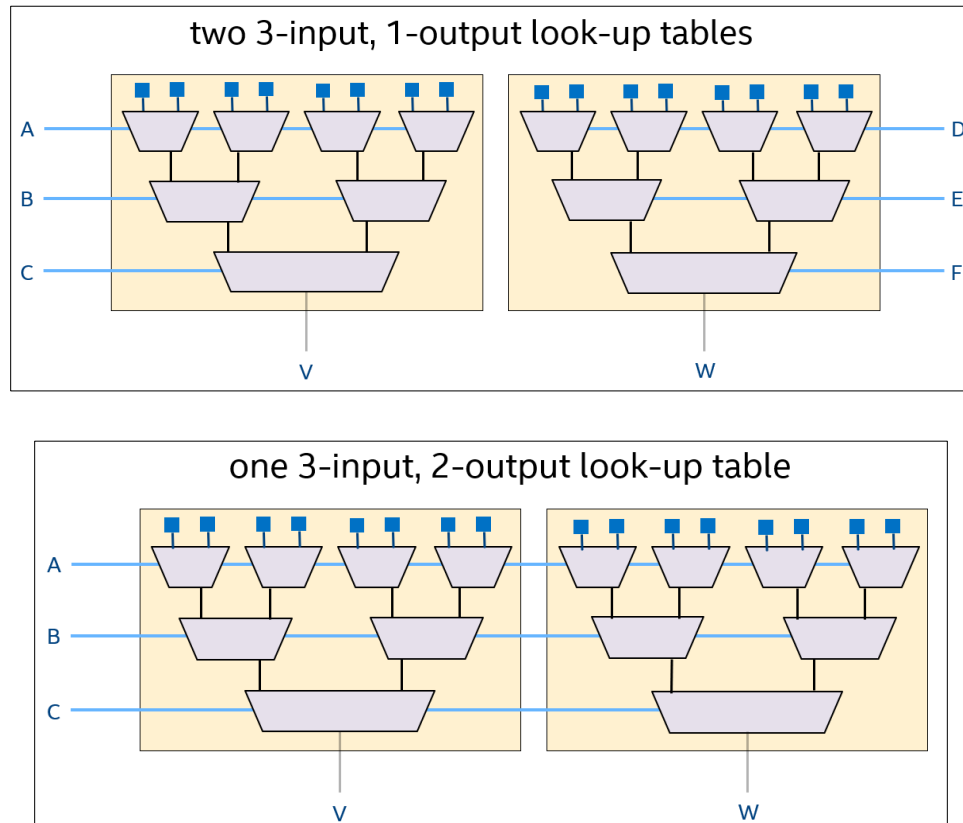
## IX. GROUND 2

187. After reviewing Fujii, Betz, and claims 6 and 12 of the '737 patent, it is my opinion that each of these claims would have been obvious to a POSA in view of Fujii in combination with Betz and the general knowledge of a POSA (Ground 2). The basis for my opinion and the details of my analysis are below.

### A. **Claim 6: “The look up table as claimed in claim 4, wherein said 6-input 3-output look up table comprises four 3-input 2-output LUT units.”**

188. As described above, Fujii-Betz renders obvious “[t]he look up table as claimed in claim 4.” Claim 6’s additional limitations would have been obvious in light of Fujii and the general knowledge of a POSA as of the effective filing date of the '737 patent. In particular, it would have been obvious to modify the 6-input, 3-output PL1 with eight 3-input, 1-output “*LUT units*” discussed above ([5]) to comprise “*four 3-input 2-output LUT units*.”

189. As of the effective filing date, it would have been well understood by a POSA that a 3-input, 2-output “*LUT unit*” can be made of two 3-input, 1-output “*LUT units*” with common inputs:



190. It was well known that “that the memory cells” used to construct look-up tables “are restructurable (by means of programmable switches), so that LUTs with varying number of inputs can be implemented.” Bhat, 35. “If there are 32 [ $2^5$ ] memory cells,” for example, “these can be structured as a 5-input, 1-output LUT, or a **4-input, 2-output LUT**, or a 3-input, 4-output LUT, and so on. Additionally, **two 4-input, 1-output LUTs** (with no common inputs) and four 3-input, 1-output LUTs, and so on, are possible.” Bhat, 35. In the same way, sixteen memory bits

can be structured as two 3-input, 1-output lookup tables or one 3-input, 2-output look-up table that can implement an arbitrary logic circuit of its inputs on both of its outputs independently.

191. A skilled artisan would have been motivated to modify the Fujii-Betz PL1 (*see supra* [1D], [5]) in this way because it would have involved a simple substitution (four 3-input, 2-output “*LUT units*” instead of eight 3-input, 1-output “*LUT units*”) of known elements (configurations of 16 memory cells) to obtain predictable results (a “*LUT unit*” that can perform an arbitrary function of three inputs on each of its two outputs). Moreover, a POSA would have been motivated to implement the 6-input, 3-output look-up table in PL1 with “*four 3-input 2-output LUT units*” in order to “efficiently implement[]” the “small fanin functions” known to be used by “many logic circuits.” He, 3. In particular, there are a limited set of look-up tables that can efficiently perform small fanin functions—namely “2 or 3-input LUTs” (He, 3), and a POSA would have been motivated to use “*3-input, 2-output LUT units*.” It was well known that 3-input look-up tables could be designed to have more than one output. Brown-FPGA, 99. Thus, “*3-input, 2-output LUT units*” would have been among the limited set of small look-up tables capable of implementing small fanin functions efficiently, and it would have been obvious to try all of those small look-up tables to optimize the speed and area of the modified logic element. See He, 1-4 (optimization study).

192. A POSA would have had a reasonable expectation of success in making this modification. As discussed ([5]), a POSA would have had a reasonable expectation of success in modifying the 6-input, 3-output PL1 to comprise “*eight 3-input 1-output LUT units.*” As discussed above (*supra* p.115), a POSA would have also understood that a 3-input, 2-output look-up table comprises two 3-input, 1-output look-up tables with common inputs. Methods of using common inputs between look-up tables were well known at the time and could be accomplished in many, routine ways, e.g., a pre-look-up-table decoding section or pass gates. *See, e.g.,* Agrawal at 32:65-33:29 (pre-LUT decoding section), FIG.6; Sharpe-Geisler at 3:25-40 (pass gates), FIG.6; Bhat, 35 (noting LUTs are “restructurable (by means of programmable switches),” like the pass gates in Sharpe-Geisler). For example, Agrawal teaches that a “pre-LUT decoding section” selectively intercepts the inputs of two 8-bit (3-input, 1-output) look-up tables and ensures those inputs are common. Agrawal, 32:65-33:50, FIG.6. An example of this pre-LUT decoding section is depicted as 623a and 624a in FIG.6 of Agrawal:

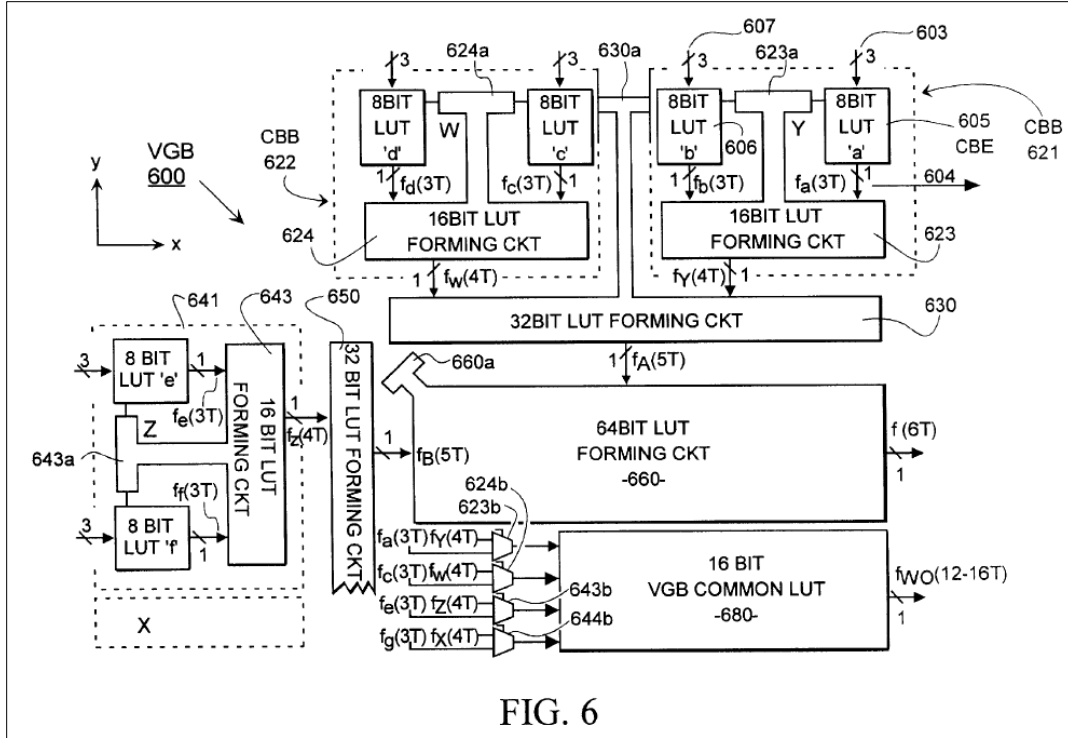


FIG. 6

193. Fujii-Betz therefore renders obvious “[t]he look up table as claimed in claim 4, wherein said 6-input 3-output look up table comprises four 3-input 2-output LUT units.”

**B. Claim 12: “The programmable logic circuit device as claimed in claim 10, wherein said 6-input 3-output look up table comprises four 3-input 2-output LUT units.”**

194. Fujii-Betz renders obvious “[t]he programmable logic circuit device as claimed in claim 10.” See *supra* [7PRE]-[7J], [10]. Claim 12’s additional limitations are substantially similar to those in claim 6 and would have been obvious in light of Fujii and the general knowledge of a POSA. See *supra* [6].

## **X. GROUND 3 AND GROUND 4**

195. To the extent the phrase “a memory” in the challenged claims (*see* [1D] and [7J]) requires a single memory, Fujii in combination with Betz and Chow (Ground 3) and Fujii in combination with Betz, the general knowledge of a POSA, and Chow (Ground 4) render those claims obvious. For the reasons discussed above, Fujii in combination with Betz renders obvious all other limitations in claims 1-5 7-11, and 13-16, *see supra* §VIII, and Fujii in combination with Betz and the general knowledge of a POSA renders obvious all other limitations in claims 6 and 12, *see supra* §IX.

196. Chow discloses a single memory to program multiple configuration bits together in an SRAM-based FPGA. Chow, 325 (“Three techniques for programming the configuration bits were identified. The simplest is to connect all of the bits into a long shift register.”); *see also* Chow, Abstract.

197. A POSA would have been motivated to control the SRAM-based switches taught by Betz using data stored in a single memory within PL1 as disclosed in Chow, because doing so would have made it easier to program and re-program the values of control signals A and B in Fujii’s PL1 when programming the logic element PL1 by a user with the desired logic circuit, including setting the desired use mode. *See* Chow, 325-26 & FIG.12 (describing as the “simplest” method of programming configuration bits in an SRAM-based FPGA

“connect[ing] all of the bits into a long shift register”); *see also* Sharpe-Geisler, 3:16-27, 3:40-54, 4:56-5:29, FIG.6 (describing how multiple pass gates are “controlled from a single memory cell bit 606”); *see also* Brown-FPGA, 1-4 (explaining that memories in programmable devices “can be erased and re-programmed many times”); Fujii, ¶¶1, 2, 38; Betz, 11-12 (SRAM cells used to “configure the programmable...logic blocks”). This combination also would have merely amounted to use of a known technique (programming a shift register with configuration bits) to improve a similar device (Fujii-Betz PL1) in the same way (implementing the simplest technique to program the configuration bits).

198. A POSA would have had a reasonable expectation of success in making this combination. Chow describes “connect[ing] all of the bits into a long shift register” as the “simplest” method of programming configuration bits in an SRAM-based FPGA, so a POSA would have expected success in using the simplest method available.

## **XI. CONCLUSION**

199. This declaration and my opinions herein are made to the best of my knowledge and understanding, and based on the material available to me, at the time of signing this declaration. I declare that all statements made herein on my own knowledge are true and that all statements made on information and belief are believed to be true, and further, that these statements were made with the

knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 or Title 18 of the United States Code.

Date: 4/19/2023

  
\_\_\_\_\_  
Stephen W. Melvin, Ph.D.

## XII. APPENDIX: CLAIM LISTING

The following claim listing assigns element labels (e.g., [1PRE], [1A], etc.) to certain claims for convenience of reference.

|                                                                                                                                                                                                                        |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>CLAIM 1</b>                                                                                                                                                                                                         |
| <b>[1PRE]</b> A look up table of M inputs and N outputs, comprising:                                                                                                                                                   |
| <b>[1A]</b> a plurality of LUT units; and                                                                                                                                                                              |
| <b>[1B]</b> an internal configuration control circuit controlling an internal configuration of said plurality of LUT units,                                                                                            |
| <b>[1C]</b> wherein said internal configuration control circuit comprises a plurality of selectors selecting I/O signals of said plurality of LUT units, and                                                           |
| <b>[1D]</b> a selector control circuit having a memory, controlling said plurality of selectors in accordance with data stored in said memory, and defining the internal configuration of said plurality of LUT units. |
| <b>CLAIM 2</b>                                                                                                                                                                                                         |
| <b>[2A]</b> “The look up table as claimed in claim 1, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and”     |
| <b>[2B]</b> “an output signal selector provided at an output of said LUT units selecting an output signal,”                                                                                                            |
| <b>[2C]</b> “said input signal selector and said output signal selector being controlled in accordance with the data stored in said memory.”                                                                           |
| <b>CLAIM 3</b>                                                                                                                                                                                                         |
| <b>[3A]</b> The look up table as claimed in claim 1, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and       |
| <b>[3B]</b> an output signal selector provided at an output of said LUT units selecting an output signal,                                                                                                              |
| <b>[3C]</b> said input signal selector and said output signal selector being controlled in accordance with the input signal.                                                                                           |
| <b>CLAIM 4</b>                                                                                                                                                                                                         |
| <b>[4]</b> The look up table as claimed in claim 1, wherein said look up table of M inputs and N outputs is a 6-input 3-output look up table.                                                                          |
| <b>CLAIM 5</b>                                                                                                                                                                                                         |
| <b>[5]</b> The look up table as claimed in claim 4, wherein said 6-input 3-output look up table comprises eight 3-input 1-output LUT units.                                                                            |
| <b>CLAIM 6</b>                                                                                                                                                                                                         |

|                                                                                                                                                                                                                               |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| [6] The look up table as claimed in claim 4, wherein said 6-input 3-output look up table comprises four 3-input 2-output LUT units.                                                                                           |
| <b>CLAIM 7</b>                                                                                                                                                                                                                |
| [7PRE] A programmable logic circuit device comprising:                                                                                                                                                                        |
| [7A] a plurality of logic blocks;                                                                                                                                                                                             |
| [7B] a plurality of routing wires connected to each of said logic blocks;                                                                                                                                                     |
| [7C] a plurality of switch circuits provided at an intersection of each of said routing wires;                                                                                                                                |
| [7D] a plurality of connection blocks provided between an I/O line of each of said logic blocks and each of said routing wires; and                                                                                           |
| [7E] an I/O block performing an input/output operation with external equipment,                                                                                                                                               |
| [7F] wherein each of said logic blocks has a look up table of M inputs and N outputs, comprising:                                                                                                                             |
| [7G] a plurality of LUT units; and                                                                                                                                                                                            |
| [7H] an internal configuration control circuit controlling an internal configuration of said plurality of LUT units,                                                                                                          |
| [7I] wherein said internal configuration control circuit comprises a plurality of selectors selecting I/O signals of said plurality of LUT units, and                                                                         |
| [7J] a selector control circuit having a memory, controlling said plurality of selectors in accordance with data stored in said memory, and defining the internal configuration of said plurality of LUT units.               |
| <b>CLAIM 8</b>                                                                                                                                                                                                                |
| [8A] The programmable logic circuit device as claimed in claim 7, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and |
| [8B] an output signal selector provided at an output of said LUT units selecting an output signal,                                                                                                                            |
| [8C] said input signal selector and said output signal selector being controlled in accordance with the data stored in said memory.                                                                                           |
| <b>CLAIM 9</b>                                                                                                                                                                                                                |
| [9A] The programmable logic circuit device as claimed in claim 7, wherein said plurality of selectors include: an input signal selector provided at an input of at least one of said LUT units to select an input signal; and |
| [9B] an output signal selector provided at an output of said LUT units selecting an output signal,                                                                                                                            |
| [9C] said input signal selector and said output signal selector being controlled in accordance with the input signal.                                                                                                         |
| <b>CLAIM 10</b>                                                                                                                                                                                                               |

|                                                                                                                                                                                                                                                                                                           |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>[10]</b> The programmable logic circuit device as claimed in claim 7, wherein said look up table of M inputs and N outputs is a 6-input 3-output look up table.                                                                                                                                        |
| <b>CLAIM 11</b>                                                                                                                                                                                                                                                                                           |
| <b>[11]</b> The programmable logic circuit device as claimed in claim 10, wherein said 6-input 3-output look up table comprises eight 3-input 1-output LUT units.                                                                                                                                         |
| <b>CLAIM 12</b>                                                                                                                                                                                                                                                                                           |
| <b>[12]</b> The programmable logic circuit device as claimed in claim 10, wherein said 6-input 3-output look up table comprises four 3-input 2-output LUT units.                                                                                                                                          |
| <b>CLAIM 13</b>                                                                                                                                                                                                                                                                                           |
| <b>[13PRE]</b> A method of configuring a look up table of M inputs and N outputs, comprising:                                                                                                                                                                                                             |
| <b>[13A]</b> providing a plurality of LUT units, and                                                                                                                                                                                                                                                      |
| <b>[13B]</b> selectively controlling I/O signals of said plurality of LUT units to set a predetermined mode of an internal configuration.                                                                                                                                                                 |
| <b>CLAIM 14</b>                                                                                                                                                                                                                                                                                           |
| <b>[14]</b> The method of configuring a look up table as claimed in claim 13, wherein the I/O signals of said plurality of LUT units are selectively controlled in accordance with data stored in the corresponding look up table.                                                                        |
| <b>CLAIM 15</b>                                                                                                                                                                                                                                                                                           |
| <b>[15]</b> The method of configuring a look up table as claimed in claim 13, wherein an input signal input to at least one of said LUT units and an output signal output from said LUT units <sup>14</sup> are selectively controlled in accordance with data stored in the corresponding look up table. |
| <b>CLAIM 16</b>                                                                                                                                                                                                                                                                                           |
| <b>[16]</b> The method of configuring a look up table as claimed in claim 13, wherein an input signal input to at least one of said LUT units and an output signal output from said LUT units <sup>15</sup> are selectively controlled in accordance with a predetermined function of the input signal.   |

---

<sup>14</sup> As corrected by the Certificate of Correction entered April 26, 2005.

<sup>15</sup> As corrected by the Certificate of Correction entered April 26, 2005.