(19) Octrooicentrum Nederland

(11) 2012319

(12) C OCTROOI

(21) Aanvraagnummer: **2012319**

(22) Aanvraag ingediend: **25.02.2014**

(51) Int.Cl.:
G06T 1/60 (2006.01)

(30) Voorrang:
28.02.2013 KR 10-2013-0022159

(43) Aanvraag gepubliceerd:
10.09.2014

(47) Octrooi verleend:
24.02.2015

(45) Octrooischrift uitgegeven:
04.03.2015

(73) Octrooihouder(s):
Samsung Electronics Co., Ltd
te Suwon, Zuid-Korea (KR).

(72) Uitvinder(s):
Min Woo Song te Seoul (KR).
Min Soo Kim te Gyeonggi-do (KR).
Sung Chul Yoon te Gyeonggi-do (KR).
Jae Young Hur te Gyeonggi-do (KR).

(74) Gemachtigde:
Ir. H.V. Mertens c.s. te Rijswijk.

(54) **Method for rotating an original image using self-learning and apparatuses performing the method.**

(57) A method of rotating an original image includes performing a self-learning using addresses related to at least one page miss and generating address generation rules using a result of the self-learning. The method includes pre-fetching the original image from a memory device based on the address generation rules to obtain a pre-fetched image and generating a rotated image using the pre-fetched image.

NL C 2012319

METHOD FOR ROTATING AN ORIGINAL IMAGE USING SELF-LEARNING AND APPARATUSES PERFORMING THE METHOD

CROSS-REFERENCE TO RELATED APPLICATIONS

[1] This application claims priority from Korean Patent Application No. 10-2013-0022159 filed on February 28, 2013, the disclosure of which is hereby incorporated by reference in its entirety.

BACKGROUND

[2] At least one example embodiment of inventive concepts relates to an image rotation technology, and more particularly to a method of rotating an original image based on address generation rules generated through self-learning, and/or devices performing the method.

[3] As high performance mobile devices have been developed, display devices of the high performance mobile devices have been increased in resolution and size.

[4] When a user rotates the display device in a specific direction, an image displayed on the display device may switch orientations. For example, the displayed image may switch from a landscape shape or landscape orientation to a portrait shape or portrait orientation.

[5] A rotator performs the function of rotating an image to be displayed on the display device and generating a rotated image.

SUMMARY

[6] According to at least one example embodiment, a method of rotating an original image includes performing a self-learning using a pre-fetching step and using addresses related to at least one page miss occurring as a result of the pre-fetching step, and generating address generation rules using a result of the self-learning. The method further includes pre-fetching the original image from a memory device based on the address generation rules to obtain a pre-fetched image, generating a rotated image using the pre-fetched image, and determining a number of address generation rules to be generated based on a control value stored in a memory for a direction in which the original image is to be rotated.

[7] According to at least one example embodiment, the generating address generation rules generates the address generation rules based on a control value set in a special function register (SFR).

[8] According to at least one example embodiment, the pre-fetching the original image includes pre-fetching the original image from the memory device to an image buffer based on the address generation rules to obtain the pre-fetched image.

[9] According to at least one example embodiment, the method further includes programming, by a manufacturer or a user, a number of the address generation rules to be generated.

[10] According to at least one example embodiment, the method further includes determining a number of address generation rules to be generated based on a control value stored in a memory and representative of a number of page misses occurring in an initial frame of the rotated image for a direction in which the original image is to be rotated.

[11] According to at least one example embodiment, the self-learning is performed based on a position of the at least one page miss and a difference between the addresses related to the at least one page miss.

[12] According to at least one example embodiment, the method further includes initializing the address generation rules whenever a direction in which the original image is rotated is changed.

[13] According to at least one example embodiment, the addresses related to at least one page miss correspond to a position of the at least one page miss that is calculated based on a result of counting a number of an address using a counter.

[14] According to at least one example embodiment, a system on chip (SoC) includes a memory management unit (MMU). The MMU is configured to perform a self-learning using pre-fetching and using addresses related to at least one page miss occurring as a result of the pre-fetching, determine a number of address generation rules to be generated based on a control value stored in a memory for a direction in which the original image is to be rotated, generate address generation rules using a result of the self-learning, and pre-fetch an original image from a memory device to an image buffer based on the address generation rules. The rules. The SoC further includes a rotator configured to generate a rotated image using an image pre-fetched to the image buffer.

[15] According to at least one example embodiment, the MMU is configured to determine a number of the address generation rules to be generated according to a number of page misses occurring in an initial frame of the rotated image.

[16] According to at least one example embodiment, the MMU includes a special function register SFR, an address generation rule generation circuit, and a pre-fetch circuit. The special function register (SFR) is configured to store a control value. The address generation rule generation circuit is configured to generate the address generation rules based on the control value stored in the SFR. The pre-fetch circuit is configured to pre-fetch the original image to the image buffer according to the address generation rules.

[17] According to at least one example embodiment, the SoC further includes a central processing unit (CPU) configured to control a number of the address generation rules to be generated by the MMU in response to control signals indicating a direction in which the original image is rotated.

[18] According to at least one example embodiment, an application processor includes the above described SoC.

[19] According to at least one example embodiment, the MMU of the application processor of is configured to determine a number of the address generation rules to be generated according to a number of page misses occurring in an initial frame of the rotated image.

[20] According to at least one example embodiment, the MMU of the application processor includes, a special function register (SFR) configured to store a control value, an address generation rule generation circuit configured to generate the address generation rules based on the control value stored in the SFR, and a pre-fetch circuit configured to pre-fetch the original image to the image buffer according to the address generation rules.

[21] According to at least one example embodiment, a mobile device includes a memory device configured to store an original image and a memory management unit (MMU). The MMU is configured to perform a self-learning using pre-fetching and using addresses related to at least one page miss occurring as a result of the pre-fetching, determine a number of address generation rules to be generated based on a control value stored in a memory and representative of a number of page misses occurring in an initial frame of the rotated image for a direction in which the original image is to be rotated, generate address generation rules using a result of the self-learning, and pre-fetch an original image from the memory device to an image buffer according to the address generation rules. The mobile device further includes a rotator configured to generate a rotated image using the image pre-fetched to the image buffer, and a display controller configured to transmit the rotated image generated by the rotator to a display.

[22] According to at least one example embodiment, the mobile device further includes a central processing unit (CPU) configured to control a number of the address generation rules to be generated by the MMU in response to control signals indicating a direction in which the original image is to be rotated.

[23] According to at least one example embodiment, the mobile device further includes a control signal generator configured to detect the direction in which the original image is to be rotated and generate the control signals according to a result of the detection.

[24] According to at least one example embodiment, the mobile device further includes a rotation sensor configured to detect the direction in which the original image is to be rotated, and a control signal generator configured to generate the control signals in response to a sensing signal output from the rotation sensor.

[25] According to at least one example embodiment, the MMU of the mobile device includes a special function register (SFR) configured to store a control value, an address generation rule generation circuit configured to generate the address generation rules based on the control value stored in the SFR, and a pre-fetch circuit configured to pre-fetch the original image to the image buffer according to the address generation rules.

[26] According to at least one example embodiment, a mobile device includes a display configured to display a rotated image, and a computing device. The computing device is configured to detect a direction of rotation of the device, generate address generation rules using a result of self-learning, the self-learning being performed using a pre-fetching and using addresses related to at least one page miss occurring as a result of the pre-fetching, the generation of the address generation rules being based on the detected direction of rotation, wherein a number of address generation rules to be generated is based on a control value stored in a memory for a direction in which the original image is to be rotated, generate the rotated image using an original image according to the address generation rules such that a page miss does not occur in the rotated image, and transmit the rotated image to the display.

[27] According to at least one example embodiment, the computing device is configured generate the address generation rules based on a position of page misses associated with the detected direction of rotation, and a difference between addresses related to the page misses.

[28] According to at least one example embodiment, the computing device is configured to generate the rotated image by pre-fetching the original image from a memory device to an image buffer based on the address generation rules.

BRIEF DESCRIPTION OF THE DRAWINGS

[29] These and/or other aspects and advantages of inventive concepts will become apparent and more readily appreciated from the following description of example embodiments, taken in conjunction with the accompanying drawings of which:

[30] FIG. 1is a schematic block diagram of a computing system according to at least one example embodiment of inventive concepts;

[31] FIG. 2 schematically depicts a memory map of an external memory according to at least one example embodiment;

[32] FIGS. 3A and 3B each depict an address access order for displaying an original image and a display image according to at least one example embodiment;

[33] FIGS. 4A and 4B each depict an address access order for displaying an image rotated in a X-axis direction and a display image according to at least one example embodiment;

[34] FIG. 5 is a conceptual diagram for describing address generation rules for generating an image rotated in the X-axis direction according to at least one example embodiment;

[35] FIGS. 6A and 6B each depict an address access order for displaying an image rotated in a Y-axis direction and a display image according to at least one example embodiment;

[36] FIG. 7 is a conceptual diagram for describing address generation rules for generating an image rotated in the Y-axis direction according to at least one example embodiment;

[37] FIGS. 8A and 8B each depict an address access order for displaying a 90°-rotated image and a display image according to at least one example embodiment;

[38] FIG. 9 is a conceptual diagram for describing address generation rules for generating the 90°-rotated image according to at least one example embodiment;

[39] FIGS. 10A and 10B each depict an address access order for displaying a 180°-rotated image and a display image according to at least one example embodiment;

[40] FIG. 11 is a conceptual diagram for describing address generation rules for generating the 180°-rotated image according to at least one example embodiment;

[41] FIGS. 12A and 12B each depict an address access order for displaying an image which is rotated by 90° and then rotated in the X-axis direction and a display image according to at least one example embodiment;

[42] FIG. 13 is a conceptual diagram for describing address generation rules for generating the image which is rotated by 90° and then rotated in the X-axis direction according to at least one example embodiment;

[43] FIGS. 14A and 14B each depict an address access order for displaying an image which is rotated by 90° and then rotated in the Y-axis direction and a display image according to at least one example embodiment;

[44] FIG. 15 is a conceptual diagram for describing address generation rules for generating the image which is rotated by 90° and then rotated in the Y-axis direction according to at least one example embodiment;

[45] FIGS. 16A and 16B each depict an address access order for displaying an image which is rotated by 270° and a display image according to at least one example embodiment;

[46] FIG. 17 is a conceptual diagram for describing address generation rules for generating the image which is rotated by 270° according to at least one example embodiment;

[47] FIG. 18 is an example embodiment of a control value table including the number of address generation rules determined according to a rotation direction;

[48] FIG. 19 is a block diagram of the memory management unit illustrated in FIG. 1; and

[49] FIG. 20 is a flowchart for describing a method of rotating an original image according to at least one example embodiment of inventive concepts.

DETAILED DESCRIPTION OF THE EMBODIMENTS

[50] Inventive concepts now will be described more fully hereinafter with reference to the accompanying drawings, in which example embodiments are shown. Inventive concepts may, however, be embodied in many different forms and should not be construed as limited to the example embodiments set forth herein. Rather, these example embodiments are provided so that this disclosure will be thorough and complete, and will fully convey the scope of inventive concepts to those skilled in the art. In the drawings, the size and relative sizes of layers and regions may be exaggerated for clarity. Like numbers refer to like elements throughout.

[51] It will be understood that when an element is referred to as being "connected" or "coupled" to another element, it can be directly connected or coupled to the other element or intervening elements may be present. In contrast, when an element is referred to as being "directly connected" or "directly coupled" to another element, there are no intervening elements present. As used herein, the term "and/or" includes any and all combinations of one or more of the associated listed items and may be abbreviated as "/".

[52] It will be understood that, although the terms first, second, etc. may be used herein to describe various elements, these elements should not be limited by these terms. These terms are only used to distinguish one element from another. For example, a first signal could be termed a second signal, and, similarly, a second signal could be termed a first signal without departing from the teachings of the disclosure.

[53] The terminology used herein is for the purpose of describing particular embodiments only and is not intended to be limiting. As used herein, the singular forms "a", "an" and "the" are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will be further understood that the terms "comprises" and/or "comprising," or "includes" and/or "including" when used in this specification, specify the presence of stated features, regions, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, regions, integers, steps, operations, elements, components, and/or groups thereof.

[54] Unless otherwise defined, all terms (including technical and scientific terms) used herein have the same meaning as commonly understood by one of ordinary skill in the art to which this invention belongs. It will be further understood that terms, such as those defined in commonly used dictionaries, should be interpreted as having a meaning that is consistent with their meaning in the context of the relevant art and/or inventive concepts, and will not be interpreted in an idealized or overly formal sense unless expressly so defined herein.

[55] FIG. 1 is a schematic block diagram of a computing system according to at least one example embodiment of inventive concepts. Referring to FIG. 1, a computing system 100 may include a computing device 110, an external memory 120, a display 130, and a rotation sensor 140.

[56] The computing system 100 may be embodied in a personal computer (PC), a portable electronic device (or a mobile device), an electronic device including a display 130 displaying a rotatable image, etc.

[57] The portable electronic device may be embodied in a laptop computer, a mobile phone, a smart phone, a tablet PC, a personal digital assistant (PDA), an enterprise digital assistant (EDA), a digital still camera, a digital video camera, a portable multimedia player (PMP), a personal navigation device or portable navigation device (PND), a handheld game console, a mobile internet device (MID), an e-book, etc.

[58] The computing device 110 may process (e.g., rotate) an original image output from an external memory 120 or an internal memory (not shown) embodied therein, and transmit the processed image to the display 130.

[59] The computing device 110 may perform a self-learning using addresses (e.g., a difference between addresses) related to at least one page miss, generate address generation rules using a result of the self-learning, pre-fetch an original image from a memory device, e.g., the external memory 120 or an internal memory, according to the address generation rules, generate a rotated image using the pre-fetched image, and transmit the rotated image to the display 130.

[60] The computing device 110 may be embodied in a printed circuit board (PCB) such asa motherboard, an integrated circuit (IC), a system on chip (SoC), or the like.

[61] The computing device 110 includes a central processing unit (CPU) 113, a memory management unit (MMU) 115, a rotator 117, an image processing module 118, a memory controller 119, a display controller 121, and a control signal generator 123.

[62] The CPU 113 may control an operation of the computing device 110. For example, the CPU 113 may control an operation of at least one of a plurality of components 115, 117, 119, 121, and 123 through a bus 111.

[63] The CPU 113 may control or set the number of address generation rules to be generated in the MMU 115 in response to control signals indicating a direction in which an original image stored in the external memory 120 or an internal memory is to be rotated.

[64] As illustrated in FIG. 18, a control value table 113-1 including each of control values for each of rotation directions may be stored in a memory (not shown) included in the CPU 113, or loaded from other memory to the CPU 113 when the computing system is booted. The other memory may be the external memory 120 or an additional memory.

[65] For example, in response to the control signals, the CPU 113 may set or program control values controlling the number of address generation rules to be generated in the MMU 115 in a special function register (SFR) 116. Here, the CPU 113 may refer to the control value table 113-1 based on the control signals.

[66] The MMU 115 may perform a self-learning using a difference between addresses related to at least one page miss, generate address generation rules using a result of the self-learning, and pre-fetch data included in an original image from the external memory 120 to an image buffer 115-1 of the MMU 115 based on the address generation rules.

[67] When data stored in the external memory 120 is pre-fetched by pages by the MMU 115, the image buffer 115-1 may perform a function of a page buffer.

[68] For convenience of description, an embodiment is illustrated where data stored in the external memory 120 is pre-fetched by the MMU 115; however, data to be pre-fetched by the MMU 115 may be embodied inside the computing device 110.

[69] The rotator 117 may generate a rotated image using an image pre-fetched to the MMU 115, e.g., the image buffer 115-1.

[70] According to at least one example embodiment, an image processing module 118 may process a rotated image output from the rotator 117, and transmit the processed image to the display controller 121 through a bus 111 (dashed arrow).

[71] According to at least one other example embodiment, the image processing module 118 may process the rotated image output from the rotator 117 and transmit the processed image to the display controller 121 (solid arrow).

[72] The memory controller 119 may transmit data of an original image stored in the external memory 120 to the MMU 115 by a constant size, e.g., a page, according to a control of the MMU 115, e.g., in response to a pre-fetch operation of the MMU 115. In addition, the memory controller 119 may transmit an image stored in the external memory 120 to the display controller 121 according to a control of the CPU 113.

[73] The display controller 121 may transmit the rotated image processed by the image processing module 118 to the display 130.

[74] The control signal generator 123 may generate a plurality of control signals in response to a sensing signal output from the rotation sensor 123. The plurality of control signals may be interpreted by the CPU 113, and be used as a plurality of indication signals indicating a direction in which an original image is rotated according to a result of the interpretation.

[75] The external memory 120 may store an original image, e.g., a still image or a moving image. The external memory 120 may be embodied in a hard disk drive (HDD), a solid state drive (SSD), or the like.

[76] The external memory 120 may be a volatile memory device or a non-volatile memory device. The volatile memory device may be embodied in a dynamic random access memory(DRAM), a static random access memory(SRAM), a thyristor RAM(T-RAM), a zero capacitor RAM(Z-RAM), a Twin Transistor RAM(TTRAM), etc.

[77] The non-volatile memory device may be an electrically erasable programmable read-only memory(EEPROM), a flash memory, a Magnetic RAM(MRAM), a Spin-Transfer Torque(STT)-MRAM, a Conductive bridging RAM(CBRAM), a Ferroelectric RAM (FeRAM), a Phase change RAM(PRAM), a Resistive RAM(RRAM), a Nanotube RRAM, a Polymer RAM(PoRAM), a Nano Floating Gate Memory(NFGM), a holographic memory, a Molecular Electronics Memory Device, an Insulator Resistance Change Memory, etc.

[78] The non-volatile memory device may be a flash-based memory device, e.g., a secure digital(SD) card, a multimedia card(MMC), an embedded-MMC(eMMC), a universal serial bus(USB) flash driver, a universal flash storage(UFS), etc.

[79] According to a control of the display controller 121, the display 130 may display an original image which is not rotatable or a rotated image.

[80] The display 130 may be a flat panel display. The flat panel display may be a thin film transistor-liquid crystal display(TFT-LCD), a light emitting diode(LED) display, an organic LED(OLED) display, an active-matrix OLED(AMOLED) display, a flexible display, etc.

[81] The rotation sensor 140 is a sensor which may perform a function of detecting rotation or a rotation direction of the computing system 100 and generating a sensing signal. For example, the rotation sensor 140 may bean accelerometer sensor, a gyro sensor, or a combined sensor thereof.

[82] FIG. 2 schematically illustrates a memory map of the external memory according to at least one example embodiment. The memory map 121shown in FIG. 2 is exemplarily illustrated for convenience of description. The memory map 121 denotes each data “A” to “Z”, and “a” to “v” stored in each memory region corresponding to each address D1 to D48. For example, each data “A” to “Z” and “a” to “v” may denote page data.

[83] A first frame of an original image includes data “A” to “P”, a second frame of the original image includes “Q” to “f”, and a third frame of the original image includes data “g” to “v”.

[84] Referring to FIGS. 2, 3A, 3B, 4A, 4B, 6A, 6B, 8A, 8B, 10A, 10B, 12A, 12B, 14A, 14B, 16A, and 16B, a number “1” to “32” in each small box indicates an address access order or a data pre-fetch order according to a rotation direction, and English alphabet letters “A” to “Z” and “a” to “f” in each larger box denote data included in the original image.

[85] In addition, a first frame OFF of the original image includes data “A” to “P”, and a second frame OSF of the original image includes data “Q” to “Z” and “a” to “f”. Here, the MMU 115 per-fetches each data included in each frame OFF and OSF from the external memory 120 to the image buffer 115-1 according to the address access order “1” to “32”.

[86] FIGS. 3A and 3B each illustrate an address access order for displaying an original image and a display image according to at least one example embodiment. FIG. 3A illustrates each frame OFF and OSF including data to be accessed by the MMU 115 according to the address access order “1” to “32”. FIG. 3B is an image displayed on the display device 130.

[87] In order to display a first frame FF, the MMU 115 pre-fetches each data “A” to “P” from the external memory 120 based on addresses ADD1=D1 to ADD1=D16 sequentially output from the rotator 117 according to the address access order “1” to “16”, and transmits each pre-fetched data DATA=“A” to DATA=“P” to the rotator 117.

[88] For example, the MMU 115 pre-fetches each current data “A” to “P”, corresponding to each current address D1 to D16 sequentially output from the rotator 117, from the external memory 120 to the image buffer 115-1 through the memory controller 119.

[89] While each current data “A” to “P” is sequentially transmit to the rotator 117, the MMU 115 estimates each next address D2 to D17 and pre-fetches each next data “B” to “Q” from the external memory 120 to the image buffer 115-1 according to a result of expectation, so that a page miss does not occur in the MMU 115 even though each next address D2 to D17 is input.

[90] Accordingly, the first frame OFF of the original image is normally displayed on the display 130. That is, the address access order “1” to “16” is the same as an order of the address ADD1=D1 to ADD1=D16 sequentially output from the rotator 117, so that the page miss does not occur in the MMU 115.

[91] The rotator 117 may transmit each data DATA=“A” to DATA=“P” output from the MMU 115 to the image processing module 118, and the image processing module 118 may process each data DATA=“A” to DATA=“P”, and transmit each processed data to the display controller 121 directly or through a bus 111.

[92] Accordingly, as illustrated in FIG. 3B, the display controller 121 in a normal state NORMAL may display the first frame FF, including each data “A” to “P” transmitted from the rotator 117, through the display 130.

[93] In order to display a second frame SF, the MMU 115 pre-fetches each data “Q” to “f” from the external memory 120 based on addresses ADD1=D17 to ADD1=D32 sequentially output from the rotator 117 according to an address access order “17” to “32”, and transmits each pre-fetched data DATA=“Q” to DATA=“f” to the rotator 117. Here, the address access order “17” to “32” is the same as an order of an address ADD1=D17 to ADD1=D32 sequentially output from the rotator 117, so that a page miss does not occur in the MMU 115.

[94] The rotator 117 may transmit each data DATA=“Q” to DATA=“f” output from the MMU 115 to the image processing module 118, process the each data DATA=“Q” to DATA=“f”, and transmit the each processed data to the display controller 121 directly or through the bus 111.

[95] Accordingly, as illustrated in FIG. 3B, the display controller 121 in a normal state NORMAL may display a second frame SF including each data “Q” to “f” transmitted from the rotator 117 through the display 130. That is, a first frame OFF of an original image in a normal state NORMAL is displayed as it is, and a second frame OSF of the original image is displayed as it is.

[96] FIGS. 4A and 4B each illustrate an address access order for displaying an image rotated in an X-axis direction and a display image according to at least one example embodiment.FIG. 5 is a conceptual diagram for describing address generation rules for generating an image rotated in the X-axis direction according to at least one example embodiment.

[97] Referring to FIGS. 1, 2, 3A, 3B, 4A, 4B, and 5, when a user rotates the computing system 100 in the X axis direction, the rotation sensor 140 senses the rotation and outputs the generated sensing signal to the control signal generator 123 according to a result of the sensing. The control signal generator 123 outputs control signals to the CPU 113 in response to the sensing signal.

[98] The CPU 113 reads a control value, e.g., 2, stored in the control value table 113-1 shown in FIG. 18, and sets or programs the read control value in the SFR 116 included in the MMU 115.

[99] The control value is the number of page misses occurring when the computing system 100 pre-fetches data included in the first frame OFF of the original image after being rotated in the X-axis direction, i.e., the number of address generation rules to be generated in the MMU 115.

[100] Accordingly, the MMU 115 may perform a self-learning using differences between addresses related to two page misses, generate two address generation rules using a result of the self-learning, and pre-fetch each data from the external memory 120 based on the two address generation rules.

[101] That is, the self-learning denotes a process of generating address generation rules using page miss positions and differences between addresses related to the page misses.

[102] A box including a diagonal line of FIG. 4A indicates a page miss position. Accordingly, a page miss occurs in each address access order "4" and "16".

[103] A first frame OFF of the original image includes data "A" to "P", a second frame OSF of the original image includes data "Q" to "f", and each data "A" to "P" and "Q" to "f" is accessed or pre-fetched by the MMU 115 according to an address access order "1" to "32".

[104] First of all, the rotator 117 transmits an address ADD1=D13 for data "M" to the MMU 115. The MMU 115 transmits the address ADD1=D13 to the memory controller 119, and the memory controller 119 reads the data "M" corresponding to the address ADD1=D13 from a memory region of the external memory 120 and transmits the read data "M" to the MMU 115.

[105] While the MMU 115 transmits data DATA=M to the rotator 117, the MMU 115 estimates a next sequential address D14 after a current address D13, and pre-fetches data “N” corresponding to the estimated address D14 from a memory region of the external memory 120 to the image buffer 115-1 using the memory controller 119.

[106] Before an address generation rule is applied to the MMU 115, the MMU 115 estimates an address in a sequential manner.

[107] The rotator 117 transmits the address ADD1=D14 for data “N” to the MMU 115. The data “N” is pre-fetched to the image buffer 115-1 of the MMU 115, so that the MMU 115 transmits the data DATA= “N” to the rotator 117.

[108] While the data DATA=“N” is transmitted to the rotator 117, the MMU 115 estimates a next sequential address D15 after a current address D14, and pre-fetches data “O” stored in a memory region of the external memory 120 corresponding to the estimated address D15 to the image buffer 115-1 using the memory controller 119.

[109] The rotator 117 transmits an address ADD1=D15 for data “O” to the MMU 115. The data “O” is pre-fetched to the image buffer 115-1 of the MMU 115, so that the MMU 115 transmits the data DATA=O to the rotator 117.

[110] While the data DATA=“O” is transmitted to the rotator 117, the MMU 115 estimates a next sequential address D16 after the current address D15, and pre-fetches data “P” stored in a memory region of the external memory 120 corresponding to the estimated address D16 to the image buffer 115-1 using the memory controller 119.

[111] The rotator 117 transmits the address ADD1=D16 for data “P” to the MMU 115. The data “P” is pre-fetched to the image buffer 115-1 of the MMU 115, so that the MMU 115 transmits the data DATA=“P” to the rotator 117.

[112] While the data DATA=“P”is transmitted to the rotator 117, the MMU 115 estimates a next sequential address D17 after a current address D16, and pre-fetches data “Q” stored in a memory region of the external memory 120 corresponding to the estimated address D17 to the image buffer 115-1 using the memory controller 119.

[113] However, the rotator 117 transmits an address ADD1=D9 for data “I” to the MMU 115. Here, the data “I” is not present in the image buffer 115-1 of the MMU 115, and a first page miss occurs.

[114] The MMU 115 pre-fetches data “I” corresponding to the address ADD1=D9 from the external memory 120 to the image buffer 115-1 using the memory controller 119.

[115] Here, the MMU 115 stores a difference between the address D16 for the data “P” and the address D9 for the data “I”, i.e., a first depth value, e.g., “-7”, in an internal memory, e.g., 203-1 of FIG. 19. In addition, the MMU 115 stores a position where a first page miss occurs (hereinafter, “a first page miss position”), e.g., “4”, in the internal memory.

[116] That is, the MMU 115 performs a first self-learning using the first depth value and the first page miss position, and generates a first address generation rule using a result of the first self-learning.

[117] The MMU 115 may estimate at which position in each frame OFF and OSF of the original image a page miss occurs using the first page miss position. Accordingly, the MMU 115 may estimate each address using the first address generation rule in each position where a page miss occurs (e.g., 12, 8, and 4), and pre-fetch each data corresponding to each estimated address.

[118] While the data DATA="I" is transmitted to the rotator 117, the MMU 115 estimates a next sequential address D10 after the current address D9, and pre-fetches data "J" corresponding to the estimated address D10 from the external memory 120 to the image buffer 115-1 using the memory controller 119.

[119] A process of processing each data "J", "K", and "L" is substantially the same as a process of processing each data "N", "O", and "P". Accordingly, each data "J", "K", and "L" corresponding to each address ADD1=D10, D11, and D12, respectively, is pre-fetched by the MMU 115, and each pre-fetched data "J", "K", and "L" is transmitted to the rotator 117.

[120] While data DATA="L" is transmitted to the rotator 117, the MMU 115 pre-fetches data "E" corresponding to an address D5 estimated according to a first address generation rule to the image buffer 115-1 instead of data "M" corresponding to a next sequential address D13 after the current address D12. Accordingly, even if the address ADD1=D5 is input from the rotator 117, a page miss does not occur in the MMU 115.

[121] A process of processing each data "F", "G", and "H" is substantially the same as a process of processing each data "J", "K", and "L". Accordingly, each data "F", "G", and "H" corresponding to each address ADD1=D6, D7, and D8, respectively, is pre-fetched by the MMU 115, and each pre-fetched data "F", "G", and "H" is transmitted to the rotator 117.

[122] While the data DATA="H" is transmitted to the rotator 117, the MMU 115 pre-fetches data "A" corresponding to an address D1 estimated according to a first address generation rule, instead of pre-fetching data "I" corresponding to a next sequential address D9 after a current address D8, to the image buffer 115-1. Accordingly, even if the address ADD1=D1 is input from the rotator 117, a page miss does not occur in the MMU 115.

[123] A process of processing each data "B", "C", and "D" is substantially the same as the process of processing each data "F", "G", and "H". Accordingly, each data "B", "C", and "D" corresponding to each address ADD1=D2, D3, and D4 is pre-fetched by the MMU 115, and each pre-fetched data "B", "C", and "D" is transmitted to the rotator 117.

[124] However, while last data DATA="D" of the first frame OFF of the original image is transmitted to the rotator 117, the MMU 115 pre-fetches data, stored in a memory region corresponding to an address of a previous frame of the original image which is estimated according to the first address generation rule, to the image buffer 115-1.

[125] According to a seventeenth address access order 17, the rotator 117 transmits an address ADD1=D29 for data "c1" included in the second frame OSF of the original image to the MMU 115. Here, since the data "c1" required by the rotator 117 is not present in the image buffer 115-1 of the MMU 115, a second page miss occurs.

[126] The MMU 115 pre-fetches the data "c1" stored in a memory region of the external memory 120 corresponding to an address ADD1=D29 to the image buffer 15-1 using the memory controller 119.

[127] The MMU 115 stores a difference between an address D4 of the data "D" and an address D29 of the data "c1", i.e., a second depth value, e.g., "+25", in an internal memory, e.g., 203-1 of FIG. 19. In addition, the MMU 115 stores a position where a second page miss occurs (hereinafter, referred to as "a second page miss position"), e.g., "16", in the internal memory.

[128] That is, the MMU 115 performs a second self-learning by using addresses D4 and D29 related to a second page miss and a second page miss position, e.g., "16", and generates a second address generation rule using a result of the second self-learning.

[129] The MMU 115 may estimate at which position a page miss occurs by using the first page miss position, e.g., 4, and a second page miss position, e.g., 16. Accordingly, the MMU 115 may estimate each address by using a first address generation rule and a second address generation rule at each position 20, 24, 28, and 32 where a page miss occurs, and pre-fetch each data by using each estimated address.

[130] In a first frame of the original image, i.e., an initial frame OFF of a rotated image, two page misses occur.

[131] Accordingly, as illustrated in FIG. 4A, when pre-fetching data stored in a second frame OSF of the original image, a page miss does not occur in the MMU 115. As illustrated in FIG. 4B, in the first frame FF, the rotator 117 transmits each data output from the MMU 117 to the display 130 through the image processing module 118 and the display controller 121. Accordingly, an image processed according to an address access order 1 to 16, i.e., an X-axis rotated image FF, is displayed on the display 130.

[132] While data DATA="c1" pre-fetched to the image buffer 115-1 is transmitted to the rotator 117, the MMU 115 pre-fetches data "d" corresponding to a next sequential address D30 estimated according to a first address generation rule to the image buffer 115-1.

[133] While each data "f", "b", and "X" is transmitted to the rotator 117, the MMU 115 pre-fetches each data "Y", "U", and "Q" to the image buffer 115-1 according to a first address generation rule. Accordingly, even if each address ADD1=D25, D21, and D17 is input to the rotator 117, a page miss does not occur in the MMU 115.

[134] Referring to FIG. 2, while data DATA="T" is transmitted to the rotator 117, the MMU 115 pre-fetches data "s" corresponding to an address D45 estimated based on a second address generation rule to the image buffer 115-1. Accordingly, even if the address D45 is input to the rotator 117, a page miss does not occur in the MMU 115.

[135] As illustrated in FIG. 4B, in the second frame SF, the rotator 117 transmits each data output from the MMU 115 to the display 130 through the image processing module 118 and the display controller 121. Accordingly, an image processed according to the address access order "17" to "32", i.e., X-axis rotated image SF, is displayed on the display 130.

[136] As described above, the MMU 115 may perform a self-learning using a page miss position and a difference between addresses related to the page miss whenever a page miss occurs according to a control value stored in the SFR 116, generate address generation rules using a result of the self-learning, and pre-fetch data included in an original image to the image buffer 115-1 using the address generation rules.

[137] After the computing system 100 is rotated, page misses may occur at a process of pre-fetching data included in the first frame OFF of the original image. However, according to at least one example embodiment, data is pre-fetched according to address generation rules generated through a self-learning after the second frame OSF of the original image so that a page miss does not occur.

[138] The MMU 115 may reduce the number of page misses occurring according to an image rotation, thereby reducing data pre-fetch time and improving a performance of the computing device 10.

[139] Whenever the computing system 100 is rotated, the SFR 116 is set by a new control value, and page miss positions and address generation rules are initialized.

[140] FIGS. 6A and 6B each illustrate an address access order for displaying an image rotated in a Y-axis direction and a display image according to at least one example embodiment.FIG. 7 is a conceptual diagram for describing address generation rules for generating an image rotated in the Y-axis direction according to at least one example embodiment.

[141] Referring to FIGS. 1, 2, 3A, 3B, 6A, 6B, and 7, when a user rotates the computing system 100 in a Y-axis direction, the rotation sensor 140 senses the rotation, and outputs a sensing signal generated based on a result of the sensing to the control signal generator 123. The control signal generator 123 outputs control signals to the CPU 113 in response to the sensing signal.

[142] The CPU 113 reads a control value, e.g., 2, stored in a control value table 113-1 shown in FIG. 18, and sets or programs the read control value in the SFR 116 included in the

MMU 115 in response to the control signals.

[143] The control value indicates the number of page misses occurring when pre-fetching data included in the first frame OFF of the original image after the computing system 100 is rotated in the Y-axis direction, i.e., the number of address generation rules to be generated in the MMU 115.

[144] As described above, the MMU 115 performs a self-learning using two page miss positions and differences between addresses related to each of the two page misses, generates two address generation rules using a result of the self-learning, and pre-fetches each data included in each frame OFF and OSF of the original image from the external memory 120 based on the two address generation rules.

[145] The rotator 117 transmits an address ADD1=D4 for data "D" to the MMU 115. The MMU 115 pre-fetches the data "D" corresponding to the address ADD1=D4 from the external memory 120 to the image buffer 115-1 using the memory controller 119.

[146] While data DATA="D" is transmitted to the rotator 117, the MMU 115 estimates a next address D5, and pre-fetches data "E" corresponding to the estimated address D5 from the external memory 120 to the image buffer 115-1 using the memory controller 119.

[147] Afterwards, the rotator 117 transmits an address ADD1=D3 for data "C" to the MMU 115. Here, the data "C" is not present (or stored) in the image buffer 115-1 of the MMU 115, so that a first page miss occurs.

[148] The MMU 115 pre-fetches the data "C" corresponding to the address ADD1=D3 from the external memory 120 to the image buffer 115-1 using the memory controller 119.

[149] Here, the MMU 115 may store a difference between an address D4 for the data "D" and an address D3 for the data "C", i.e., a first depth value, e.g., "-1", in the internal memory 203-1 of FIG. 19. Moreover, the MMU 115 may store a first page miss position, e.g., "1", in the internal memory.

[150] The MMU 115 performs a first self-learning using addresses D4 and D3 related to the first page miss and the first page miss position, and generates a first address generation rule using a result of the first self-learning. The MMU 115 may estimate at which position in the first frame of the original image a page miss occurs using the first page miss position.

[151] Accordingly, the MMU 115 may estimate each address using the first address generation rule at each position, e.g., 5, 9, and 13, where a page miss occurs, and pre-fetch each data using the estimated address.

[152] While the data DATA="C" is transmitted to the rotator 117, the MMU 115 estimates a next address D2, and pre-fetches data "B" corresponding to the estimated address D2 to the image buffer 115-1. While the data DATA="B" is transmitted to the rotator 117, the MMU 115 estimates a next address D1, and pre-fetches data "A" corresponding to the estimated address D1 to the image buffer 115-1. While the data DATA="A" is transmitted to the rotator 117, the MMU 115 estimates an address of a previous frame, and pre-fetches data corresponding to the estimated address to the image buffer 115-1.

[153] The rotator 117 transmits an address ADD1=D8 for data "H" to the MMU 115. Here, the data "H"is not present in the image buffer 115-1 of the MMU 115, so that a second page miss occurs.

[154] The MMU 115 pre-fetches the data "H" corresponding to the address ADD1=D8 from the external memory 120 to the image buffer 115-1 using the memory controller 119.

[155] Here, the MMU 115 may store a difference between an address D1 for the data "A" and an address D8 for the data "H", i.e., a second depth value, e.g., "+7", in the internal memory 203-1 of FIG. 19. Moreover, the MMU 115 may store a second page miss position, e.g., "4", in the internal memory.

[156] The MMU 115 may estimate at which position, e.g., 5, 8, 9, 12, 13, and 16, in the first frame OFF of the original image a page miss occurs using the first page miss position, e.g., 1, and the second page miss position, e.g., 4.

[157] The MMU 115 performs a second self-learning using the second depth value and the second page miss position, and generates a second address generation rule using a result of the second self-learning. Accordingly, the MMU 115 may estimate each address using a first address generation rule and a second address generation rule at each position, e.g., 5, 8, 9, 12, 13, and 16, where a page miss occurs, and pre-fetch each data corresponding to each estimated address.

[158] As illustrated in FIG. 6B, in the first frame FF, the MMU 115 transmits each data pre-fetched from the external memory 120 to the image buffer 115-1 to the rotator 117 according to the address access order "1" to "16".

[159] Accordingly, each data output from the MMU 117 is transmitted to the display 130 through the image processing module 118 and the display controller 121. Accordingly, an image processed based on the address access order 1 to 16, i.e., Y-axis rotated image FF, is displayed on the display 130.

[160] While the pre-fetched data DATA="M" is transmitted to the rotator 117, the MMU 115 pre-fetches data "T" corresponding to a next address D20 to the image buffer 115-1 according to a second address generation rule. Accordingly, even if the address D20 is input, a page miss does not occur in the MMU 117.

[161] The MMU 117 pre-fetches data included in a second frame OSF of the original image based on the first address generation rule and the second address generation rule, so that a page miss does not occur in the MMU 115 from the second frame OSF of the original image.

[162] FIGS. 8A and 8B each illustrate an address access order for displaying a 90°-rotated image and a display image according to at least one example embodiment.FIG. 9 is a conceptual diagram for describing address generation rules for generating the 90°-rotated image according to at least one example embodiment.

[163] Referring to FIGS. 1, 2, 3A, 3B, 8A, 8B, and 9, when a user rotates the computing system 100 at 90°, the rotation sensor 140 senses the rotation, and outputs a sensing signal generated according to a result of the sensing to the control signal generator 123. The control signal generator 123 outputs control signals to the CPU 113 in response to the sensing signal.

[164] The CPU 113, in response to the control signals, reads a control value, e.g., 3, stored in the control value table 113-1 shown in FIG. 18 and sets or programs the read control value in the SFR 116 included in the MMU 115.

[165] The control value, e.g., 3, denotes the number of page misses occurring when pre-fetching data included in the first frame OFF of the original image, i.e., the number of address generation rules to be generated in the MMU 115, after the computing system 100 is rotated by 90°.

[166] The MMU 115 performs a self-learning using three page miss positions and differences of addresses related to three page misses, generates three address generation rules using a result of the self-learning, and pre-fetches data from the external memory 120 based on the three address generation rules.

[167] While data “M” is transmitted to the rotator 117, the MMU 115 pre-fetches data “N” corresponding to a next sequential address D14 after the address D13 from the external memory 120 to the image buffer 113-1. When an address ADD1=D9 corresponding to data “I” is input, a first page miss occurs in the MMU 115.

[168] Here, the MMU 115 calculates a difference between addresses D13 and D9, i.e., a first depth value, e.g., “-4”, and stores the first depth value and a first page miss position, e.g., “1”, in the internal memory 203-1 of FIG. 19. The MMU 115 generates a first address generation rule using the first depth value and the first page miss position.

[169] The MMU 115 pre-fetches each data “I”, “E”, and “A” corresponding to each address D9, D5, and D1 according to a first address generation rule, and transmits each pre-fetched data “I”, “E”, and “A” to the rotator 117.

[170] While the data “A” is transmitted to the rotator 117, the MMU 115 pre-fetches data corresponding to an address of a previous frame estimated according to a first address generation rule. When an address ADD1=D14 corresponding to data “N” is input, a second page miss occurs in the MMU 115.

[171] Here, the MMU 115 calculates a difference between addresses D1 and D14, i.e., a second depth value, e.g., “+13”, and stores the second depth value and a second page miss position, e.g., “4”, in the internal memory. The MMU 115 generates a second address generation rule using the second depth value and the second page miss position.

[172] The MMU 115 may estimate at which position in the first frame OFF of the original image a page miss will occur by using a first page miss position and a second page miss position. Accordingly, the MMU 115 may estimate each address by using the first page miss position and the second page miss position at each position 5, 8, 9, 12, 13, and 16 where a page miss will occur.

[173] According to the first address generation rule, the second address generation rule, and an address access order 5 to 16, the MMU 115 pre-fetches each data "N", "J", "F", "B", "O", "K", "G", "C", "P", "L", "H", and "D" and transmits each pre-fetched data "N", "J", "F", "B", "O", "K", "G", "C", "P", "L", "H", and "D" to the rotator 117.

[174] While the data "D" is transmitted to the rotator 117, the MMU 115 pre-fetches data "Q" corresponding to a next sequential address D17 after an address D4 from the external memory 120 to the image buffer 113-1. When an address ADD1=D29 corresponding to data "c1" is input, a third page miss occurs in the MMU 115.

[175] Here, the MMU 115 calculates a difference (referred to as "a third depth value") between addresses D4 and D29, e.g., "+25", and stores the third depth value and a third page miss position, e.g., "16", in the internal memory. The MMU 115 generates a third address generation rule using the third depth value and the third page miss position.

[176] The MMU 115 may estimate at which position in the second frame OSF of the original image a page miss will occur by using the first address generation rule to the third address generation rule.

[177] Accordingly, the MMU 115 pre-fetches each data corresponding to each estimated address from the external memory 120 to the image buffer 115-1, so that a page miss does not occur in a frame after the second frame OSF of the original image.

[178] As shown in FIG. 8B, the first frame FF includes each data pre-fetched based on the first address generation rule and the second address generation rule. One or more frames after the second frame SF includes each data pre-fetched based on the first address generation rule to the third address generation rule, so that a page miss does not occur.

[179] FIGS. 10A and 10B each illustrate an address access order for displaying a 180°-rotated image and a display image according to at least one example embodiment.FIG. 11 is a conceptual diagram for describing address generation rules for generating the 180°-rotated image according to at least one example embodiment.

[180] Referring to FIGS. 1, 2, 3A, 3B, 10A, 10B, and 11, when a user rotates the computing system 100 at 180°, the rotation sensor 140 senses the rotation and outputs a sensing signal generated according to a result of the sensing to the control signal generator 123. The control signal generator 123 outputs control signals to the CPU 113 in response to the sensing signal.

[181] The CPU 113, in response to the control signals, reads a control value, e.g., 2, stored in the control value table 113-1 shown in FIG. 18, and sets or programs the read control value in the SFR 116 included in the MMU 115.

[182] The control value, e.g., 2, denotes the number of page misses occurring when pre-fetching data included in the first frame OFF of the original image after the computing system 100 is rotated by 180°, i.e., the number of address generation rules to be generated in the MMU 115.

[183] The MMU 115 performs a self-learning using two page miss positions and differences between addresses related to two page misses, generates two address generation rules using a result of the self-learning, and pre-fetches data from the external memory 120 based on the two address generation rules.

[184] While data “P” is transmitted to the rotator 117, the MMU 115 pre-fetches data “Q” corresponding to a next sequential address D17 after an address D16 from the external memory 120 to the image buffer 113-1. When an address ADD1=D15 corresponding to data “O” is input, a first page miss occurs in the MMU 115.

[185] Here, the MMU 115 calculates a first difference between addresses D16 and D15, i.e., a first depth value, e.g., “-1”, and stores the first depth value and a first page miss position, e.g., “1”, in the internal memory 203-1 of FIG. 19. The MMU 115 generates a first address generation rule using the first depth value and the first page miss position.

[186] The MMU 115 may estimate at which position in the first frame OFF of the original frame a page miss will occur by using a first page miss position.

[187] The MMU 115 pre-fetches each data “O”, “N”, and “M” based on a first address generation rule, and transmits each pre-fetched data “O”, “N”, and “M” to the rotator 117.

[188] While data “M” is transmitted to the rotator 117, the MMU 115 pre-fetches data “L” based on the first address generation rule.

[189] The MMU 115 pre-fetches each data “L”, “K”, “J”, “I”, “H”, “G”, “F”, “E”, “D”, “C”, “B”, and “A” based on the first address generation rule, and transmits each pre-fetched data “L”, “K”, “J”, “I”, “H”, “G”, “F”, “E”, “D”, “C”, “B”, and “A” to the rotator 117.

[190] While last data “A” is transmitted to the rotator 117, the MMU 115 pre-fetches data corresponding to an address of a previous frame. When an address ADD1=D32 corresponding to data “f” is input, a second page miss occurs in the MMU 115.

[191] Here, the MMU 115 calculates a second difference between addresses D1 and D32, i.e., a second depth value, e.g., “+31”, and stores the second depth value and a second page miss position, e.g., “16”, in the internal memory 203-1 of FIG. 19. The MMU 115 generates a second address generation rule using the second depth value and the second page miss position.

[192] According to the first address generation rule and an address access order "17" to "32", the MMU 115 pre-fetches each data and transmits each pre-fetched data to the rotator 117.

[193] While the data "Q" is transmitted to the rotator 117, the MMU 115 pre-fetches data corresponding to an address of a next frame estimated according to a second address generation rule. Even if the address of the next frame is input, a page miss does not occur in the MMU 115.

[194] As illustrated in FIG. 10B, a first frame FF includes each data pre-fetched according to a first address generation rule. One or more frames after a second frame SF includes each data pre-fetched based on the first address generation rule and the second address generation rule, so that the page miss does not occur.

[195] FIGS. 12A and 12B each illustrate an address access order for displaying an image which is rotated by 90° and then rotated in the X-axis direction and a display image according to at least one example embodiment.FIG. 13 is a conceptual diagram for describing address generation rules for generating the image which is rotated by 90° and then rotated in the X-axis direction according to at least one example embodiment.

[196] Referring to FIGS. 1, 2, 3A, 3B, 12A, 12B, and 13, when a user rotates the computing system 100 at 90°, and then rotates it in a X-axis direction again, the rotation sensor 140 senses the rotation, and outputs a sensing signal generated according to a result of the sensing to the control signal generator 123. The control signal generator 123 outputs a control signal to the CPU 113 in response to the sensing signal.

[197] The CPU 113, in response to the control signals, reads a control value, e.g., 3, stored in the control value table 113-1 illustrated in FIG. 18, and set or program the read control value in the SFR 116 included in the MMU 115.

[198] The control value, e.g., 3, denotes the number of page misses occurring when pre-fetching data included in the first frame OFF of the original image after the computing system 100 is rotated by 90°-, and then is rotated again in the X-axis direction, i.e., the number of address generation rules to be generated in the MMU 115.

[199] The MMU 115 performs a self-learning using three page miss positions and differences between addresses related to three page misses, generates three address generation rules using a result of the self-learning, and pre-fetches each data from the external memory 120 according to the three address generation rules.

[200] While data "P" is transmitted to the rotator 117, the MMU 115 pre-fetches data "Q" corresponding to a next sequential address D17 after an address D16 to the image buffer 113-1. Accordingly, when an address ADD1=D12 corresponding to data "L" is input, a first page miss occurs in the MMU 115.

[201] Here, the MMU 115 calculates a difference between addresses D16 and D12, i.e., a first depth value, e.g., “-4”, and stores the first depth value and a first page miss position, e.g., “1”, in the internal memory.

[202] The MMU 115 generates a first address generation rule using the first depth value and the first page miss position. Accordingly, the MMU 115 pre-fetches each data “L”, “H”, and “D” according to an address access order and the first address generation rule, and transmits each pre-fetched data “L”, “H”, and “D” to the rotator 117.

[203] While data “D” is transmitted to the rotator 117, the MMU 115 estimates a next sequential address after an address D4, and pre-fetches data of a previous frame of an original frame based on a result of the estimation. Accordingly, when an address ADD1=D15 corresponding to data “O” is input, a second page miss occurs in the MMU 115.

[204] The MMU 115 calculates a difference between addresses D15 and D4, i.e., a second depth value, e.g., “+11”, and stores the second depth value and a second page miss position, e.g., “4”, in the internal memory. The MMU 115 generates a second address generation rule by using the second depth value and the second page miss position.

[205] The MMU 115 pre-fetches each data “O”, “K”, “G”, “C”, “N”, “J”, “F”, “B”, “M”, “I”, “E”, and “A” according to the first address generation rule, the second address generation rule, and the address access order, and transmits each pre-fetched data “O”, “K”, “G”, “C”, “N”, “J”, “F”, “B”, “M”, “I”, “E”, and “A” to the rotator 117.

[206] While last data “A” is transmitted to the rotator 117, the MMU 115 estimates a next sequential address D12 after a current address D1 according to the second address generation rule, and pre-fetches data “L” according to a result of the estimation. When an address ADD1=D32 is input, a third page miss occurs in the MMU 115.

[207] Here, the MMU 115 calculates a difference between addresses D1 and D32, i.e., a third depth value, e.g., “+31”, and stores the third depth value and a third page miss position, e.g., “16”, in the internal memory. The MMU 115 generates a third address generation rule using the third depth value and the third page miss position.

[208] As illustrated in FIG. 12B, a first frame FF includes each data pre-fetched according to the first address generation rule and the second address generation rule.

[209] The second frame SF and a subsequent frame include each data pre-fetched according to the first address generation rule to the third address generation rule, so that a page miss does not occur.

[210] FIGS. 14A and 14B each illustrate an address access order for displaying an image which is rotated by 90° and then rotated in the Y-axis direction and a display image, according to at least one example embodiment.FIG. 15 is a conceptual diagram for describing address generation rules for generating the image which is rotated by 90° and then rotated in the Y-axis direction, according to at least one example embodiment.

[211] Referring to FIGS. 1, 2, 3A, 3B, 14A, 14B, and 15, when a user rotates the computing system 100 by 90°, and then rotates it again in the Y-axis direction, the rotation sensor 140 senses the rotation, and outputs a sensing signal generated according to a result of the sensing to the control signal generator 123. The control signal generator 123 outputs control signals to the CPU 113 in response to the sensing signal.

[212] The CPU 113, in response to the control signals, reads a control value, e.g., 3, stored in the control value table 113-1 illustrated in FIG. 18, and sets or programs the read control value in the SFR 116 included in the MMU 115.

[213] The control value, e.g., 3, denotes the number of page misses occurring when pre-fetching data included in the first frame OFF of the original image after the computing system 100 is rotated by 90°, and then rotated again in the Y-axis direction, i.e., the number of address generation rules to be generated in the MMU 115.

[214] The MMU 115 may perform a self-learning by using three page miss positions and differences between addresses related to the three page misses, generate three address generation rules using a result of the self-learning, and pre-fetch each data from the external memory 120 according to the three address generation rules.

[215] While data “A” is transmitted to the rotator 117, the MMU 115 estimates a next sequential address D2 after a current address D1, and pre-fetches data “B” to the image buffer 113-1 according to a result of the estimation. When an address ADD1=D5 is input, a first page miss occurs in the MMU 115.

[216] Here, the MMU 115 calculates a difference between addresses D1 and D5, i.e., a first depth value, e.g., “+4”, and stores the first depth value and a first page miss position, e.g., “1”, in the internal memory. The MMU 115 generates a first address generation rule using the first depth value and the first page miss position. Accordingly, the MMU 115 pre-fetches each data “E”, “I”, and “M” according to an address access order and a first address generation rule, and transmits each pre-fetched data “E”, “I”, and “M” to the rotator 117.

[217] While data “M” is transmitted to the rotator 117, the MMU 115 estimates a next sequential address D17 after an address D13 based on the first address generation rule, and pre-fetches data “Q” according to a result of the estimation. When an address ADD1=D2 is input, a second page miss occurs in the MMU 115.

[218] The MMU 115 calculates a difference between addresses D13 and D2, i.e., a second depth value, e.g., “-11”, and stores the second depth value and a second page miss position in the internal memory. The MMU 115 generates a second address generation rule using the second depth value and the second page miss position.

[219] The MMU 115 pre-fetches each data “B”, “F”, “J”, “N”, “C”, “G”, “K”, “O”, “D”, "H", "L", and "P" and transmits each pre-fetched data "B", "F", "J", "N", "C", "G", "K", "O", "D", "H", "L", and "P" to the rotator 117 based on the first address generation rule, the second address generation rule, and the address access order.

[220] While last data "P" is transmitted to the rotator 117, the MMU 115 estimates a next sequential address D5 after a current address D16 according to the second address generation rule, and pre-fetches data "E" according to a result of the estimation. When an address ADD1=D17 is input, a third page miss occurs in the MMU 115.

[221] Here, the MMU 115 calculates a difference between addresses D16 and D17, i.e., a third depth value, e.g., "+1", and stores the third depth value and the third page miss position, e.g., "16", in the internal memory. The MMU 115 generates a third address generation rule by using the third depth value and the third page miss position.

[222] As illustrated in FIG. 12B, a first frame FF includes each data pre-fetched according to the first address generation rule and the second address generation rule.

[223] A second frame SF and a subsequent frame include each data pre-fetched according to the first to the third address generation rules, so that a page miss does not occur in the MMU 115.

[224] FIGS. 16A and 16B each illustrate an address access order for displaying an image which is rotated by 270° and a display image, according to at least one example embodiment.FIG. 17 is a conceptual diagram for describing address generation rules for generating the image which is rotated by 270°, according to at least one example embodiment.

[225] Referring to FIGS. 1, 2, 3A, 3B, 16A, 16B, and 17, when a user rotates the computing system 100 by 270°, the rotation sensor 140 senses the rotation, and outputs a sensing signal generated according to a result of the sensing to the control signal generator 123. The control signal generator 123 outputs control signals to the CPU 113 in response to the sensing signal.

[226] The CPU 113, in response to the control signals, reads a control value, e.g., 3, stored in the control value table 113-1 illustrated in FIG. 18, and sets or programs the read control value in the SFR 116 included in the MMU 115.

[227] The control value, e.g., 3, indicates the number of page misses occurring when pre-fetching data corresponding to a first frame after the computing system 100 is rotated by270°, i.e., the number of address generation rules to be generated in the MMU 115.

[228] The MMU 115 may perform a self-learning by using three page miss positions and differences between addresses related to the three page misses, generate three address generation rules using a result of the self-learning, and pre-fetch each data from the external memory 120 according to the three address generation rules.

[229] While data "D" is transmitted to the rotator 117, the MMU 115 estimates a next sequential address D5 after a current address D4, and pre-fetches data "E" to the image buffer

113-1 according to a result of the estimation. Accordingly, when an address ADD1=D8 is input, a first page miss occurs in the MMU 115.

[230] Here, the MMU 115 calculates a difference between addresses D4 and D8, i.e., a first depth value, e.g., “+4”, and stores the first depth value and a first page miss position 1 in the internal memory. The MMU 115 generates a first address generation rule by using the first depth value and the first page miss position. Accordingly, the MMU 115 pre-fetches each data “H”,“L”, and “P” according to an address access order and the first address generation rule, and transmits each pre-fetched data “H”, “L”, and “P” to the rotator 117.

[231] While the data “P” is transmitted to the rotator 117, the MMU 115 estimates a next sequential address D20 after an address D16 according to the first address generation rule, and pre-fetches data “T” according to a result of the estimation. When an address ADD1=D3 is input, a second page miss occurs in the MMU 115.

[232] The MMU 115 calculates a difference between addresses D16 and D3, i.e., a second depth value, e.g., “-13”, and stores the second depth value and a second page miss position, e.g., “4”, in the internal memory. The MMU 115 generates a second address generation rule by using the second depth value and the second page miss position.

[233] The MMU 115 pre-fetches each data “C”, “G”, “K”, “O”, “B”, “F”, “J”, “N”, “A”, “E”, “I”, and “M” based on the first address generation rule, the second address generation rule, and the address access order, and transmits each pre-fetched data “C”, “G”, “K”, “O”, “B”, “F”, “J”, “N”, “A”, “E”, “I”, and “M” to the rotator 117.

[234] While last data “M” is transmitted to the rotator 117, the MMU 115 estimates a next sequential address after a current address D13, i.e., an address of a previous frame, based on the second address generation rule, and pre-fetches data of the previous frame based on a result of the estimation. When an address ADD1=D20 is input, a third page miss occurs in the MMU 115. The MMU 115 pre-fetches data “T”.

[235] Here, the MMU 115 calculates a difference between addresses D16 and D17, i.e., a third depth value, e.g., “+7”, and stores the third depth value and the third page miss position, e.g., “16”, in the internal memory 203-1. The MMU 115 generates a third address generation rule by using the third depth value and the third page miss position.

[236] As illustrated in FIG. 16B, the first frame FF includes each data pre-fetched according to the first address generation rule and the second address generation rule.

[237] The second frame SF and a sequential frame include each data pre-fetched according to the first address generation rule to the third address generation rule, so that a page miss does not occur.

[238] FIG. 18 is an example embodiment of a control value table including the number of address generation rules determined according to a rotation direction.

[239] The control value table 113-1 includes a control value for each rotation direction. Each control value may also denote a depth value. Each depth value may be set by a manufacturer or a user.

[240] Control values included in the control value table 113-1 illustrated in FIG. 18 are exemplary values for describing the inventive concepts. According to at least one example embodiment, the control value table 113-1 may be embedded in an internal memory of the CPU 113, and loaded in the CPU 113 after being stored in a memory accessed by the CPU 113, and referred by the CPU 113 after being stored in an additional memory.

[241] FIG. 19 is a block diagram of the memory management unit illustrated in FIG. 1. Referring to FIGS. 1 to 19, the memory management unit 115 includes an address conversion circuit 201, a page miss detection circuit 203, a page table 205, an address generation rule generation circuit 207, a pre-fetch circuit 209, the image buffer 115-1, and the SFR 116.

[242] The address conversion circuit 201 performs a function of converting the address ADD1 output from the rotator 117. For example, the address conversion circuit 201 may be embodied in a translation look aside buffer (TLB).

[243] In the same manner as a general TLB, the TLB 201 generates a TLB hit TLB_HIT or a TLB miss TLB_MISS. For example, when a TLB miss TLB_MISS is generated, an address to be converted is acquired from the page table 205, and the acquired address is cached or TLB-written TLB_WRin the TLB 201.

[244] The page miss detection circuit 203 may count the number of the address ADD1 output from the rotator 117 using a counter, calculate a page miss position based on a result of the counting and a page miss PAGE_MISS, store a result of the calculation in the internal memory 203-1, and transmit the stored result of calculation to the address generation rule generation circuit 207.

[245] The address generation rule generation 207 estimates an address of a memory region of the external memory 120 where data to be pre-fetched is stored based on a control value DEPIN stored in the SFR 116, at least one page miss position output from the page miss detection circuit 203, and a converted address(e.g., an address generated according to TLB hit TLB_HIT, an address generated according to TLB miss TLB_MISS, or an address generated according to page miss PAGE_MISS), and generates the estimated address ADD2.

[246] For example, the address ADD1 may include a first address and an offset, and the estimated address ADD2 may include a second address and the offset.

[247] For example, the first address may be a virtual address, and the second address may be a physical address corresponding to the virtual address.

[248] When TLB hit TLB_HIT is generated, the address generation rule generation circuit 207 generates the estimated address ADD2 using an offset and an address output from the

TBL 201.

[249] When TLB miss TLB_MISS is generated, the address generation rule generation circuit 207 generates the estimated address ADD2 using an offset and an address acquired from the page table 205.

[250] When the page miss PAGE_MISS is generated, the pre-fetch circuit 209 pre-fetches data DATA_pre to the image buffer 115-1 using the address ADD2 generated according to the TLB miss TLB_MISS, and the address ADD2 for the pre-fetched data is cached or table-written TABLE_WR in the page table 205.

[251] The pre-fetch circuit 209 receives the estimated address ADD2, and transmits the estimated address ADD2 to the memory controller 119.The memory controller 119 reads data from the external memory 120 based on the address ADD2 output from the pre-fetch circuit 209, and transmits the read data DATA_pre, i.e., the pre-fetched data, to the pre-fetch circuit 209.

[252] The pre-fetch circuit 209 transmits the pre-fetched data DATA_pre to the image buffer 115-1. The image buffer 115-1 transmits the pre-fetched data DATA_pre to the rotator 117 as data DATA.

[253] The SFR 116 stores a control value DEPIN transmitted from the CPU 113. The address generation rule generation circuit 207 adjusts the number of address generation rules according to the control value DEPIN.

[254] Referring to FIGS. 4A, 4B, and 19, an operation of the MMU 115 is exemplarily described. For convenience of description in FIGS. 4A and 4B, it is assumed that an address access order is the same as a page miss position.

[255] Whenever an address D13, D14, D15, and D16 for each data “M”, “N”, “O”, and “P” is required, the page miss detection circuit 203 increases the count value. In FIGS. 4A and 4B, it is assumed that the count value is the same as an address access order. For example, the page miss detection circuit 203 may perform a function of a counter which may detect a position where a page miss occurs using the count value.

[256] When the first page miss occurs, the page miss detection circuit 203 calculates a first page miss position based on the count value. Here, the first page miss position is “4”, and the “4” is transmitted to the address generation rule generation circuit 207 after being stored in the internal memory 203-1.

[257] The address generation rule generation circuit 207 calculates a first depth value, e.g., “-7”, and generates a first address generation rule based on the first depth value and a first page miss position.

[258] For example, the address generation rule generation circuit 207 may estimate a next address using the first depth value, e.g., “-7”, at a position, e.g., “8”, “12”, and “16”, corresponding to the first page miss position, and generate the estimated address ADD2.

[259] The page miss detection circuit 203 increases a count value until an address D4 for data “D” is input.

[260] When a second page miss occurs, the page miss detection circuit 203 calculates a second page miss position based on the count value. Here, a second page miss position is “16”, and the “16” is transmitted to the address generation rule generation circuit 207 after being stored in the internal memory 203-1.

[261] The address generation rule generation circuit 207 calculates a second depth value, e.g., “+25”, and generates a second address generation rule based on the second depth value and the second page miss position.

[262] As illustrated in FIG. 4A, the address generation rule generation circuit 207 generates the estimated address ADD2 according to the first address generation rule and the second address generation rule, so that, when pre-fetching data included in the second frame OSF of the original image, a page miss does not occur.

[263] A method of operating the MMU 115 per rotation direction is substantially the same as a method described referring to FIGS. 4A, 4B, and 19, and as such, a detailed description for these is omitted.

[264] FIG. 20 is a flowchart for describing a method of rotating the original image according to at least one example embodiment.

[265] The computing device 110 recognizes (or determines) rotation or a rotation direction of the computing device 110, and outputs control signals corresponding to a result of the recognition to the CPU 113. The CPU 113 programs a control value store in the control value table 113-1 in the SFR 116 of the MMU 115 based on the control signals (S110).

[266] The MMU 115 performs a self-learning using addresses related to at least one page miss(S120).

[267] The MMU 115generates address generation rules using a result of the self-learning, pre-fetches the original image, i.e., data, from the external memory 120, and transmits the pre-fetched data to the rotator 117 (S140). That is, the computing device 110 pre-fetches data included in the original image from the external memory 120 based on the address generation rules, and generates a rotated image using the pre-fetched data (S140).

[268] The computing device 110 displays the rotated image through the display 130 (S150). When the computing system 110 rotates in a specific direction, i.e., when a direction in which the original image rotates is changed, the MMU 116 initiates the address generation rules previously generated.

[269] A method according to an example embodiment of the inventive concepts and a device performing the method may perform a self-learning using addresses related to a plurality of page misses occurring according to a rotation direction, generates address generation rules using a result of the self-learning, pre-fetches the original image from a memory device based on the address generation rules, and generate the rotated image using the pre-fetched image.

[270] Accordingly, the method and/or the device may reduce an occurrence frequency of page miss. Accordingly, the method and the device may reduce a delay of data processing time depending on an occurrence of the page miss, thereby improving performance of the device.

[271] While inventive concepts have been particularly shown and described with reference to exampleembodiments thereof, it will be understood by those of ordinary skill in the art that various changes in forms and details may be made therein without departing from the scope of inventive concepts as defined by the following claims.

CONCLUSIES

1. Werkwijze voor het roteren van een oorspronkelijk beeld, waarbij de werkwijze omvat:

het uitvoeren van een zelfleren gebruikmakend van een stap van vooraf ophalen en gebruikmakend van adressen die betrekking hebben op ten minste een paginamisser die optreedt als resultaat van de stap van vooraf ophalen;

het genereren van adresinformatieregels die een resultaat van het zelfleren gebruiken;

het vooraf ophalen van het oorspronkelijke beeld uit een geheugeninrichting op basis van de adresgeneratieregels voor het verkrijgen van een vooraf opgehaald beeld; en

het genereren van een geroteerd beeld gebruikmakend van het vooraf opgehaalde beeld,

waarbij de werkwijze verder omvat:

bepalen van een aantal adresgeneratieregels, te genereren op basis van een besturingswaarde opgeslagen in een geheugen voor een richting waarin het oorspronkelijke beeld dient te worden geroteerd.

2. Werkwijze volgens conclusie 1, waarbij de genererende adresgeneratieregels de adresgeneratieregels genereren op basis van een besturingswaarde die is ingesteld in een speciaal functieregister (SFR).

3. Werkwijze volgens conclusie 1, waarbij het vooraf ophalen van het oorspronkelijke beeld het vooraf ophalen van het oorspronkelijke beeld uit de geheugeninrichting naar een beeldbuffer op basis van de adresgeneratieregels voor het verkrijgen van het vooraf opgehaalde beeld omvat.

4. Werkwijze volgens conclusie 1, verder omvattende:

het programmeren, door een fabrikant of een gebruiker, van een aantal te genereren adresgeneratieregels.

5. Werkwijze volgens conclusie 1, verder omvattende:

het bepalen van een aantal te genereren adresgeneratieregels op basis van een besturingswaarde opgeslagen in een geheugen en representatief voor een aantal paginamissers die optreden in een initieel raster van het geroteerde beeld voor een richting waarin het oorspronkelijke beeld dient te worden geroteerd.

6. Werkwijze volgens conclusie 1, waarbij het zelfleren wordt uitgevoerd op basis van een positie van de ten minste ene paginamisser en een verschil tussen de adressen dat betrekking heeft op de ten minste ene paginamisser.

7. Werkwijze volgens conclusie 1, verder omvattende:

het initialiseren van de adresgeneratieregels wanneer een richting waarin het oorspronkelijke beeld wordt geroteerd, wordt gewijzigd.

8. Werkwijze volgens conclusie 1, waarbij de adressen die betrekking hebben op ten minste een paginamisser corresponderen met een positie van de ten minste ene paginamisser die wordt berekend op basis van een resultaat van het tellen van een nummer van een adres gebruikmakend van een teller.

9. Systeem op chip (SoC), omvattende:

een geheugenmanagementeenheid (MMU) die is geconfigureerd voor:

het uitvoeren van een zelfleren gebruikmakend van vooraf ophalen en gebruikmakend van adressen die zijn gerelateerd met ten minste een paginamisser die optreedt als resultaat van het vooraf ophalen,

het bepalen van een aantal adresgeneratieregels, te genereren op basis van een besturingswaarde opgeslagen in een geheugen voor een richting waarin het oorspronkelijke beeld dient te worden geroteerd,

het genereren van adresgeneratieregels gebruikmakend van een resultaat van het zelfleren, en

het vooraf ophalen van een oorspronkelijk beeld uit een geheugeninrichting naar een beeldbuffer op basis van de adresgeneratieregels; en

een rotator die is geconfigureerd voor het genereren van een geroteerd beeld gebruikmakend van een beeld dat vooraf is opgehaald naar het beeldbuffer.

10. SoC volgens conclusie 9, waarbij de MMU is geconfigureerd is voor het bepalen van een aantal van de te genereren adresgeneratieregels overeenkomstig een aantal paginamissers die optreden in een initieel raster van het geroteerde beeld.

11. SoC volgens conclusie 9, waarbij de MMU omvat:

een speciale-functieregister (SFR) dat is geconfigureerd voor het opslaan van een besturingswaarde, een adresgeneratieregelgeneratieschakeling die is geconfigureerd voor het genereren van de adresgeneratieregels op basis van de besturingswaarde die is opgeslagen in het SFR, en

een vooraf-ophaalschakeling die is geconfigureerd voor het vooraf ophalen van het oorspronkelijke beeld naar het beeldbuffer overeenkomstig de adresgeneratieregels.

12. SoC volgens conclusie 9, verder omvattende:

een centrale verwerkingseenheid (CPU) die is geconfigureerd voor het besturen van een aantal van de door de MMU te genereren adresgeneratieregels in reactie op besturingssignalen die een richting aangeven waarin het oorspronkelijke beeld wordt geroteerd.

13. Toepassingsprocessor omvattende de SoC volgens conclusie 9.

14. Toepassingsprocessor volgens conclusie 13, waarbij de MMU is geconfigureerd voor het bepalen van een aantal van de te genereren adresgeneratieregels overeenkomstig een aantal paginamissers die optreden in een initieel raster van het geroteerde beeld.

15. Toepassingsprocessor volgens conclusie 13, waarbij de MMU omvat:

een speciale-functieregister (SFR) dat is geconfigureerd voor het opslaan van een besturingswaarde;

een adresgeneratieregelgeneratieschakeling die is geconfigureerd voor het genereren van de adresgeneratieregels op basis van de besturingswaarde die is opgeslagen in het SFR; en

een vooraf-ophaalschakeling die is geconfigureerd voor het vooraf ophalen van het oorspronkelijke beeld naar het beeldbuffer overeenkomstig de adresgeneratieregels.

16. Mobiele inrichting, omvattende:

een geheugeninrichting die is geconfigureerd voor het opslaan van een oorspronkelijk beeld;

een geheugenmanagementeenheid (MMU) die is geconfigureerd voor:

het uitvoeren van een zelfleren gebruikmakend van vooraf ophalen en gebruikmakend van adressen die betrekking hebben op ten minste een paginamisser die optreedt als resultaat van het vooraf ophalen;

het bepalen van een aantal adresgeneratieregels, te genereren op basis van een besturingswaarde opgeslagen in een geheugen en representatief voor een aantal paginamissers die optreden in een initieel raster van het geroteerde beeld voor een richting waarin het oorspronkelijke beeld dient te worden geroteerd;

het genereren van adresgeneratieregels gebruikmakend van een resultaat van het zelfleren; en

het vooraf ophalen van een oorspronkelijk beeld uit de geheugeninrichting naar een beeldbuffer overeenkomstig de adresgeneratieregels;

een rotator die is geconfigureerd voor het genereren van een geroteerd beeld gebruikmakend van het beeld dat vooraf is opgehaald naar het beeldbuffer; en

een weergavebesturingsorgaan dat is geconfigureerd voor het overdragen van het door de rotator gegenereerde geroteerde beeld naar een weergave-eenheid.

17. Mobiele inrichting volgens conclusie 16, verder omvattende:

een centrale verwerkingseenheid (CPU) die is geconfigureerd voor het besturen van een aantal van de door de MMU te genereren adresgeneratieregels in reactie op besturingssignalen die een richting aangeven waarin het oorspronkelijke beeld dient te worden geroteerd.

18. Mobiele inrichting volgens conclusie 17, verder omvattende:

een besturingssignaalgenerator die is geconfigureerd voor het detecteren van de richting waarin het oorspronkelijke beeld dient te worden geroteerd en het genereren van de besturingssignalen overeenkomstig een resultaat van de detectie.

19. Mobiele inrichting volgens conclusie 17, verder omvattende:

een rotatiesensor die is geconfigureerd voor het detecteren van de richting waarin het oorspronkelijke beeld dient te worden geroteerd; en

een besturingssignaalgenerator die is geconfigureerd voor het genereren van de besturingssignalen in reactie op een meetsignaal afgegeven vanuit de rotatiesensor.

20. Mobiele inrichting volgens conclusie 16, waarbij de MMU omvat:

een speciale-functieregister (SFR) dat is geconfigureerd voor het opslaan van een besturingswaarde;

een adresgeneratieregelgeneratieschakeling die is geconfigureerd voor het genereren van de adresgeneratieregels op basis van de besturingswaarde die is opgeslagen in het SFR, en

een vooraf-ophaalschakeling die is geconfigureerd voor het vooraf ophalen van het oorspronkelijke beeld naar het beeldbuffer overeenkomstig de adresgeneratieregels.

21. Mobiele inrichting, omvattende:

een weergave-eenheid die is geconfigureerd voor het weergeven van een geroteerd beeld; en

een berekeningsinrichting die is geconfigureerd voor:

het detecteren van een richting van rotatie van de inrichting,

het genereren van adresgeneratieregels gebruikmakend van zelfleren, waarbij het zelfleren wordt uitgevoerd gebruikmakend van een vooraf ophalen en gebruikmakend van adressen die betrekking hebben op ten minste een paginamisser die optreedt als resultaat van het vooraf ophalen, waarbij de generatie van de adresgeneratieregels is gebaseerd op de gedetecteerde richting van rotatie, waarbij een nummer van te genereren adresgeneratieregels is gebaseerd op een besturingswaarde opgeslagen in een geheugen voor een richting waarin het oorspronkelijke beeld dient te worden geroteerd;

het genereren van het geroteerde beeld gebruikmakend van een oorspronkelijk beeld overeenkomstig de adresgeneratieregels, zodanig dat een paginamisser niet optreedt in het geroteerde beeld, en

het overdragen van het geroteerde beeld naar de weergave-eenheid.

22. Mobiele inrichting volgens conclusie 21, waarbij de berekeningsinrichting is geconfigureerd voor het genereren van de adresgeneratieregels op basis van:

een positie van paginamissers die zijn geassocieerd met de gedetecteerde richting van rotatie; en

een verschil tussen adressen die betrekking hebben op de paginamissers.

23. Mobiele inrichting volgens conclusie 22, waarbij de berekeningsinrichting is geconfigureerd voor het genereren van het geroteerde beeld door het vooraf ophalen van het oorspronkelijke beeld uit een geheugeninrichting naar een beeldbuffer op basis van de adresgeneratieregels.

FIG. 1

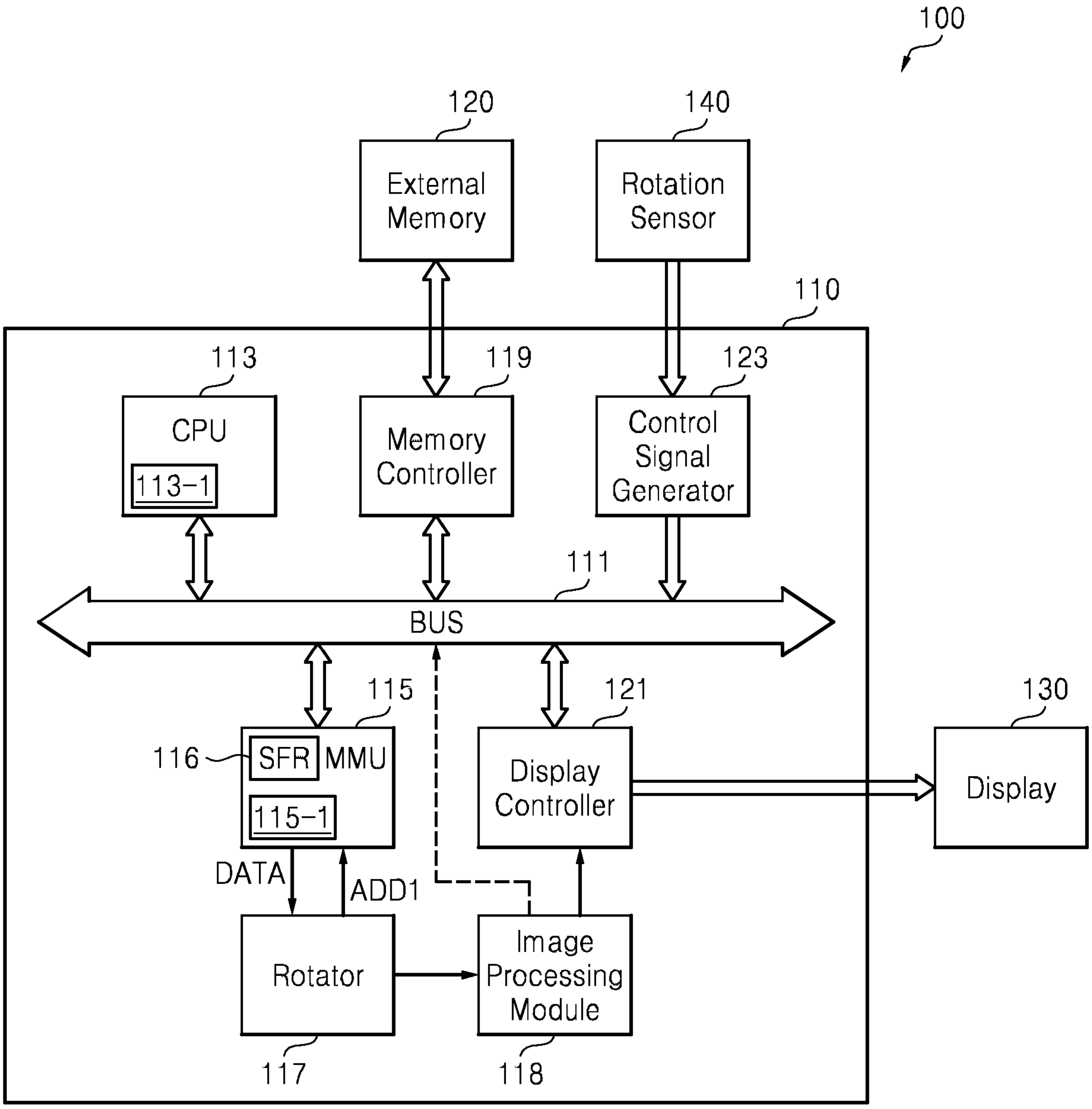
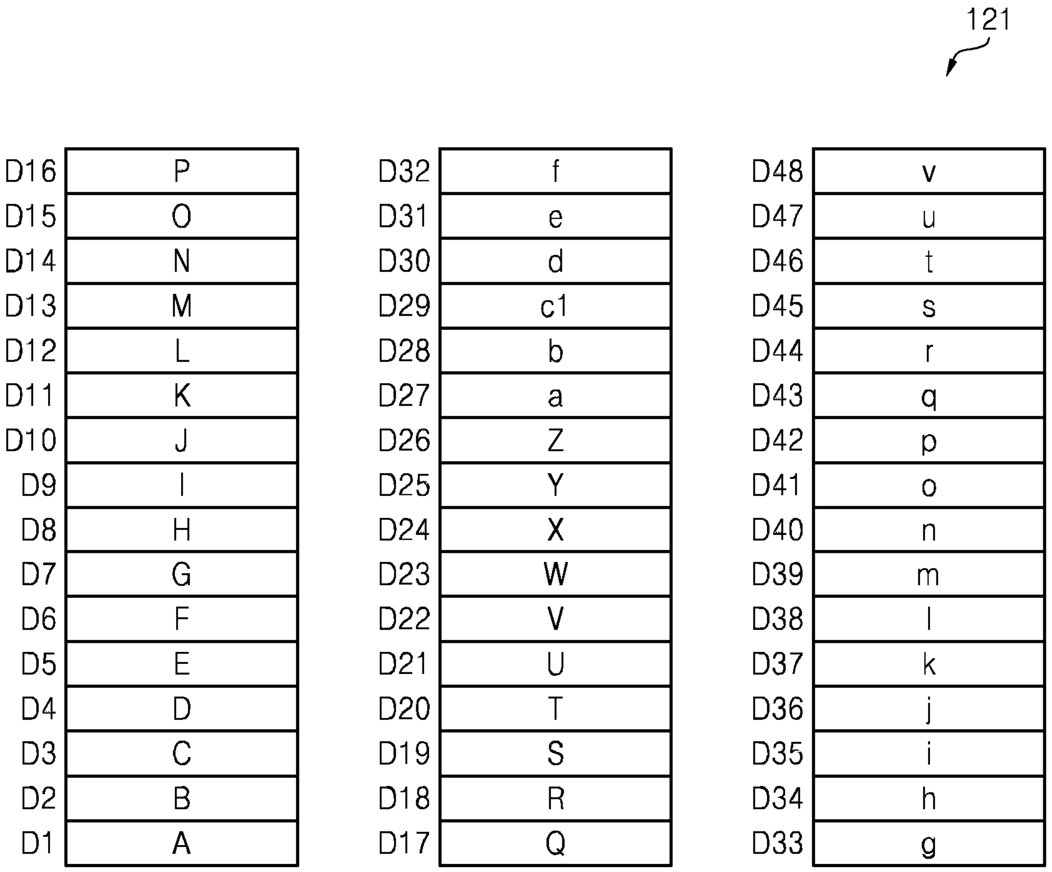
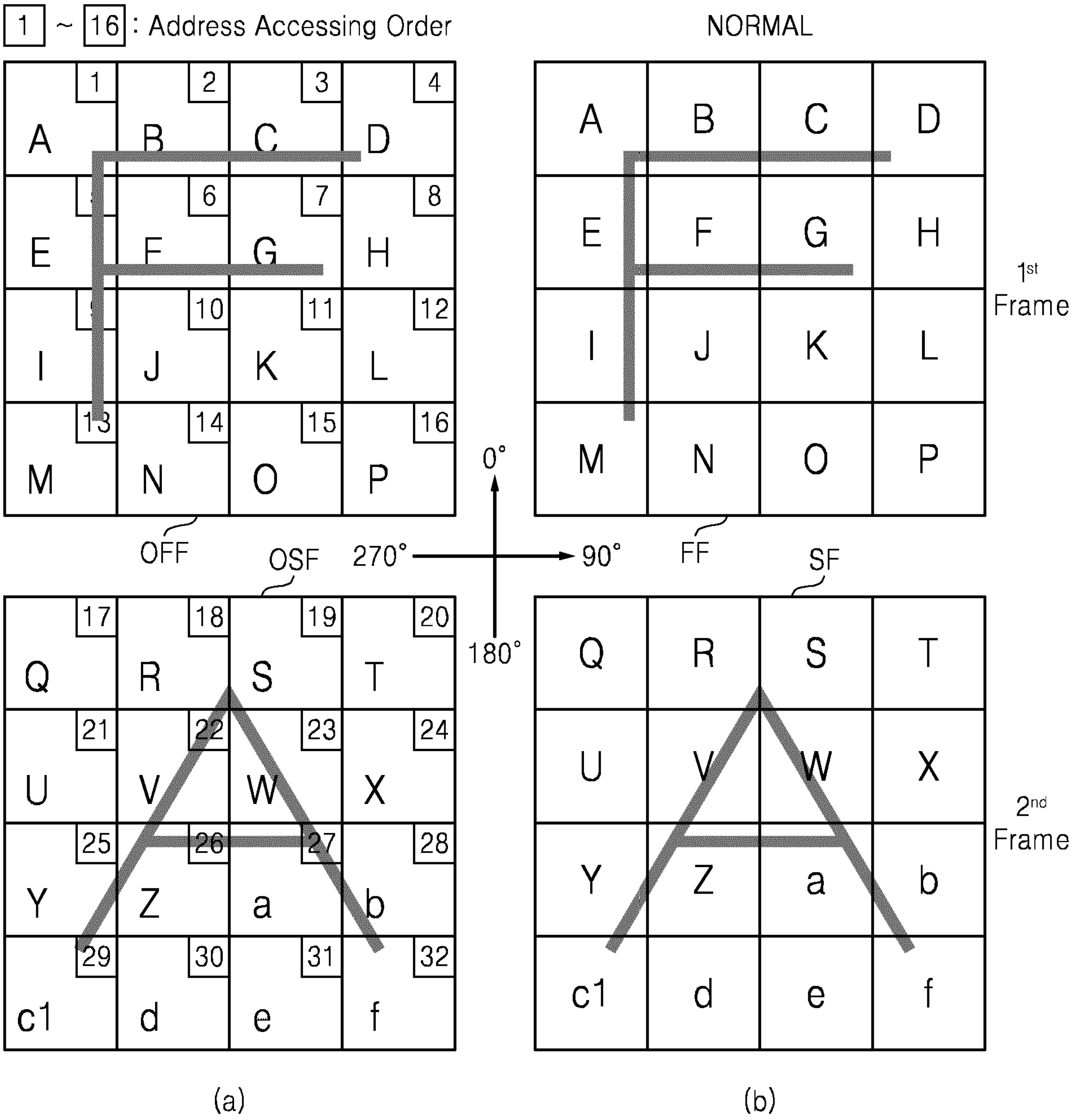

100
120
External Memory
140
Rotation Sensor
110
113
CPU
113-1
119
Memory Controller
123
Control Signal Generator
111
BUS
115
116
SFR
MMU
115-1
121
Display Controller
130
Display
DATA
ADD1
Rotator
Image Processing Module
117
118

FIG. 2



D16	P	D32	f	D48	v
D15	O	D31	e	D47	u
D14	N	D30	d	D46	t
D13	M	D29	c1	D45	s
D12	L	D28	b	D44	r
D11	K	D27	a	D43	q
D10	J	D26	Z	D42	p
D9	I	D25	Y	D41	o
D8	H	D24	X	D40	n
D7	G	D23	W	D39	m
D6	F	D22	V	D38	l
D5	E	D21	U	D37	k
D4	D	D20	T	D36	j
D3	C	D19	S	D35	i
D2	B	D18	R	D34	h
D1	A	D17	Q	D33	g

FIG. 3



1 ~ 16 : Address Accessing Order
NORMAL
1
2
3
4
A
B
C
D
5
6
7
8
E
F
G
H
9
10
11
12
I
J
K
L
13
14
15
16
M
N
O
P
1st Frame
0°
270°
90°
180°
OFF
FF
OSF
SF
17
18
19
20
Q
R
S
T
21
22
23
24
U
V
W
X
25
26
27
28
Y
Z
a
b
29
30
31
32
c1
d
e
f
2nd Frame
(a)
(b)

FIG. 4

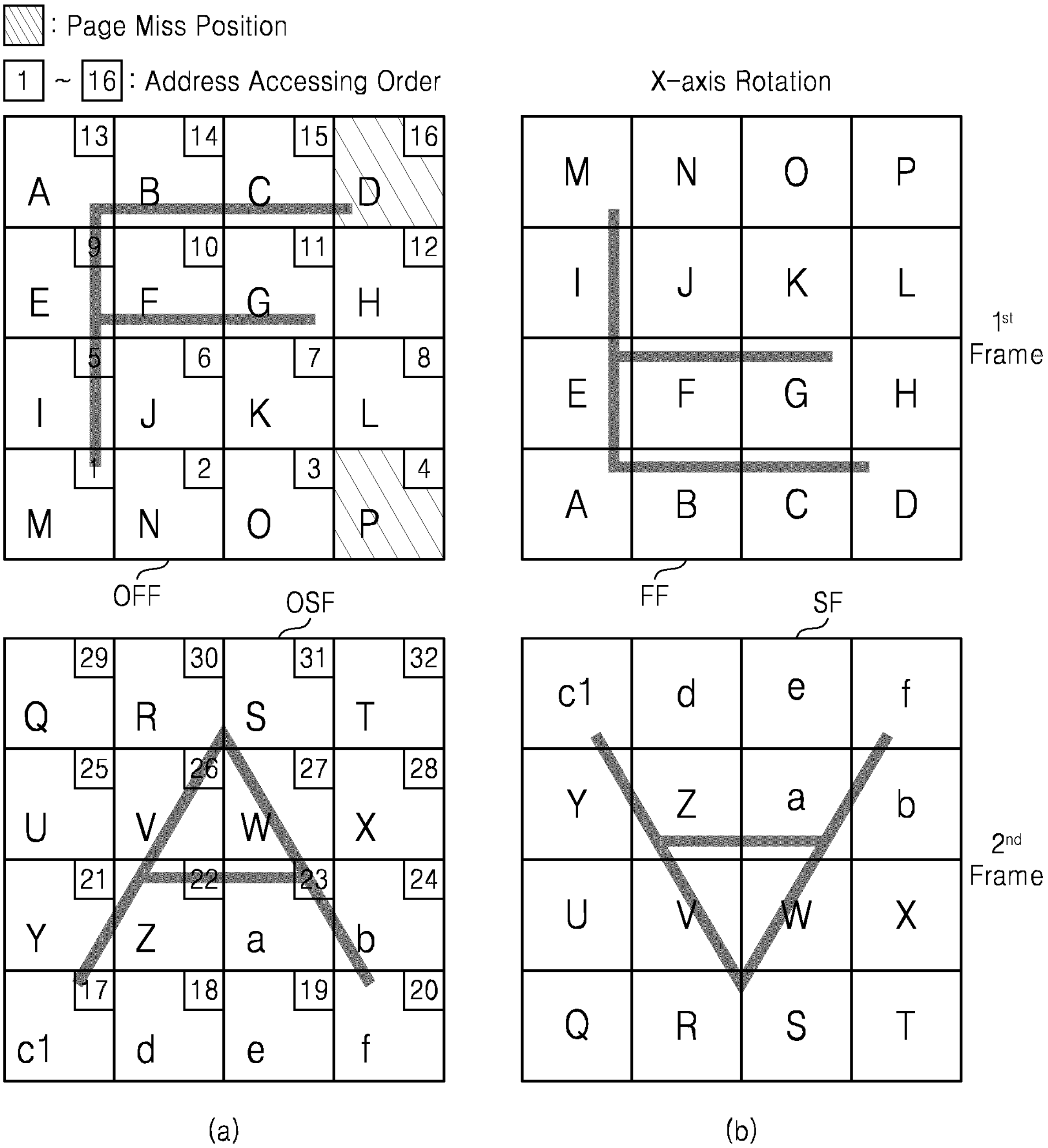
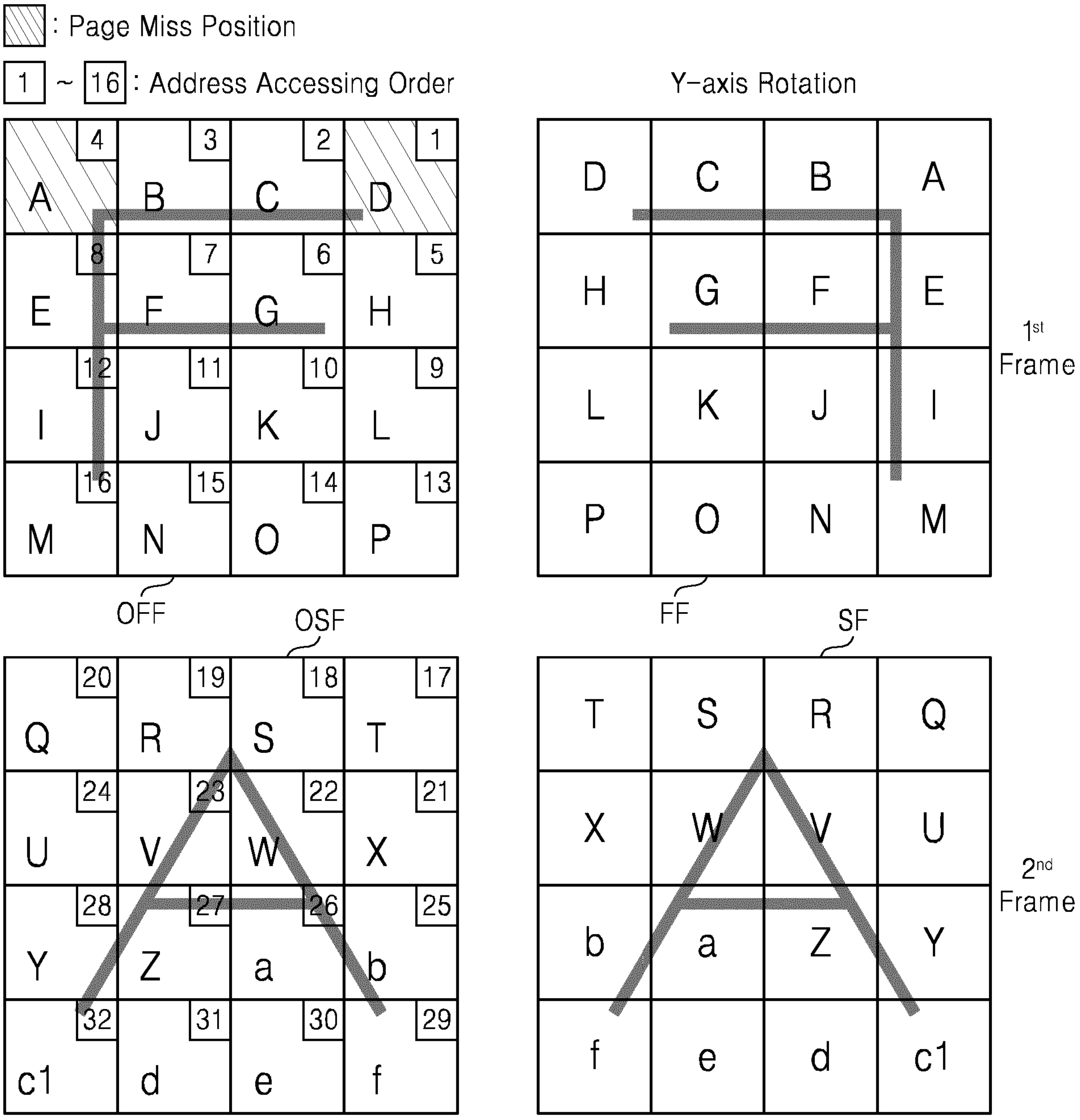

▨ : Page Miss Position
1 ~ 16 : Address Accessing Order
X-axis Rotation
13 14 15 16
A B C D
9 10 11 12
E F G H
5 6 7 8
I J K L
1 2 3 4
M N O P
OFF
OSF
29 30 31 32
Q R S T
25 26 27 28
U V W X
21 22 23 24
Y Z a b
17 18 19 20
c1 d e f
M N O P
I J K L
E F G H
A B C D
FF
SF
c1 d e f
Y Z a b
U V W X
Q R S T
1st Frame
2nd Frame
(a)
(b)

FIG. 5

Address Generation Rule #	Page Miss Position	Address Difference (Depth Value)
1	4	−7(=D16⇒D9)
2	16	+25(=D4⇒D29)
3	−	−

FIG. 6


: Page Miss Position
1 ~ 16 : Address Accessing Order
Y-axis Rotation
4 A 3 B 2 C 1 D
8 E 7 F 6 G 5 H
12 I 11 J 10 K 9 L
16 M 15 N 14 O 13 P
D C B A
H G F E
L K J I
P O N M
1st Frame
OFF
OSF
FF
SF
20 Q 19 R 18 S 17 T
24 U 23 V 22 W 21 X
28 Y 27 Z 26 a 25 b
32 c1 31 d 30 e 29 f
T S R Q
X W V U
b a Z Y
f e d c1
2nd Frame

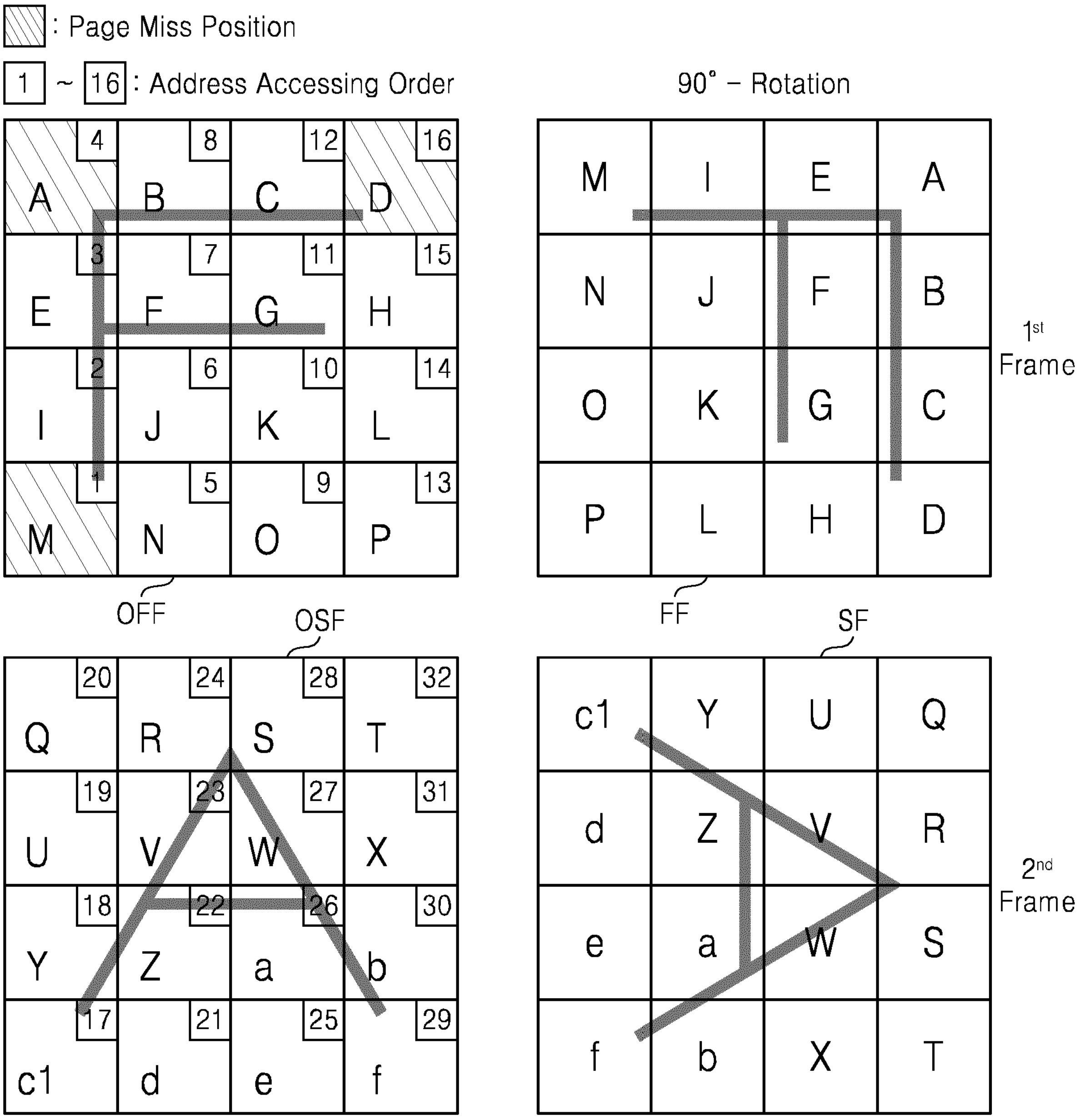
(a)

(b)

FIG. 7

Address Generation Rule #	Page Miss Position	Address Difference (Depth Value)
1	1	−1(=D4⇒D3)
2	4	+7(=D1⇒D8)
3	−	−

FIG. 8


▨ : Page Miss Position
1 ~ 16 : Address Accessing Order
90° - Rotation
A 4
B 8
C 12
D 16
E 3
F 7
G 11
H 15
I 2
J 6
K 10
L 14
M 1
N 5
O 9
P 13
OFF
OSF
Q 20
R 24
S 28
T 32
U 19
V 23
W 27
X 31
Y 18
Z 22
a 26
b 30
c1 17
d 21
e 25
f 29
M I E A
N J F B
O K G C
P L H D
1st Frame
FF
SF
c1 Y U Q
d Z V R
e a W S
f b X T
2nd Frame

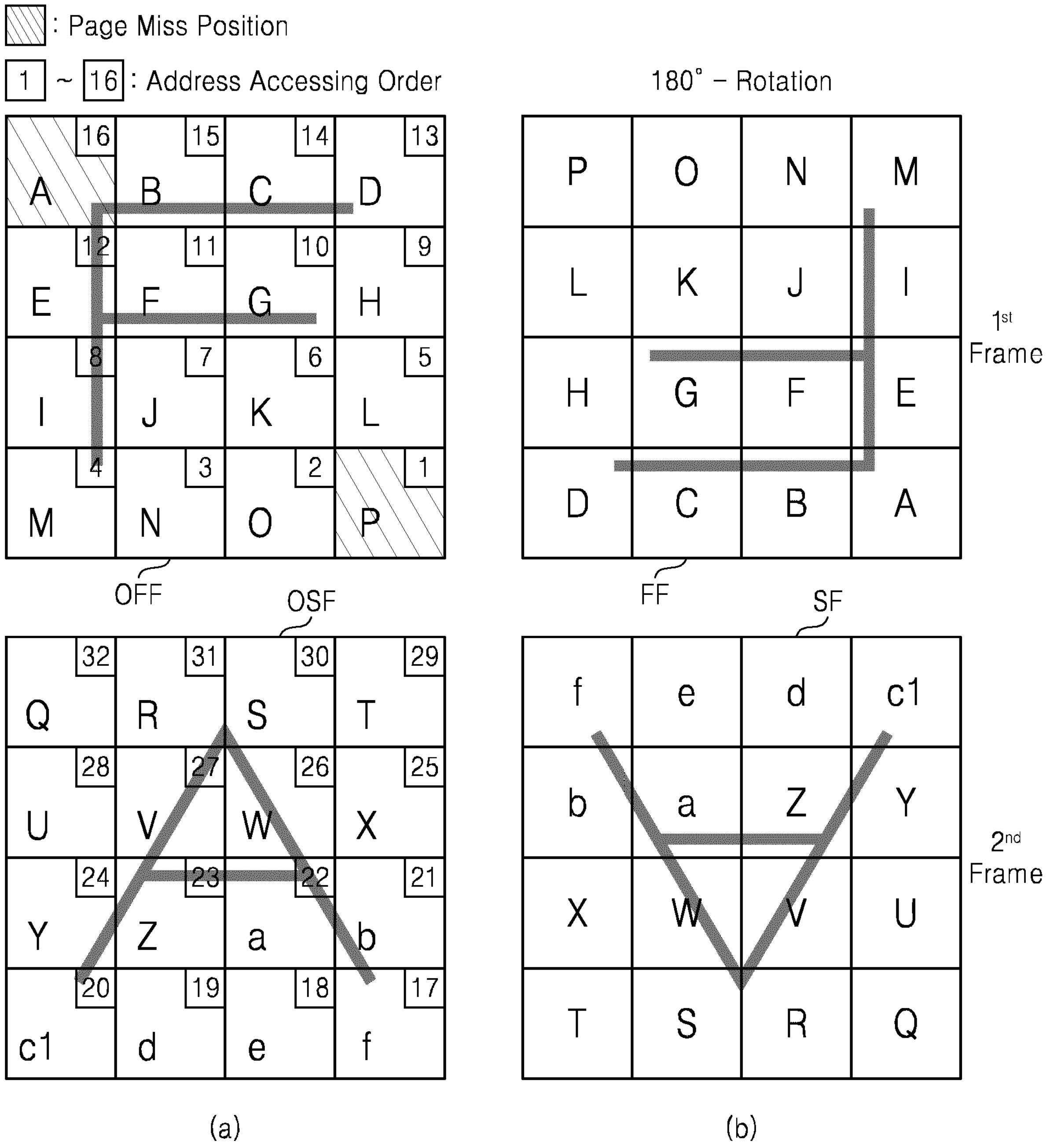
(a)

(b)

FIG. 9

Address Generation Rule #	Page Miss Position	Address Difference (Depth Value)
1	1	−4(=D13⇒D9)
2	4	+13(=D1⇒D14)
3	16	+25(=D4⇒D29)

FIG. 10



▨ : Page Miss Position
1 ~ 16 : Address Accessing Order
180° – Rotation
16 A
15 B
14 C
13 D
12 E
11 F
10 G
9 H
8 I
7 J
6 K
5 L
4 M
3 N
2 O
1 P
OFF
OSF
32 Q
31 R
30 S
29 T
28 U
27 V
26 W
25 X
24 Y
23 Z
22 a
21 b
20 c1
19 d
18 e
17 f
P O N M
L K J I
H G F E
D C B A
1st Frame
FF
SF
f e d c1
b a Z Y
X W V U
T S R Q
2nd Frame

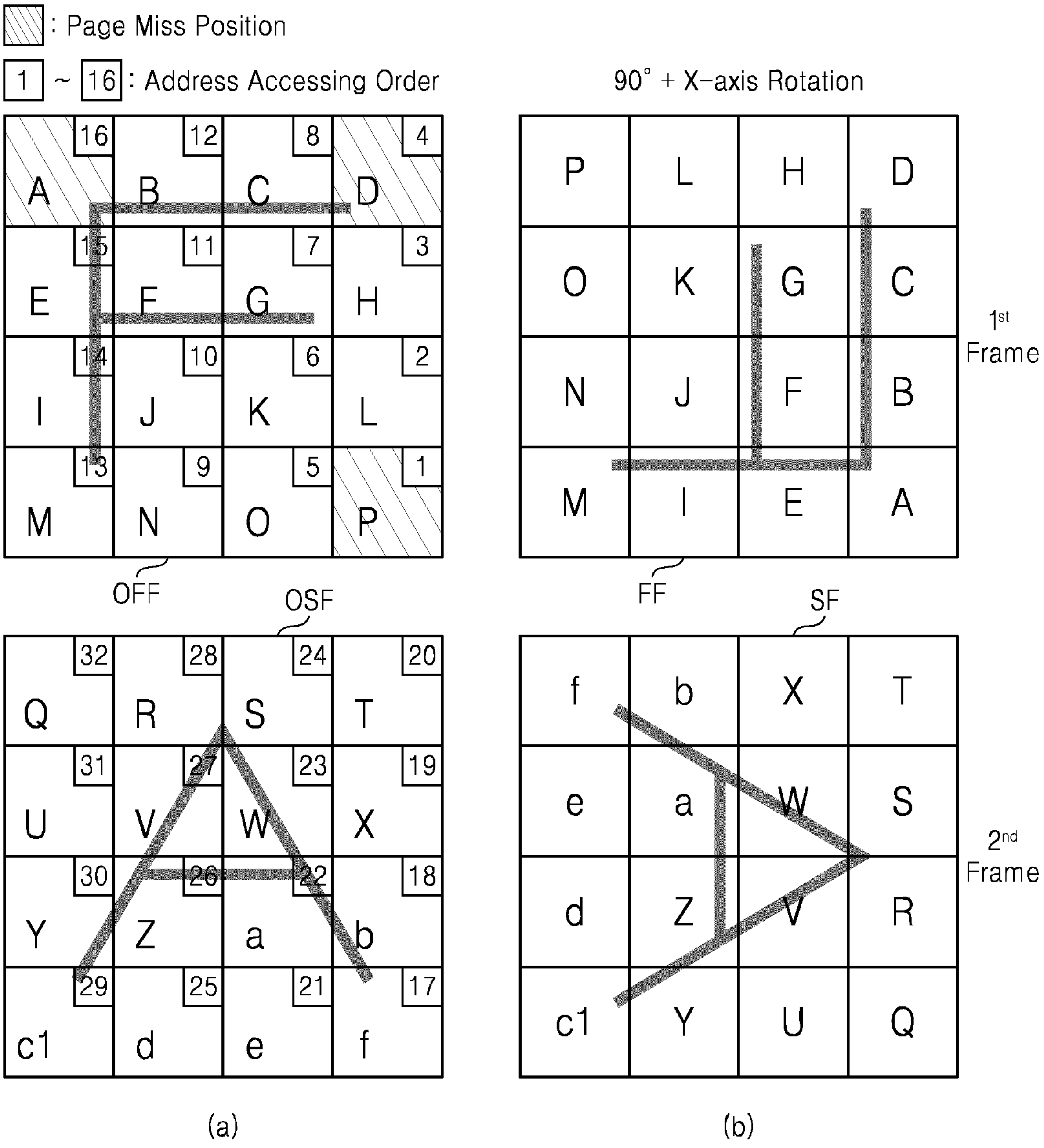
(a)

(b)

FIG. 11

Address Generation Rule #	Page Miss Position	Address Difference (Depth Value)
1	1	−1(=D16⇒D15)
2	16	+31(=D1⇒D32)

FIG. 12

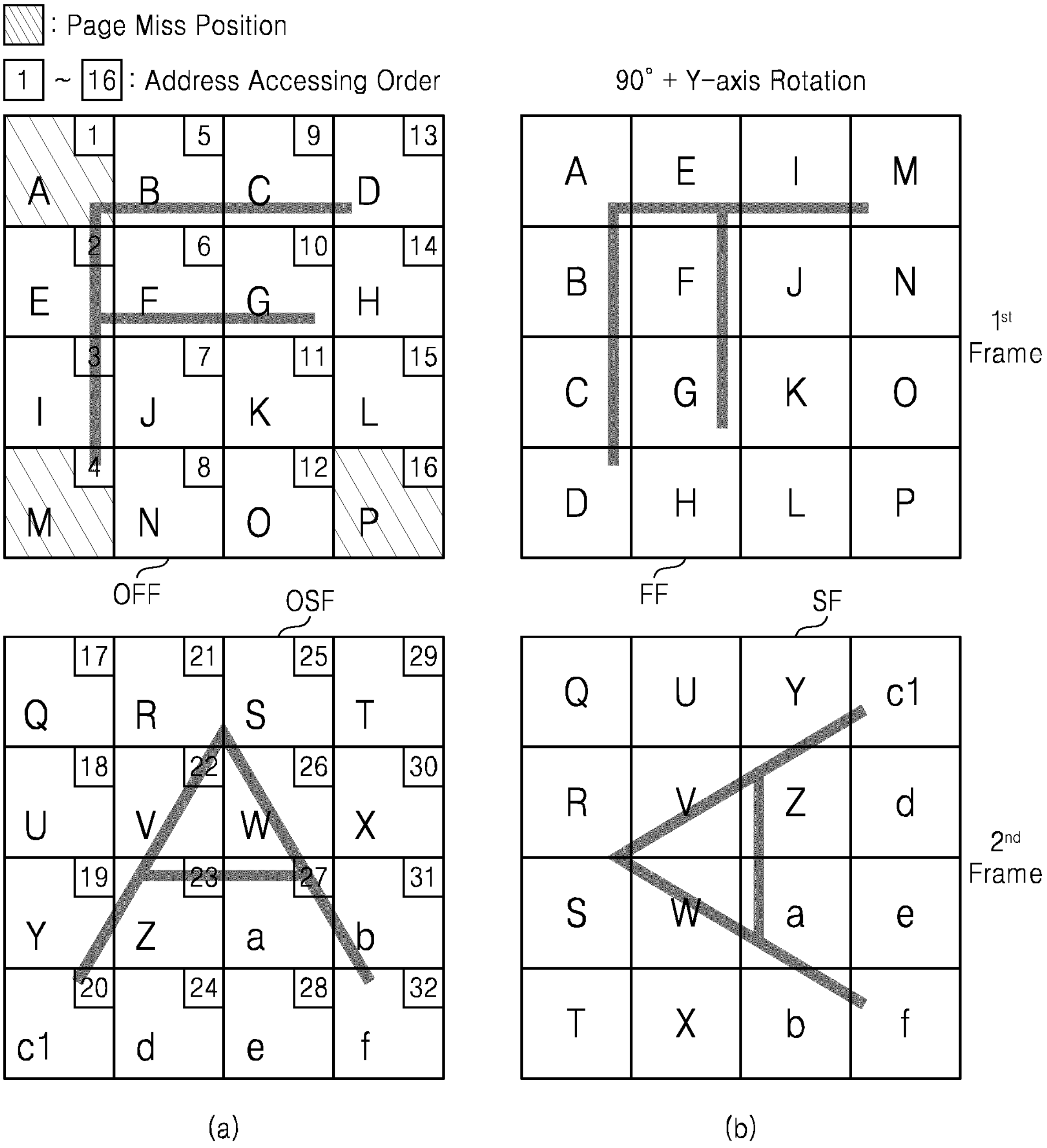


▨ : Page Miss Position
1 ~ 16 : Address Accessing Order
90° + X-axis Rotation
16 A 12 B 8 C 4 D
15 E 11 F 7 G 3 H
14 I 10 J 6 K 2 L
13 M 9 N 5 O 1 P
OFF
OSF
32 Q 28 R 24 S 20 T
31 U 27 V 23 W 19 X
30 Y 26 Z 22 a 18 b
29 c1 25 d 21 e 17 f
P L H D
O K G C
N J F B
M I E A
1st Frame
FF
SF
f b X T
e a W S
d Z V R
c1 Y U Q
2nd Frame
(a)
(b)

FIG. 13

Address Generation Rule #	Page Miss Position	Address Difference (Depth Value)
1	1	−4(=D16⇒D12)
2	4	+11(=D4⇒D15)
3	16	+31(=D1⇒D32)

FIG. 14

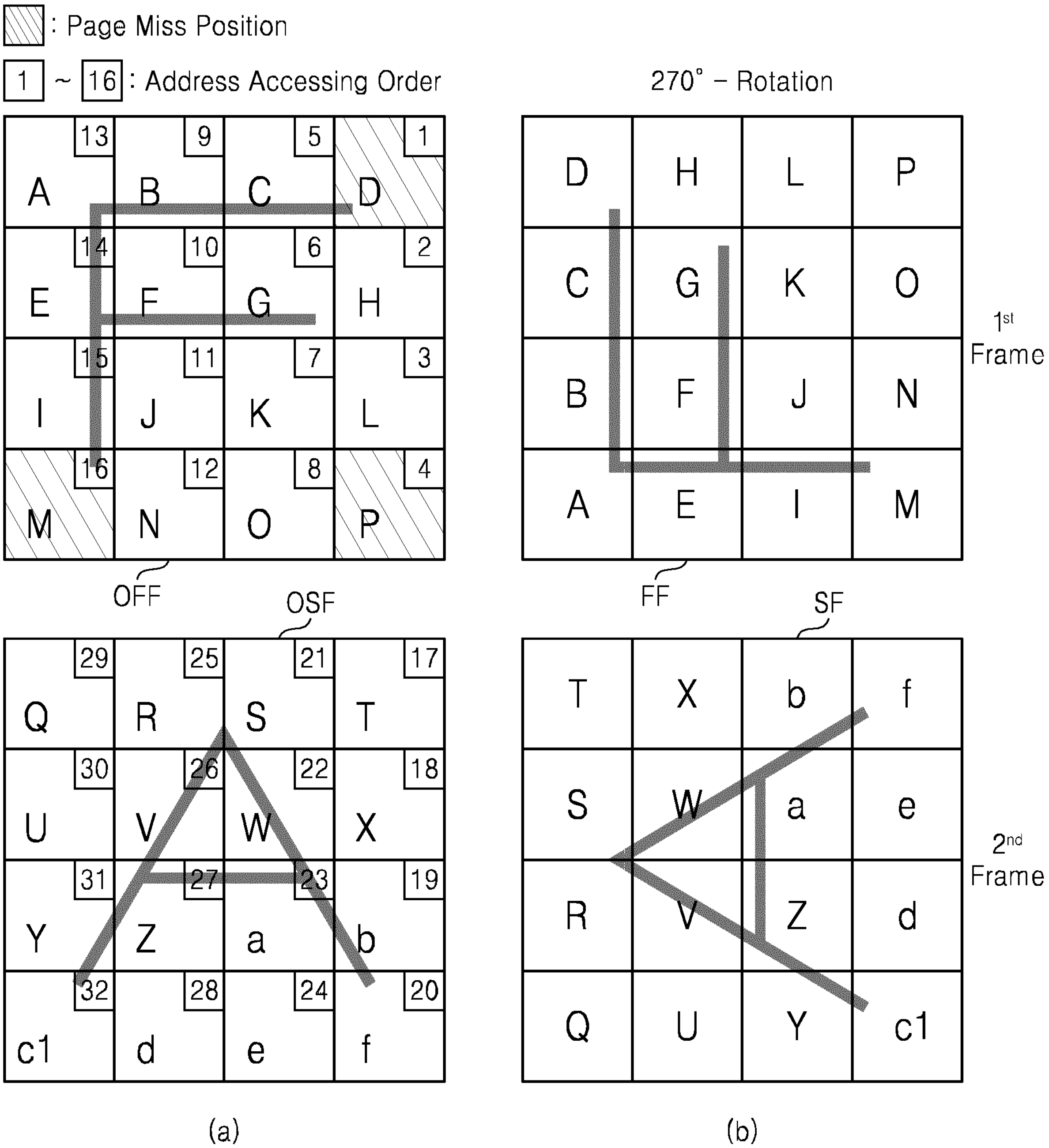


▨ : Page Miss Position
1 ~ 16 : Address Accessing Order
90° + Y-axis Rotation
A 1
B 5
C 9
D 13
E 2
F 6
G 10
H 14
I 3
J 7
K 11
L 15
M 4
N 8
O 12
P 16
OFF
OSF
Q 17
R 21
S 25
T 29
U 18
V 22
W 26
X 30
Y 19
Z 23
a 27
b 31
c1 20
d 24
e 28
f 32
A E I M
B F J N
C G K O
D H L P
FF
SF
1st Frame
Q U Y c1
R V Z d
S W a e
T X b f
2nd Frame
(a)
(b)

FIG. 15

Address Generation Rule #	Page Miss Position	Address Difference (Depth Value)
1	1	+4(=D1⇒D5)
2	4	−11(=D13⇒D2)
3	16	+1(=D16⇒D17)

FIG. 16



▨ : Page Miss Position
1 ~ 16 : Address Accessing Order
270° − Rotation
13 A 9 B 5 C 1 D
14 E 10 F 6 G 2 H
15 I 11 J 7 K 3 L
16 M 12 N 8 O 4 P
OFF
OSF
29 Q 25 R 21 S 17 T
30 U 26 V 22 W 18 X
31 Y 27 Z 23 a 19 b
32 c1 28 d 24 e 20 f
D H L P
C G K O
B F J N
A E I M
1st Frame
FF
SF
T X b f
S W a e
R V Z d
Q U Y c1
2nd Frame

(a)

(b)

FIG. 17

Address Generation Rule #	Page Miss Position	Address Difference (Depth Value)
1	1	+4(=D4⇒D8)
2	4	−13(=D16⇒D3)
3	12	+7(=D13⇒D20)

FIG. 18

113-1

Rotation Direction	Control Value
X-axis	2
Y-axis	2
90°	3
180°	2
90° + X-axis	3
90° + Y-axis	3
270°	3

FIG. 19

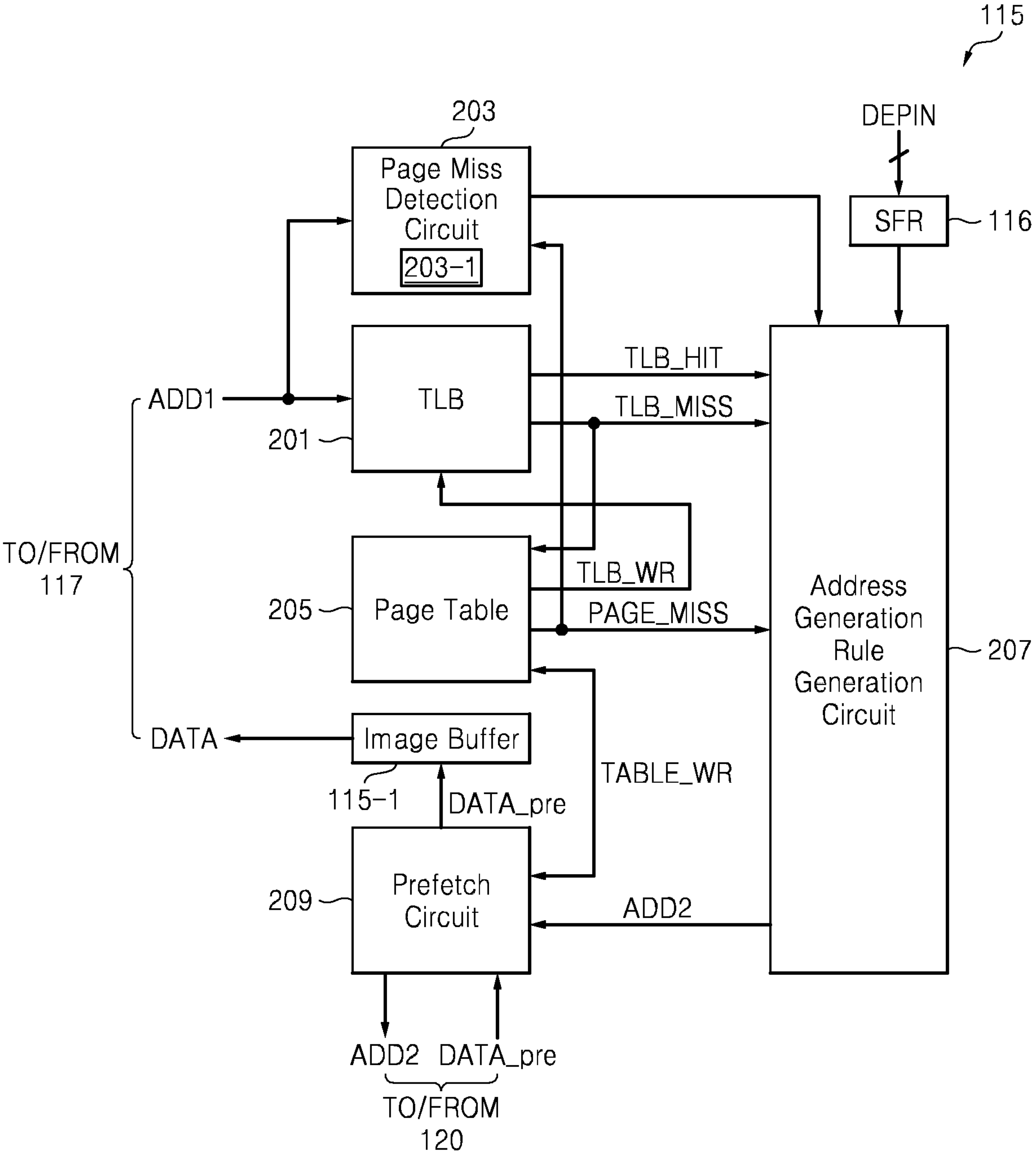
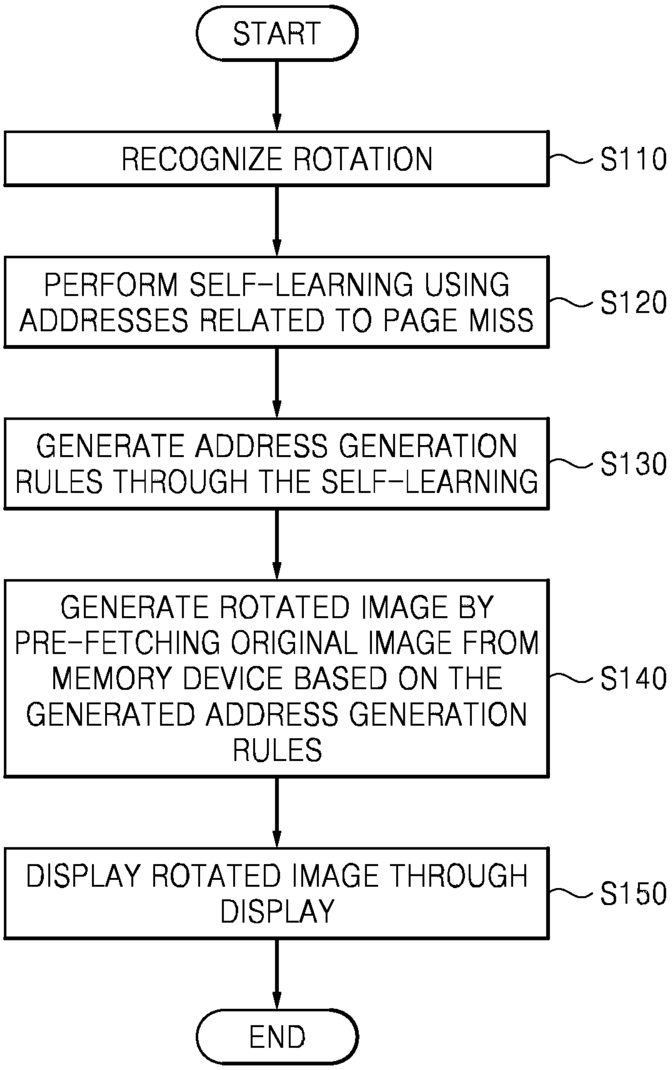

115
203
Page Miss Detection Circuit
203-1
DEPIN
SFR
116
TLB
TLB_HIT
TLB_MISS
ADD1
201
TO/FROM 117
Page Table
205
TLB_WR
PAGE_MISS
Address Generation Rule Generation Circuit
207
DATA
Image Buffer
115-1
DATA_pre
TABLE_WR
Prefetch Circuit
209
ADD2
ADD2
DATA_pre
TO/FROM 120

FIG. 20


START
RECOGNIZE ROTATION
S110
PERFORM SELF-LEARNING USING ADDRESSES RELATED TO PAGE MISS
S120
GENERATE ADDRESS GENERATION RULES THROUGH THE SELF-LEARNING
S130
GENERATE ROTATED IMAGE BY PRE-FETCHING ORIGINAL IMAGE FROM MEMORY DEVICE BASED ON THE GENERATED ADDRESS GENERATION RULES
S140
DISPLAY ROTATED IMAGE THROUGH DISPLAY
S150
END

Agentschap NL
Ministerie van Economische Zaken

OCTROOIAANVRAAG NR.:
NO 138915
NL 2012319

ONDERZOEKSRAPPORT

BETREFFENDE HET RESULTAAT VAN HET ONDERZOEK NAAR DE STAND VAN DE TECHNIEK

RELEVANTE LITERATUUR

Categorie[1]	Literatuur met, voor zover nodig, aanduiding van speciaal van belang zijnde tekstgedeelten of figuren.	Van belang voor conclusie(s) nr:	Classificatie (IPC)
X	US 6 633 968 B2 (ZWIEGINCEW ARTHUR [US] ET AL) 14 oktober 2003 (2003-10-14) * regel 66 - kolom 8, regel 9 * -----	1-30	INV. G06T1/60
X	US 5 305 389 A (PALMER MARK L [US]) 19 april 1994 (1994-04-19) * kolom 2, regel 13 - regel 37 * * kolom 5, regel 54 - regel 56 * -----	1-30	
X	VANDERWIEL S P ET AL: "Data prefetch mechanisms", ACM COMPUTING SURVEYS, ACM, NEW YORK, NY, US, US, deel 32, nr. 2, 1 juni 2000 (2000-06-01), bladzijden 174-199, XP002977351, ISSN: 0360-0300 * bladzijde 186, rechter kolom, alinea 2 - bladzijde 191, rechter kolom, alinea 1 * -----	1-30	
A	US 6 560 693 B1 (PUZAK THOMAS R [US] ET AL) 6 mei 2003 (2003-05-06) * kolom 7, regel 33 - regel 39; conclusie 1 * -----	1-30	Onderzochte gebieden van de techniek G06T
A	EP 0 318 778 A2 (IBM [US]) 7 juni 1989 (1989-06-07) * samenvatting * ----- -/--	1-30	

Indien gewijzigde conclusies zijn ingediend, heeft dit rapport betrekking op de conclusies ingediend op:

Plaats van onderzoek: München	Datum waarop het onderzoek werd voltooid: 24 november 2014	Bevoegd ambtenaar: Pierfederici, A

[1] CATEGORIE VAN DE VERMELDE LITERATUUR

X: de conclusie wordt als niet nieuw of niet inventief beschouwd ten opzichte van deze literatuur
Y: de conclusie wordt als niet inventief beschouwd ten opzichte van de combinatie van deze literatuur met andere geciteerde literatuur van dezelfde categorie, waarbij de combinatie voor de vakman voor de hand liggend wordt geacht
A: niet tot de categorie X of Y behorende literatuur die de stand van de techniek beschrijft
O: niet-schriftelijke stand van de techniek
P: tussen de voorrangsdatum en de indieningsdatum gepubliceerde literatuur
T: na de indieningsdatum of de voorrangsdatum gepubliceerde literatuur die niet bezwarend is voor de octrooiaanvrage, maar wordt vermeld ter verheldering van de theorie of het principe dat ten grondslag ligt aan de uitvinding
E: eerdere octrooi(aanvrage), gepubliceerd op of na de indieningsdatum, waarin dezelfde uitvinding wordt beschreven
D: in de octrooiaanvrage vermeld
L: om andere redenen vermelde literatuur
&: lid van dezelfde octrooifamilie of overeenkomstige octrooipublicatie

RELEVANTE LITERATUUR

Categorie[1]	Literatuur met, voor zover nodig, aanduiding van speciaal van belang zijnde tekstgedeelten of figuren.	Van belang voor conclusie(s) nr:
A	CHEN T-F ET AL: "EFFECTIVE HARDWARE-BASED DATA PREFETCHING FOR HIGH-PERFORMANCE PROCESSORS", IEEE TRANSACTIONS ON COMPUTERS, IEEE SERVICE CENTER, LOS ALAMITOS, CA, US, deel 44, nr. 5, 1 mei 1995 (1995-05-01), bladzijden 609-623, XP000525553, * samenvatting * -----	1-30
A	JEAN-LOUP BAER ET AL: "An effective on-chip preloading scheme to reduce data access penalty", SUPERCOMPUTING, 1991. SUPERCOMPUTING '91. PROCEEDINGS OF THE 1991 ACM/IEEE CONFERENCE ON, IEEE, PISCATAWAY, NJ, USA, 18 november 1991 (1991-11-18), bladzijden 176-186, XP031577282, * samenvatting * -----	1-30

[1] CATEGORIE VAN DE VERMELDE LITERATUUR

X: de conclusie wordt als niet nieuw of niet inventief beschouwd ten opzichte van deze literatuur
Y: de conclusie wordt als niet inventief beschouwd ten opzichte van de combinatie van deze literatuur met andere geciteerde literatuur van dezelfde categorie, waarbij de combinatie voor de vakman voor de hand liggend wordt geacht
A: niet tot de categorie X of Y behorende literatuur die de stand van de techniek beschrijft
O: niet-schriftelijke stand van de techniek
P: tussen de voorrangsdatum en de indieningsdatum gepubliceerde literatuur

T: na de indieningsdatum of de voorrangsdatum gepubliceerde literatuur die niet bezwarend is voor de octrooiaanvrage, maar wordt vermeld ter verheldering van de theorie of het principe dat ten grondslag ligt aan de uitvinding
E: eerdere octrooi(aanvrage), gepubliceerd op of na de indieningsdatum, waarin dezelfde uitvinding wordt beschreven
D: in de octrooiaanvrage vermeld
L: om andere redenen vermelde literatuur
&: lid van dezelfde octrooifamilie of overeenkomstige octrooipublicatie

AANHANGSEL BEHORENDE BIJ HET RAPPORT BETREFFENDE HET ONDERZOEK NAAR DE STAND VAN DE TECHNIEK, UITGEVOERD IN DE OCTROOIAANVRAGE NR.

NO 138915
NL 2012319

Het aanhangsel bevat een opgave van elders gepubliceerde octrooiaanvragen of octrooien (zogenaamde leden van dezelfde octrooifamilie), die overeenkomen met octrooischriften genoemd in het rapport.
De opgave is samengesteld aan de hand van gegevens uit het computerbestand van het Europees Octrooibureau per
De juistheid en volledigheid van deze opgave wordt noch door het Europees Octrooibureau, noch door het Bureau voor de Industriële eigendom gegarandeerd; de gegevens worden verstrekt voor informatiedoeleinden.

24-11-2014

In het rapport genoemd octrooigeschrift	Datum van publicatie	Overeenkomend(e) geschrift(en)	Datum van publicatie
US 6633968 B2	14-10-2003	GEEN	
US 5305389 A	19-04-1994	GEEN	
US 6560693 B1	06-05-2003	CN 1306246 A US 6560693 B1	01-08-2001 06-05-2003
EP 0318778 A2	07-06-1989	EP 0318778 A2 JP H01169538 A US 4943908 A	07-06-1989 04-07-1989 24-07-1990

Algemene informatie over dit aanhangsel is gepubliceerd in de 'Official Journal' van het Europees Octrooibureau nr 12/82 blz 448 ev

SCHRIFTELIJKE OPINIE

DOSSIER NUMMER NO138915	INDIENINGSDATUM 25.02.2014	VOORRANGSDATUM 28.02.2013	AANVRAAGNUMMER NL2012319
CLASSIFICATIE INV. G06T1/60			
AANVRAGER Samsung Electronics Co., Ltd			

Deze schriftelijke opinie bevat een toelichting op de volgende onderdelen:

- ☒ Onderdeel I Basis van de schriftelijke opinie
- □ Onderdeel II Voorrang
- □ Onderdeel III Vaststelling nieuwheid, inventiviteit en industriële toepasbaarheid niet mogelijk
- □ Onderdeel IV De aanvraag heeft betrekking op meer dan één uitvinding
- ☒ Onderdeel V Gemotiveerde verklaring ten aanzien van nieuwheid, inventiviteit en industriële toepasbaarheid
- □ Onderdeel VI Andere geciteerde documenten
- □ Onderdeel VII Overige gebreken
- □ Onderdeel VIII Overige opmerkingen

	DE BEVOEGDE AMBTENAAR

Aanvraag nr.:

SCHRIFTELIJKE OPINIE

NL2012319

Onderdeel I Basis van de Schriftelijke Opinie

1. Deze schriftelijke opinie is opgesteld op basis van de meest recente conclusies ingediend voor aanvang van het onderzoek.

2. Met betrekking tot **nucleotide en/of aminozuur sequenties** die genoemd worden in de aanvraag en relevant zijn voor de uitvinding zoals beschreven in de conclusies, is dit onderzoek gedaan op basis van:

   a. type materiaal:

   □ sequentie opsomming

   □ tabel met betrekking tot de sequentie lijst

   b. vorm van het materiaal:

   □ op papier

   □ in elektronische vorm

   c. moment van indiening/aanlevering:

   □ opgenomen in de aanvraag zoals ingediend

   □ samen met de aanvraag elektronisch ingediend

   □ later aangeleverd voor het onderzoek

3. □ In geval er meer dan één versie of kopie van een sequentie opsomming of tabel met betrekking op een sequentie is ingediend of aangeleverd, zijn de benodigde verklaringen ingediend dat de informatie in de latere of additionele kopieën identiek is aan de aanvraag zoals ingediend of niet meer informatie bevatten dan de aanvraag zoals oorspronkelijk werd ingediend.

4. Overige opmerkingen:

Aanvraag nr.:

SCHRIFTELIJKE OPINIE

NL2012319

Onderdeel V Gemotiveerde verklaring ten aanzien van nieuwheid, inventiviteit en industriële toepasbaarheid

1. Verklaring

Nieuwheid	Ja:	Conclusies
	Nee:	Conclusies 1-30
Inventiviteit	Ja:	Conclusies
	Nee:	Conclusies 1-30
Industriële toepasbaarheid	Ja:	Conclusies 1-30
	Nee:	Conclusies

2. Citaties en toelichting:

Zie aparte bladzijde

Re Item V

Reasoned statement with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

Reference is made to the following documents:

D1 US 6 633 968 B2 (ZWIEGINCEW ARTHUR [US] ET AL) 14 oktober 2003 (2003-10-14)

D2 US 5 305 389 A (PALMER MARK L [US]) 19 april 1994 (1994-04-19)

D3 VANDERWIEL S P ET AL: "Data prefetch mechanisms",ACM COMPUTING SURVEYS, ACM, NEW YORK, NY, US, US,deel 32, nr. 2, 1 juni 2000 (2000-06-01), bladzijden 174-199, XP002977351,ISSN: 0360-0300

1 **CLARITY**

1.1 Claims 1, 10, 20 and 27 are not clear.

1.2 As a matter of fact, the feature:
- *"performing self-learning using addresses related to at least one page miss"* requires, according to the description, a prefetching step as an object of the *"self-learning"*. However, this step is NOT mentioned in the preamble of the claim. Hence, the mentioned claim is unclear.

1.3 The same objection applies accordingly to claims 10, 20 and 27 which are also not clear.

1.4 Moreover, although claims 20 and 27 have been drafted as separate device claims, they appear to relate effectively to the same subject-matter and to differ from each other only with regard to the definition of the subject-matter for which protection is sought and/or in respect of the terminology used for the features of that subject-matter. The aforementioned claims therefore lack conciseness and as such are not clear.

2 **CLARITY - DEPENDENT CLAIMS**

2.1 Claims 4, 12 and 21 are not clear because the *"generation rules"* are not precisely specified.

2.2 Claims 5, 6 and 22 are not clear because the *"determining"* step is not detailed.

2.3 Claim 9 is deeply unclear.

3 INVENTIVE STEP

3.1 Furthermore, the above-mentioned lack of clarity notwithstanding, the subject-matter of claims 1, 10, 20 and 27 does not involve an inventive step.

3.2 The document D1 is regarded as being the closest prior art to the subject-matter of claim 1, and discloses:
- *"A method ~~for rotating~~ an original image, the method including:
performing self-learning using addresses related to at least one page miss;
generating address information rules using a result of the self-learning;
prefetching of the original image from a memory device on the basis of the address generation rules so as to obtain a prefetched image;
and generating a rotated image using the prefetched image"* (D1, column 6, line 66 - column 8, line 9).

3.3 The subject-matter of claim 1 therefore differs from the subject-matter disclosed in D1 in that the claimed method is explicitly applied to an image rotation.

3.4 However, document D1 is aware that the disclosed technique may be applied to an *"application program module"* (D1, column 4, lines 31-34).

3.5 An image rotation is usually implemented using an *"application program module"*.

3.6 Hence, to apply the teachings of D1 to an image rotation would appear obvious for the man skilled in the art, and the present claim 1 is not inventive over the cited prior art.

3.7 Moreover, claim 1 lacks inventive step, also when starting at either D2 or D3 as closest prior art (see the corresponding citations of the search report).

3.8 The same reasoning applies, mutatis mutandis, to the subject-matter of the corresponding independent claims 10, 20 and 27 which therefore are also considered not inventive.

4 DEPENDENT CLAIMS

4.1 Dependent claims 2-9, 11-19, 21-26 and 28-30 do not contain any features which, in combination with the features of any claim to which they refer, meet the requirements of inventive step, see documents D1-D3 and the corresponding passages cited in the search report.