(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
20 September 2012 (20.09.2012)

WIPO | PCT

(10) International Publication Number
WO 2012/125343 A2

(51) International Patent Classification:
G09G 3/34 (2006.01) *G09G 3/20* (2006.01)

(21) International Application Number:
PCT/US2012/027879

(22) International Filing Date:
6 March 2012 (06.03.2012)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/453,064 15 March 2011 (15.03.2011) US
13/279,192 21 October 2011 (21.10.2011) US

(71) Applicant *(for all designated States except US)*: **QUALCOMM MEMS TECHNOLOGIES, INC.** [US/US]; 5775 Morehouse Drive, San Diego, California 92121 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **PARMAR, Manu** [IN/US]; 2581 Junction Avenue, San Jose, California 95134 (US). **GILLE, Jennifer Lee** [US/US]; 2581 Junction Avenue, San Jose, California 95134 (US). **LEWIS, Alan** [GB/US]; 2581 Junction Avenue, San Jose, California 95134 (US).

(74) Agent: **ABUMERI, Mark M.**; Knobbe Martens Olson & Bear LLP, 2040 Main Street, Fourteenth Floor, Irvine, California 92614 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *without international search report and to be republished upon receipt of that report (Rule 48.2(g))*

(54) **Title**: METHODS AND APPARATUS FOR DITHER SELECTION

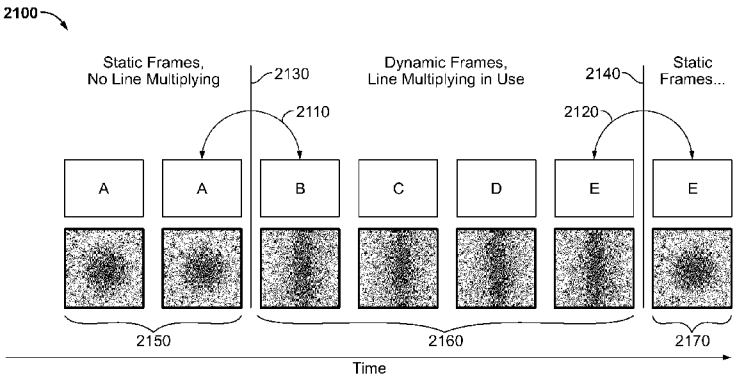


Figure 21

(57) **Abstract**: Systems, methods and apparatus including computer programs encoded on computer storage media optimize display image quality under a variety of imaging environments. Dynamic frame streams such as those present in video applications may require a higher frame rate to adequately convey motion in the stream. A line multiplying image pipeline may be utilized for dynamic frames, which lowers the resolution of the displayed image. When dithering line multiplied images, a noise signal including asymmetrical high frequency components around zero frequency may be utilized. The display of static frames, such as photographs, may be achieved with acceptable image quality using a relatively lower display frame rate. Such a frame rate may enable the display of a high resolution image. A noise signal tailored for higher resolution, non line multiplied frames, such as a noise signal with symmetric high frequency components around zero frequency may be utilized for static frames.

WO 2012/125343 A2

METHODS AND APPARATUS FOR DITHER SELECTION

TECHNICAL FIELD

[0001] This disclosure relates to image data processing for improving the display appearance of images that are rendered in displays that address lines simultaneously. The processing is especially suitable when used in conjunction with electromechanical display elements.

DESCRIPTION OF THE RELATED TECHNOLOGY

[0002] Electromechanical systems include devices having electrical and mechanical elements, actuators, transducers, sensors, optical components (e.g., mirrors) and electronics. Electromechanical systems can be manufactured at a variety of scales including, but not limited to, microscales and nanoscales. For example, microelectromechanical systems (MEMS) devices can include structures having sizes ranging from about a micron to hundreds of microns or more. Nanoelectromechanical systems (NEMS) devices can include structures having sizes smaller than a micron including, for example, sizes smaller than several hundred nanometers. Electromechanical elements may be created using deposition, etching, lithography, and/or other micromachining processes that etch away parts of substrates and/or deposited material layers, or that add layers to form electrical and electromechanical devices.

[0003] One type of electromechanical systems device is called an interferometric modulator (IMOD). As used herein, the term interferometric modulator or interferometric light modulator refers to a device that selectively absorbs and/or reflects light using the principles of optical interference. In some implementations, an interferometric modulator may include a pair of conductive plates, one or both of which may be transparent and/or reflective, wholly or in part, and capable of relative motion upon application of an appropriate electrical signal. In an implementation, one plate may include a stationary layer deposited on a substrate and the other plate may include a metallic membrane separated from the stationary layer by an air gap. The position of one plate in relation to another can change the optical interference of light incident on the interferometric modulator. Interferometric modulator devices have a wide range of

applications, and are anticipated to be used in improving existing products and creating new products, especially those with display capabilities.

SUMMARY

[0004] The systems, methods and devices of the disclosure each have several innovative aspects, no single one of which is solely responsible for the desirable attributes disclosed herein.

[0005] One innovative aspect of the subject matter described in this disclosure can be implemented in a method of generating and displaying image data on an electronic display, including storing a current frame to an electronic memory, determining a display mode based on whether a current frame stream is static or dynamic, selecting a dithering method based on the display mode, dithering an intermediate frame using the dithering method to produce a dithered frame, and updating an electronic display with data derived from the dithered frame.

[0006] Another innovative aspect can be implemented in a display apparatus including a display connection means, an electronic memory, and an electronic processor, wherein the processor is configured to read and write to the electronic memory and to generate a display signal on the display connection means, a host program configured to select a display mode based on whether a current frame stream is static or dynamic, select a dithering method based on the display mode, dither an intermediate frame using the dithering method to produce a dithered frame, and generate a display signal on the display connection means based on data derived from the dithered frame.

[0007] Another innovative aspect can be implemented in a display apparatus including a display displaying lines derived from image data dithered based on whether a current frame stream was characterized as dynamic or static.

[0008] Another innovative aspect can be implemented in a display apparatus including means for storing a current frame in an electronic memory, means for determining a display mode based on whether a current frame stream is static or dynamic, means for selecting a dithering method based on the display mode, means for dithering an intermediate frame using the dithering method to produce a dithered frame, and means for updating an electronic display with data derived from the dithered frame.

[0009] Another innovative aspect can be implemented in a computer readable storage medium having instructions stored thereon that cause a processing circuit to perform: storing a current frame to an electronic memory, determining a display mode based on whether a current frame stream is static or dynamic, selecting a dithering method based on the display mode, dithering an intermediate frame using the dithering method to produce a dithered frame, and updating an electronic display with data derived from the dithered frame.

[0010] Details of one or more implementations of the subject matter described in this specification are set forth in the accompanying drawings and the description below. Other features, aspects, and advantages will become apparent from the description, the drawings, and the claims. Note that the relative dimensions of the following figures may not be drawn to scale.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] The patent or application file contains at least one drawing executed in color. Copies of this patent or patent application publication with color drawings will be provided by the Office upon request and payment of the necessary fee.

[0012] Figure 1 shows an example of an isometric view depicting two adjacent pixels in a series of pixels of an interferometric modulator (IMOD) display device.

[0013] Figure 2 shows an example of a system block diagram illustrating an electronic device incorporating a 3x3 interferometric modulator display.

[0014] Figure 3 shows an example of a diagram illustrating movable reflective layer position versus applied voltage for the interferometric modulator of Figure 1.

[0015] Figure 4 shows an example of a table illustrating various states of an interferometric modulator when various common and segment voltages are applied.

[0016] Figure 5A shows an example of a diagram illustrating a frame of display data in the 3x3 interferometric modulator display of Figure 2.

[0017] Figure 5B shows an example of a timing diagram for common and segment signals that may be used to write the frame of display data illustrated in Figure 5A.

[0018] Figure 6A shows an example of a partial cross-section of the interferometric modulator display of Figure 1.

[0019] Figures 6B–6E show examples of cross-sections of varying implementations of interferometric modulators.

[0020] Figure 7 shows an example of a flow diagram illustrating a manufacturing process for an interferometric modulator.

[0021] Figures 8A–8E show examples of cross-sectional schematic illustrations of various stages in a method of making an interferometric modulator.

[0022] Figure 9 schematically illustrates an example array of display elements.

[0023] Figure 10 shows an example system block diagram illustrating a visual display device including a plurality of interferometric modulators.

[0024] Figure 11 shows an example of a flowchart illustrating a process for writing a portion of a frame using a line multiplying process.

[0025] Figure 12 shows an example 12x16 array of pixel data.

[0026] Figure 13 shows an example line doubled array derived from the array of Figure 12.

[0027] Figure 14 shows an example system block diagram illustrating a visual display device including a plurality of imaging pipeline components.

[0028] Figure 15 shows the data flow in one implementation of a variable frame rate display system.

[0029] Figure 16 shows the data flow of image data as it is processed by one implementation of a line multiplying image processing pipeline.

[0030] Figure 17 is a flow chart illustrating one implementation of a line multiplying image pipeline.

[0031] Figure 18 shows three example noise signals and their corresponding discrete Fourier transforms used to dither images in various implementations.

[0032] Figure 19 shows a variety of images produced by different implementations of image processing components.

[0033] Figure 20 is a flow chart illustrating one implementation of a variable frame rate image processing system with dynamic dither selection.

[0034] Figure 21 shows the selection of a dither noise signal as a frame stream transitions between a dynamic and static character.

[0035] Figures 22A and 22B show examples of system block diagrams illustrating a display device 40 that includes a plurality of interferometric modulators.

[0036] Like reference numbers and designations in the various drawings indicate like elements.

DETAILED DESCRIPTION

[0037] The following detailed description is directed to certain implementations for the purposes of describing the innovative aspects. However, the teachings herein can be applied in a multitude of different ways. The described implementations may be implemented in any device that is configured to display an image, whether in motion (e.g., video) or stationary (e.g., still image), and whether textual, graphical or pictorial. More particularly, it is contemplated that the implementations may be implemented in or associated with a variety of electronic devices such as, but not limited to, mobile telephones, multimedia Internet enabled cellular telephones, mobile television receivers, wireless devices, smartphones, bluetooth devices, personal data assistants (PDAs), wireless electronic mail receivers, hand-held or portable computers, netbooks, notebooks, smartbooks, printers, copiers, scanners, facsimile devices, GPS receivers/navigators, cameras, MP3 players, camcorders, game consoles, wrist watches, clocks, calculators, television monitors, flat panel displays, electronic reading devices (e.g., e-readers), computer monitors, auto displays (e.g., odometer display, etc.), cockpit controls and/or displays, camera view displays (e.g., display of a rear view camera in a vehicle), electronic photographs, electronic billboards or signs, projectors, architectural structures, microwaves, refrigerators, stereo systems, cassette recorders or players, DVD players, CD players, VCRs, radios, portable memory chips, washers, dryers, washer/dryers, packaging (e.g., MEMS and non-MEMS),

aesthetic structures (e.g., display of images on a piece of jewelry) and a variety of electromechanical systems devices. The teachings herein also can be used in non-display applications such as, but not limited to, electronic switching devices, radio frequency filters, sensors, accelerometers, gyroscopes, motion-sensing devices, magnetometers, inertial components for consumer electronics, parts of consumer electronics products, varactors, liquid crystal devices, electrophoretic devices, drive schemes, manufacturing processes, electronic test equipment. Thus, the teachings are not intended to be limited to the implementations depicted solely in the Figures, but instead have wide applicability as will be readily apparent to one having ordinary skill in the art.

[0038] In some display implementations, it is desired to update the displayed image at a fast rate, such as 15, 30, or 60 times per second. This is especially true when animation or video is being displayed. Because writing a line of data to a display takes a certain amount of time, a limit exists as to how fast a new image can be written. This limit will be different depending on the display technology. In some implementations, the achievable update rate is increased at the cost of reducing display resolution by simultaneously writing the same image data to two (or more) lines of the display. This essentially cuts at least in half the number of write cycles needed to write a new image to the display. In some implementations, the lines of the image are dithered before they are multiplied. Because the multiplication of the lines affects the visual results of the dithering operation, some implementations select a dither noise signal specifically adapted to optimize the appearance of line multiplied images.

[0039] In other display implementations, it may be desired to update the displayed image at a variable rate, depending on the content of the image being displayed. For example, the display of a static image, such as a photograph, may be displayed with adequate quality using a slower frame rate. As discussed above, this slower frame rate may further enable a higher display resolution in some implementations. More demanding display applications, such as video and video games, for example, may require a faster frame rate to adequately convey smooth motion effects. A video may itself include both relatively static portions with little motion, which could be adequately displayed at a slower frame rate, along with more dynamic portions, which require a faster frame rate.

[0040] Therefore, the different characteristics of photographs and videos, for example, along with characteristics within a particular video frame sequence, make it desirable to vary the frame update rate to optimize for either display resolution or smooth motion display. As the display frame rate is varied, the image pipeline may shift between a normal display mode and a line multiplied mode. Because the image quality of line multiplied displays can be optimized through the use of a specialized dither noise signal, it becomes desirable to vary the dither noise signal depending on whether the display is operating in a normal or line multiplying mode.

[0041] Particular implementations of the subject matter described in this disclosure can be implemented to realize one or more of the following potential advantages. "Line doubling," where identical image data is written to multiple lines of a display at once increases the achievable frame rate of a display. Dithering an image with a specialized noise signal before line multiplying improves the visual appearance of the line multiplied, reduced resolution display. Selecting an appropriate dither noise signal for each frame to be displayed provides for improved image quality when switching between a line doubled display mode and a non doubled display mode. Note that line doubling is just one implementation of the more generalized technique of multi-line addressing. The subject matter described herein is equally applicable to implementations that address more than two lines of a display at once, for example, three, four, or five lines of a display, such as an IMOD display, simultaneously.

[0042] One example of a suitable MEMS device, to which the described implementations may apply, is a reflective display device. Reflective display devices can incorporate interferometric modulators (IMODs) to selectively absorb and/or reflect light incident thereon using principles of optical interference. IMODs can include an absorber, a reflector that is movable with respect to the absorber, and an optical resonant cavity defined between the absorber and the reflector. The reflector can be moved to two or more different positions, which can change the size of the optical resonant cavity and thereby affect the reflectance of the interferometric modulator. The reflectance spectrums of IMODs can create fairly broad spectral bands which can be shifted across the visible wavelengths to generate different colors. The position of the spectral band can

be adjusted by changing the thickness of the optical resonant cavity, i.e., by changing the position of the reflector.

[0043] Figure 1 shows an example of an isometric view depicting two adjacent pixels in a series of pixels of an interferometric modulator (IMOD) display device. The IMOD display device includes one or more interferometric MEMS display elements. In these devices, the pixels of the MEMS display elements can be in either a bright or dark state. In the bright ("relaxed," "open" or "on") state, the display element reflects a large portion of incident visible light, e.g., to a user. Conversely, in the dark ("actuated," "closed" or "off") state, the display element reflects little incident visible light. In some implementations, the light reflectance properties of the on and off states may be reversed. MEMS pixels can be configured to reflect predominantly at particular wavelengths allowing for a color display in addition to black and white.

[0044] The IMOD display device can include a row/column array of IMODs. Each IMOD can include a pair of reflective layers, i.e., a movable reflective layer and a fixed partially reflective layer, positioned at a variable and controllable distance from each other to form an air gap (also referred to as an optical gap or cavity). The movable reflective layer may be moved between at least two positions. In a first position, i.e., a relaxed position, the movable reflective layer can be positioned at a relatively large distance from the fixed partially reflective layer. In a second position, i.e., an actuated position, the movable reflective layer can be positioned more closely to the partially reflective layer. Incident light that reflects from the two layers can interfere constructively or destructively depending on the position of the movable reflective layer, producing either an overall reflective or non-reflective state for each pixel. In some implementations, the IMOD may be in a reflective state when unactuated, reflecting light within the visible spectrum, and may be in a dark state when unactuated, reflecting light outside of the visible range (e.g., infrared light). In some other implementations, however, an IMOD may be in a dark state when unactuated, and in a reflective state when actuated. In some implementations, the introduction of an applied voltage can drive the pixels to change states. In some other implementations, an applied charge can drive the pixels to change states.

[0045] The depicted portion of the pixel array in Figure 1 includes two adjacent interferometric modulators 12. In the IMOD 12 on the left (as illustrated), a movable reflective layer 14 is illustrated in a relaxed position at a predetermined distance from an optical stack 16, which includes a partially reflective layer. The voltage V_0 applied across the IMOD 12 on the left is insufficient to cause actuation of the movable reflective layer 14. In the IMOD 12 on the right, the movable reflective layer 14 is illustrated in an actuated position near or adjacent the optical stack 16. The voltage V_{bias} applied across the IMOD 12 on the right is sufficient to maintain the movable reflective layer 14 in the actuated position.

[0046] In Figure 1, the reflective properties of pixels 12 are generally illustrated with arrows 13 indicating light incident upon the pixels 12, and light 15 reflecting from the pixel 12 on the left. Although not illustrated in detail, it will be understood by one having ordinary skill in the art that most of the light 13 incident upon the pixels 12 will be transmitted through the transparent substrate 20, toward the optical stack 16. A portion of the light incident upon the optical stack 16 will be transmitted through the partially reflective layer of the optical stack 16, and a portion will be reflected back through the transparent substrate 20. The portion of light 13 that is transmitted through the optical stack 16 will be reflected at the movable reflective layer 14, back toward (and through) the transparent substrate 20. Interference (constructive or destructive) between the light reflected from the partially reflective layer of the optical stack 16 and the light reflected from the movable reflective layer 14 will determine the wavelength(s) of light 15 reflected from the pixel 12.

[0047] The optical stack 16 can include a single layer or several layers. The layer(s) can include one or more of an electrode layer, a partially reflective and partially transmissive layer and a transparent dielectric layer. In some implementations, the optical stack 16 is electrically conductive, partially transparent and partially reflective, and may be fabricated, for example, by depositing one or more of the above layers onto a transparent substrate 20. The electrode layer can be formed from a variety of materials, such as various metals, for example indium tin oxide (ITO). The partially reflective layer can be formed from a variety of materials that are partially reflective, such as various metals, e.g., chromium (Cr), semiconductors, and dielectrics. The partially reflective

layer can be formed of one or more layers of materials, and each of the layers can be formed of a single material or a combination of materials. In some implementations, the optical stack 16 can include a single semi-transparent thickness of metal or semiconductor which serves as both an optical absorber and conductor, while different, more conductive layers or portions (e.g., of the optical stack 16 or of other structures of the IMOD) can serve to bus signals between IMOD pixels. The optical stack 16 also can include one or more insulating or dielectric layers covering one or more conductive layers or a conductive/absorptive layer.

[0048] In some implementations, the layer(s) of the optical stack 16 can be patterned into parallel strips, and may form row electrodes in a display device as described further below. As will be understood by one having skill in the art, the term "patterned" is used herein to refer to masking as well as etching processes. In some implementations, a highly conductive and reflective material, such as aluminum (Al), may be used for the movable reflective layer 14, and these strips may form column electrodes in a display device. The movable reflective layer 14may be formed as a series of parallel strips of a deposited metal layer or layers (orthogonal to the row electrodes of the optical stack 16) to form columns deposited on top of posts 18 and an intervening sacrificial material deposited between the posts 18. When the sacrificial material is etched away, a defined gap 19, or optical cavity, can be formed between the movable reflective layer 14 and the optical stack 16. In some implementations, the spacing between posts 18 may be on the order of 1–1000 um, while the gap 19 may be on the order of < 10,000 Angstroms (Å).

[0049] In some implementations, each pixel of the IMOD, whether in the actuated or relaxed state, is essentially a capacitor formed by the fixed and moving reflective layers. When no voltage is applied, the movable reflective layer 14a remains in a mechanically relaxed state, as illustrated by the pixel 12 on the left in Figure 1, with the gap 19 between the movable reflective layer 14 and optical stack 16. However, when a potential difference, e.g., voltage, is applied to at least one of a selected row and column, the capacitor formed at the intersection of the row and column electrodes at the corresponding pixel becomes charged, and electrostatic forces pull the electrodes together. If the applied voltage exceeds a threshold, the movable reflective layer 14 can

deform and move near or against the optical stack 16. A dielectric layer (not shown) within the optical stack 16 may prevent shorting and control the separation distance between the layers 14 and 16, as illustrated by the actuated pixel 12 on the right in Figure 1. The behavior is the same regardless of the polarity of the applied potential difference. Though a series of pixels in an array may be referred to in some instances as "rows" or "columns," a person having ordinary skill in the art will readily understand that referring to one direction as a "row" and another as a "column" is arbitrary. Restated, in some orientations, the rows can be considered columns, and the columns considered to be rows. Furthermore, the display elements may be evenly arranged in orthogonal rows and columns (an "array"), or arranged in non-linear configurations, for example, having certain positional offsets with respect to one another (a "mosaic"). The terms "array" and "mosaic" may refer to either configuration. Thus, although the display is referred to as including an "array" or "mosaic," the elements themselves need not be arranged orthogonally to one another, or disposed in an even distribution, in any instance, but may include arrangements having asymmetric shapes and unevenly distributed elements.

[0050] Figure 2 shows an example of a system block diagram illustrating an electronic device incorporating a 3x3 interferometric modulator display. The electronic device includes a processor 21 that may be configured to execute one or more software modules. In addition to executing an operating system, the processor 21 may be configured to execute one or more software applications, including a web browser, a telephone application, an email program, or any other software application.

[0051] The processor 21 can be configured to communicate with an array driver 22. The array driver 22 can include a row driver circuit 24 and a column driver circuit 26 that provide signals to, e.g., a display array or panel 30. The cross section of the IMOD display device illustrated in Figure 1 is shown by the lines 1–1 in Figure 2. Although Figure 2 illustrates a 3x3 array of IMODs for the sake of clarity, the display array 30 may contain a very large number of IMODs, and may have a different number of IMODs in rows than in columns, and vice versa.

[0052] In alternate implementations of electronic device 30, the processor may not communicate directly with the array driver 22 as shown in Figure 2. Instead, the processor and array driver may be connected indirectly by any one of several display

connection technologies known in the art. For example, the array driver may be attached to the device 30 via an HDMI connection. Alternatively other technologies such as 1394 "Firewire", USB, DVI, or VGA may be employed. In these implementations, processor 21 generates a display signal over the display connection to communicate with array driver 22.

[0053] Figure 3 shows an example of a diagram illustrating movable reflective layer position versus applied voltage for the interferometric modulator of Figure 1. For MEMS interferometric modulators, the row/column (i.e., common/segment) write procedure may take advantage of a hysteresis property of these devices as illustrated in Figure 3. An interferometric modulator may require, for example, about a 10-volt potential difference to cause the movable reflective layer, or mirror, to change from the relaxed state to the actuated state. When the voltage is reduced from that value, the movable reflective layer maintains its state as the voltage drops back below, e.g., 10-volts, however, the movable reflective layer does not relax completely until the voltage drops below 2-volts. Thus, a range of voltage, approximately 3 to 7-volts, as shown in Figure 3, exists where there is a window of applied voltage within which the device is stable in either the relaxed or actuated state. This is referred to herein as the "hysteresis window" or "stability window." For a display array 30 having the hysteresis characteristics of Figure 3, the row/column write procedure can be designed to address one or more rows at a time, such that during the addressing of a given row, pixels in the addressed row that are to be actuated are exposed to a voltage difference of about 10-volts, and pixels that are to be relaxed are exposed to a voltage difference of near zero volts. After addressing, the pixels are exposed to a steady state or bias voltage difference of approximately 5-volts such that they remain in the previous strobing state. In this example, after being addressed, each pixel sees a potential difference within the "stability window" of about 3–7-volts. This hysteresis property feature enables the pixel design, e.g., illustrated in Figure 1, to remain stable in either an actuated or relaxed pre-existing state under the same applied voltage conditions. Since each IMOD pixel, whether in the actuated or relaxed state, is essentially a capacitor formed by the fixed and moving reflective layers, this stable state can be held at a steady voltage within the hysteresis window without substantially consuming or losing power.

Moreover, essentially little or no current flows into the IMOD pixel if the applied voltage potential remains substantially fixed.

[0054] In some implementations, a frame of an image may be created by applying data signals in the form of "segment" voltages along the set of column electrodes, in accordance with the desired change (if any) to the state of the pixels in a given row. Each row of the array can be addressed in turn, such that the frame is written one row at a time. To write the desired data to the pixels in a first row, segment voltages corresponding to the desired state of the pixels in the first row can be applied on the column electrodes, and a first row pulse in the form of a specific "common" voltage or signal can be applied to the first row electrode. The set of segment voltages can then be changed to correspond to the desired change (if any) to the state of the pixels in the second row, and a second common voltage can be applied to the second row electrode. In some implementations, the pixels in the first row are unaffected by the change in the segment voltages applied along the column electrodes, and remain in the state they were set to during the first common voltage row pulse. This process may be repeated for the entire series of rows, or alternatively, columns, in a sequential fashion to produce the image frame. The frames can be refreshed and/or updated with new image data by continually repeating this process at some desired number of frames per second.

[0055] The combination of segment and common signals applied across each pixel (that is, the potential difference across each pixel) determines the resulting state of each pixel. Figure 4 shows an example of a table illustrating various states of an interferometric modulator when various common and segment voltages are applied. As will be readily understood by one having ordinary skill in the art, the "segment" voltages can be applied to either the column electrodes or the row electrodes, and the "common" voltages can be applied to the other of the column electrodes or the row electrodes.

[0056] As illustrated in Figure 4 (as well as in the timing diagram shown in Figure 5B), when a release voltage VC_{REL} is applied along a common line, all interferometric modulator elements along the common line will be placed in a relaxed state, alternatively referred to as a released or unactuated state, regardless of the voltage applied along the segment lines, i.e., high segment voltage VS_H and low segment voltage VS_L. In particular, when the release voltage VC_{REL} is applied along a common line, the

potential voltage across the modulator (alternatively referred to as a pixel voltage) is within the relaxation window (see Figure 3, also referred to as a release window) both when the high segment voltage VS_H and the low segment voltage VS_L are applied along the corresponding segment line for that pixel.

[0057] When a hold voltage is applied on a common line, such as a high hold voltage VC_{HOLD_H} or a low hold voltage VC_{HOLD_L}, the state of the interferometric modulator will remain constant. For example, a relaxed IMOD will remain in a relaxed position, and an actuated IMOD will remain in an actuated position. The hold voltages can be selected such that the pixel voltage will remain within a stability window both when the high segment voltage VS_H and the low segment voltage VS_L are applied along the corresponding segment line. Thus, the segment voltage swing, i.e., the difference between the high VS_H and low segment voltage VS_L, is less than the width of either the positive or the negative stability window.

[0058] When an addressing, or actuation, voltage is applied on a common line, such as a high addressing voltage VC_{ADD_H} or a low addressing voltage VC_{ADD_L}, data can be selectively written to the modulators along that line by application of segment voltages along the respective segment lines. The segment voltages may be selected such that actuation is dependent upon the segment voltage applied. When an addressing voltage is applied along a common line, application of one segment voltage will result in a pixel voltage within a stability window, causing the pixel to remain unactuated. In contrast, application of the other segment voltage will result in a pixel voltage beyond the stability window, resulting in actuation of the pixel. The particular segment voltage which causes actuation can vary depending upon which addressing voltage is used. In some implementations, when the high addressing voltage VC_{ADD_H} is applied along the common line, application of the high segment voltage VS_H can cause a modulator to remain in its current position, while application of the low segment voltage VS_L can cause actuation of the modulator. As a corollary, the effect of the segment voltages can be the opposite when a low addressing voltage VC_{ADD_L} is applied, with high segment voltage VS_H causing actuation of the modulator, and low segment voltage VS_L having no effect (i.e., remaining stable) on the state of the modulator.

[0059] In some implementations, hold voltages, address voltages, and segment voltages may be used which always produce the same polarity potential difference across the modulators. In some other implementations, signals can be used which alternate the polarity of the potential difference of the modulators. Alternation of the polarity across the modulators (that is, alternation of the polarity of write procedures) may reduce or inhibit charge accumulation which could occur after repeated write operations of a single polarity.

[0060] Figure 5A shows an example of a diagram illustrating a frame of display data in the 3x3 interferometric modulator display of Figure 2. Figure 5B shows an example of a timing diagram for common and segment signals that may be used to write the frame of display data illustrated in Figure 5A. The signals can be applied to the, e.g., 3x3 array of Figure 2, which will ultimately result in the line time 60e display arrangement illustrated in Figure 5A. The actuated modulators in Figure 5A are in a dark-state, i.e., where a substantial portion of the reflected light is outside of the visible spectrum so as to result in a dark appearance to, e.g., a viewer. Prior to writing the frame illustrated in Figure 5A, the pixels can be in any state, but the write procedure illustrated in the timing diagram of Figure 5B presumes that each modulator has been released and resides in an unactuated state before the first line time 60a.

[0061] During the first line time 60a: a release voltage 70 is applied on common line 1; the voltage applied on common line 2 begins at a high hold voltage 72 and moves to a release voltage 70; and a low hold voltage 76 is applied along common line 3. Thus, the modulators (common 1, segment 1), (1,2) and (1,3) along common line 1 remain in a relaxed, or unactuated, state for the duration of the first line time 60a, the modulators (2,1), (2,2) and (2,3) along common line 2 will move to a relaxed state, and the modulators (3,1), (3,2) and (3,3) along common line 3 will remain in their previous state. With reference to Figure 4, the segment voltages applied along segment lines 1, 2 and 3 will have no effect on the state of the interferometric modulators, as none of common lines 1, 2 or 3 are being exposed to voltage levels causing actuation during line time 60a (i.e., VC_{REL} – relax and VC_{HOLD_L} – stable).

[0062] During the second line time 60b, the voltage on common line 1 moves to a high hold voltage 72, and all modulators along common line 1 remain in a relaxed

state regardless of the segment voltage applied because no addressing, or actuation, voltage was applied on the common line 1. The modulators along common line 2 remain in a relaxed state due to the application of the release voltage 70, and the modulators (3,1), (3,2) and (3,3) along common line 3 will relax when the voltage along common line 3 moves to a release voltage 70.

[0063] During the third line time 60c, common line 1 is addressed by applying a high address voltage 74 on common line 1. Because a low segment voltage 64 is applied along segment lines 1 and 2 during the application of this address voltage, the pixel voltage across modulators (1,1) and (1,2) is greater than the high end of the positive stability window (i.e., the voltage differential exceeded a predefined threshold) of the modulators, and the modulators (1,1) and (1,2) are actuated. Conversely, because a high segment voltage 62 is applied along segment line 3, the pixel voltage across modulator (1,3) is less than that of modulators (1,1) and (1,2), and remains within the positive stability window of the modulator; modulator (1,3) thus remains relaxed. Also during line time 60c, the voltage along common line 2 decreases to a low hold voltage 76, and the voltage along common line 3 remains at a release voltage 70, leaving the modulators along common lines 2 and 3 in a relaxed position.

[0064] During the fourth line time 60d, the voltage on common line 1 returns to a high hold voltage 72, leaving the modulators along common line 1 in their respective addressed states. The voltage on common line 2 is decreased to a low address voltage 78. Because a high segment voltage 62 is applied along segment line 2, the pixel voltage across modulator (2,2) is below the lower end of the negative stability window of the modulator, causing the modulator (2,2) to actuate. Conversely, because a low segment voltage 64 is applied along segment lines 1 and 3, the modulators (2,1) and (2,3) remain in a relaxed position. The voltage on common line 3 increases to a high hold voltage 72, leaving the modulators along common line 3 in a relaxed state.

[0065] Finally, during the fifth line time 60e, the voltage on common line 1 remains at high hold voltage 72, and the voltage on common line 2 remains at a low hold voltage 76, leaving the modulators along common lines 1 and 2 in their respective addressed states. The voltage on common line 3 increases to a high address voltage 74 to address the modulators along common line 3. As a low segment voltage 64 is applied on

segment lines 2 and 3, the modulators (3,2) and (3,3) actuate, while the high segment voltage 62 applied along segment line 1 causes modulator (3,1) to remain in a relaxed position. Thus, at the end of the fifth line time 60e, the 3x3 pixel array is in the state shown in Figure 5A, and will remain in that state as long as the hold voltages are applied along the common lines, regardless of variations in the segment voltage which may occur when modulators along other common lines (not shown) are being addressed.

[0066] In the timing diagram of Figure 5B, a given write procedure (i.e., line times 60a–60e) can include the use of either high hold and address voltages, or low hold and address voltages. Once the write procedure has been completed for a given common line (and the common voltage is set to the hold voltage having the same polarity as the actuation voltage), the pixel voltage remains within a given stability window, and does not pass through the relaxation window until a release voltage is applied on that common line. Furthermore, as each modulator is released as part of the write procedure prior to addressing the modulator, the actuation time of a modulator, rather than the release time, may determine the necessary line time. Specifically, in implementations in which the release time of a modulator is greater than the actuation time, the release voltage may be applied for longer than a single line time, as depicted in Figure 5B. In some other implementations, voltages applied along common lines or segment lines may vary to account for variations in the actuation and release voltages of different modulators, such as modulators of different colors.

[0067] The details of the structure of interferometric modulators that operate in accordance with the principles set forth above may vary widely. For example, Figures 6A–6E show examples of cross-sections of varying implementations of interferometric modulators, including the movable reflective layer 14 and its supporting structures. Figure 6A shows an example of a partial cross-section of the interferometric modulator display of Figure 1, where a strip of metal material, i.e., the movable reflective layer 14 is deposited on supports 18 extending orthogonally from the substrate 20. In Figure 6B, the movable reflective layer 14 of each IMOD is generally square or rectangular in shape and attached to supports at or near the corners, on tethers 32. In Figure 6C, the movable reflective layer 14 is generally square or rectangular in shape and suspended from a deformable layer 34, which may include a flexible metal. The deformable layer 34 can

connect, directly or indirectly, to the substrate 20 around the perimeter of the movable reflective layer 14. These connections are herein referred to as support posts. The implementation shown in Figure 6C has additional benefits deriving from the decoupling of the optical functions of the movable reflective layer 14 from its mechanical functions, which are carried out by the deformable layer 34. This decoupling allows the structural design and materials used for the reflective layer 14 and those used for the deformable layer 34 to be optimized independently of one another.

[0068] Figure 6D shows another example of an IMOD, where the movable reflective layer 14 includes a reflective sub-layer 14a. The movable reflective layer 14 rests on a support structure, such as support posts 18. The support posts 18 provide separation of the movable reflective layer 14 from the lower stationary electrode (i.e., part of the optical stack 16 in the illustrated IMOD) so that a gap 19 is formed between the movable reflective layer 14 and the optical stack 16, for example when the movable reflective layer 14 is in a relaxed position. The movable reflective layer 14 also can include a conductive layer 14c, which may be configured to serve as an electrode, and a support layer 14b. In this example, the conductive layer 14c is disposed on one side of the support layer 14b, distal from the substrate 20, and the reflective sub-layer 14a is disposed on the other side of the support layer 14b, proximal to the substrate 20. In some implementations, the reflective sub-layer 14a can be conductive and can be disposed between the support layer 14b and the optical stack 16. The support layer 14b can include one or more layers of a dielectric material, for example, silicon oxynitride (SiON) or silicon dioxide (SiO_2). In some implementations, the support layer 14b can be a stack of layers, such as, for example, a SiO_2/SiON/SiO_2 tri-layer stack. Either or both of the reflective sub-layer 14a and the conductive layer 14c can include, e.g., an Al alloy with about 0.5% Cu, or another reflective metallic material. Employing conductive layers 14a, 14c above and below the dielectric support layer 14b can balance stresses and provide enhanced conduction. In some implementations, the reflective sub-layer 14a and the conductive layer 14c can be formed of different materials for a variety of design purposes, such as achieving specific stress profiles within the movable reflective layer 14.

[0069] As illustrated in Figure 6D, some implementations also can include a black mask structure 23. The black mask structure 23 can be formed in optically inactive

regions (e.g., between pixels or under posts 18) to absorb ambient or stray light. The black mask structure 23 also can improve the optical properties of a display device by inhibiting light from being reflected from or transmitted through inactive portions of the display, thereby increasing the contrast ratio. Additionally, the black mask structure 23 can be conductive and be configured to function as an electrical bussing layer. In some implementations, the row electrodes can be connected to the black mask structure 23 to reduce the resistance of the connected row electrode. The black mask structure 23 can be formed using a variety of methods, including deposition and patterning techniques. The black mask structure 23 can include one or more layers. For example, in some implementations, the black mask structure 23 includes a molybdenum-chromium (MoCr) layer that serves as an optical absorber, a SiO_2 layer, and an aluminum alloy that serves as a reflector and a bussing layer, with a thickness in the range of about 30–80 Å, 500–1000 Å, and 500–6000 Å, respectively. The one or more layers can be patterned using a variety of techniques, including photolithography and dry etching, including, for example, CF_4 and/or O_2 for the MoCr and SiO_2 layers and Cl_2 and/or BCl_3 for the aluminum alloy layer. In some implementations, the black mask 23 can be an etalon or interferometric stack structure. In such interferometric stack black mask structures 23, the conductive absorbers can be used to transmit or bus signals between lower, stationary electrodes in the optical stack 16 of each row or column. In some implementations, a spacer layer 35 can serve to generally electrically isolate the absorber layer 16a from the conductive layers in the black mask 23.

[0070] Figure 6E shows another example of an IMOD, where the movable reflective layer 14 is self supporting. In contrast with Figure 6D, the implementation of Figure 6E does not include support posts 18. Instead, the movable reflective layer 14 contacts the underlying optical stack 16 at multiple locations, and the curvature of the movable reflective layer 14 provides sufficient support that the movable reflective layer 14 returns to the unactuated position of Figure 6E when the voltage across the interferometric modulator is insufficient to cause actuation. The optical stack 16, which may contain a plurality of several different layers, is shown here for clarity including an optical absorber 16a, and a dielectric 16b. In some implementations, the optical absorber 16a may serve both as a fixed electrode and as a partially reflective layer.

[0071] In implementations such as those shown in Figures 6A–6E, the IMODs function as direct-view devices, in which images are viewed from the front side of the transparent substrate 20, i.e., the side opposite to that upon which the modulator is arranged. In these implementations, the back portions of the device (that is, any portion of the display device behind the movable reflective layer 14, including, for example, the deformable layer 34 illustrated in Figure 6C) can be configured and operated upon without impacting or negatively affecting the image quality of the display device, because the reflective layer 14 optically shields those portions of the device. For example, in some implementations a bus structure (not illustrated) can be included behind the movable reflective layer 14 which provides the ability to separate the optical properties of the modulator from the electromechanical properties of the modulator, such as voltage addressing and the movements that result from such addressing. Additionally, the implementations of Figures 6A–6E can simplify processing, such as, e.g., patterning.

[0072] Figure 7 shows an example of a flow diagram illustrating a manufacturing process 80 for an interferometric modulator, and Figures 8A–8E show examples of cross-sectional schematic illustrations of corresponding stages of such a manufacturing process 80. In some implementations, the manufacturing process 80 can be implemented to manufacture, e.g., interferometric modulators of the general type illustrated in Figures 1 and 6, in addition to other blocks not shown in Figure 7. With reference to Figures 1, 6 and 7, the process 80 begins at block 82 with the formation of the optical stack 16 over the substrate 20. Figure 8A illustrates such an optical stack 16 formed over the substrate 20. The substrate 20 may be a transparent substrate such as glass or plastic, it may be flexible or relatively stiff and unbending, and may have been subjected to prior preparation processes, e.g., cleaning, to facilitate efficient formation of the optical stack 16. As discussed above, the optical stack 16 can be electrically conductive, partially transparent and partially reflective and may be fabricated, for example, by depositing one or more layers having the desired properties onto the transparent substrate 20. In Figure 8A, the optical stack 16 includes a multilayer structure having sub-layers 16a and 16b, although more or fewer sub-layers may be included in some other implementations. In some implementations, one of the sub-layers 16a, 16b can be configured with both optically absorptive and conductive properties, such

as the combined conductor/absorber sub-layer 16a. Additionally, one or more of the sub-layers 16a, 16b can be patterned into parallel strips, and may form row electrodes in a display device. Such patterning can be performed by a masking and etching process or another suitable process known in the art. In some implementations, one of the sub-layers 16a, 16b can be an insulating or dielectric layer, such as sub-layer 16b that is deposited over one or more metal layers (e.g., one or more reflective and/or conductive layers). In addition, the optical stack 16 can be patterned into individual and parallel strips that form the rows of the display.

[0073] The process 80 continues at block 84 with the formation of a sacrificial layer 25 over the optical stack 16. The sacrificial layer 25 is later removed (e.g., at block 90) to form the cavity 19 and thus the sacrificial layer 25 is not shown in the resulting interferometric modulators 12 illustrated in Figure 1. Figure 8B illustrates a partially fabricated device including a sacrificial layer 25 formed over the optical stack 16. The formation of the sacrificial layer 25 over the optical stack 16 may include deposition of a xenon difluoride (XeF_2)-etchable material such as molybdenum (Mo) or amorphous silicon (Si), in a thickness selected to provide, after subsequent removal, a gap or cavity 19 (see also Figures 1 and 8E) having a desired design size. Deposition of the sacrificial material may be carried out using deposition techniques such as physical vapor deposition (PVD, e.g., sputtering), plasma-enhanced chemical vapor deposition (PECVD), thermal chemical vapor deposition (thermal CVD), or spin-coating.

[0074] The process 80 continues at block 86 with the formation of a support structure e.g., a post 18 as illustrated in Figures 1, 6 and 8C. The formation of the post 18 may include patterning the sacrificial layer 25 to form a support structure aperture, then depositing a material (e.g., a polymer or an inorganic material, e.g., silicon oxide) into the aperture to form the post 18, using a deposition method such as PVD, PECVD, thermal CVD, or spin-coating. In some implementations, the support structure aperture formed in the sacrificial layer can extend through both the sacrificial layer 25 and the optical stack 16 to the underlying substrate 20, so that the lower end of the post 18 contacts the substrate 20 as illustrated in Figure 6A. Alternatively, as depicted in Figure 8C, the aperture formed in the sacrificial layer 25 can extend through the sacrificial layer 25, but not through the optical stack 16. For example, Figure 8E illustrates the lower ends of the

support posts 18 in contact with an upper surface of the optical stack 16. The post 18, or other support structures, may be formed by depositing a layer of support structure material over the sacrificial layer 25 and patterning portions of the support structure material located away from apertures in the sacrificial layer 25. The support structures may be located within the apertures, as illustrated in Figure 8C, but also can, at least partially, extend over a portion of the sacrificial layer 25. As noted above, the patterning of the sacrificial layer 25 and/or the support posts 18 can be performed by a patterning and etching process, but also may be performed by alternative etching methods.

[0075] The process 80 continues at block 88 with the formation of a movable reflective layer or membrane such as the movable reflective layer 14 illustrated in Figures 1, 6 and 8D. The movable reflective layer 14 may be formed by employing one or more deposition steps, e.g., reflective layer (e.g., aluminum, aluminum alloy) deposition, along with one or more patterning, masking, and/or etching steps. The movable reflective layer 14 can be electrically conductive, and referred to as an electrically conductive layer. In some implementations, the movable reflective layer 14 may include a plurality of sub-layers 14a, 14b, 14c as shown in Figure 8D. In some implementations, one or more of the sub-layers, such as sub-layers 14a, 14c, may include highly reflective sub-layers selected for their optical properties, and another sub-layer 14b may include a mechanical sub-layer selected for its mechanical properties. Since the sacrificial layer 25 is still present in the partially fabricated interferometric modulator formed at block 88, the movable reflective layer 14 is typically not movable at this stage. A partially fabricated IMOD that contains a sacrificial layer 25 may also be referred to herein as an "unreleased" IMOD. As described above in connection with Figure 1, the movable reflective layer 14 can be patterned into individual and parallel strips that form the columns of the display.

[0076] The process 80 continues at block 90 with the formation of a cavity, e.g., cavity 19 as illustrated in Figures 1, 6 and 8E. The cavity 19 may be formed by exposing the sacrificial material 25 (deposited at block 84) to an etchant. For example, an etchable sacrificial material such as Mo or amorphous Si may be removed by dry chemical etching, e.g., by exposing the sacrificial layer 25 to a gaseous or vaporous etchant, such as vapors derived from solid XeF_2 for a period of time that is effective to

remove the desired amount of material, typically selectively removed relative to the structures surrounding the cavity 19. Other etching methods, e.g. wet etching and/or plasma etching, also may be used. Since the sacrificial layer 25 is removed during block 90, the movable reflective layer 14 is typically movable after this stage. After removal of the sacrificial material 25, the resulting fully or partially fabricated IMOD may be referred to herein as a "released" IMOD.

[0077] Figure 9 schematically illustrates an example array 100 of display elements 102. The array 100 can include a plurality of electromechanical display elements 102, which in some implementations may include interferometric modulators. A plurality of segment electrodes or segment lines 122, 124, 126 and a plurality of common electrodes or common lines 112, 114, 116 can be used to address the display elements 102, as each display element will be in electrical communication with a segment electrode and a common electrode. Segment driver circuitry 104 is configured to apply desired voltage waveforms across each of the segment electrodes, and common driver circuitry is configured to apply desired voltage waveforms across each of the column electrodes. In some implementations, some of the electrodes may be in electrical communication with one another, such as segment electrodes 124a and 124b, such that the same voltage waveform can be simultaneously applied across each of the segment electrodes.

[0078] Still with reference to Figure 9, in an implementation in which the display 100 includes a color display or a monochrome grayscale display, the individual electromechanical elements 102 may include subpixels of larger pixels, wherein the pixels include some number of subpixels. In an implementation in which the array includes a color display including a plurality of interferometric modulators, the various colors may be aligned along common lines, such that substantially all of the display elements along a give common line include display elements configured to display the same color. Some implementations of color displays include alternating lines of red, green, and blue subpixels. For example, lines 112 may correspond to lines of red interferometric modulators, lines 114 may correspond to lines of green interferometric modulators, and lines 116 may correspond to lines of blue interferometric modulators. In one implementation, each 3x3 array of interferometric modulators 102 forms a pixel such

as pixels 130a–130d. In the illustrated implementation in which two of the segment electrodes are shorted to one another, such a 3x3 pixel will be capable of rendering 64 different colors (e.g., a 6-bit color depth), because each set of three common color subpixels in each pixel can be placed in four different states. When using this arrangement in a monochrome grayscale mode, the state of the three pixel sets for each color are made to be identical, in which case each pixel can take on four different gray level intensities. It will be appreciated that this is just one example, and that larger groups of interferometric modulators may be used to form pixels having a greater color range at the cost of overall pixel count or resolution.

[0079] Figure 10 is an example system block diagram illustrating a visual display device 40 including a plurality of interferometric modulators. The display device 40 can be, for example, a cellular or mobile telephone. However, the same components or slight variations thereof are also illustrative of various other types of display devices such as televisions, laptop or notebook computers, and portable media players.

[0080] The display device 40 may include a housing, a display array 58, an antenna 43, a speaker 45, an input device 48, and a microphone 46. The housing may generally formed from any of a variety of manufacturing processes, including injection molding, and vacuum forming. In addition, the housing may be made from any of a variety of materials, including but not limited to plastic, metal, glass, rubber, and ceramic, or a combination thereof. In one implementation the housing includes removable portions that may be interchanged with other removable portions of different color, or containing different logos, pictures, or symbols.

[0081] The display array 58 of display device 40 may be any of a variety of displays including a bi-stable display, or interferometric modulator display as described herein. In other implementations, the display 58 includes a flat-panel display, such as plasma, EL, OLED, STN LCD, or TFT LCD as described above, or a non-flat-panel display, such as a CRT or other tube device.

[0082] The illustrated display device 40 can include additional components associated therewith. For example, in one implementation, the display device 40 includes a network interface 27 that includes an antenna 43 which is coupled to a transceiver 47. The transceiver 47 is connected to a processor 56, which is connected to conditioning

hardware 52. The conditioning hardware 52 may be configured to condition a signal (e.g. filter a signal). Conditioning hardware 52 generally includes amplifiers and filters for transmitting signals to the speaker 45, and for receiving signals from the microphone 46. Conditioning hardware 52 may be discrete components within the display device 40, or may be incorporated within the processor 56 or other components.

[0083] The conditioning hardware 52 is connected to a speaker 45 and a microphone 46. The processor 56 is also connected to an input device 48 and a driver controller 29. A power supply (not shown) provides power to all components as required by the particular display device 40 design. The power supply can include a variety of energy storage devices as are well known in the art. For example, in one implementation, the power supply is a rechargeable battery, such as a nickel-cadmium battery or a lithium ion battery. In another implementation, the power supply is a renewable energy source, a capacitor, or a solar cell, including a plastic solar cell, and solar-cell paint. In another implementation, the power supply is configured to receive power from a wall outlet.

[0084] The network interface 27 includes the antenna 43 and the transceiver 47 so that the display device 40 can communicate with one ore more devices over a network. In one implementation the network interface 27 may also have some processing capabilities to relieve requirements of the processor 56. The antenna 43 is any antenna for transmitting and receiving signals. In one implementation, the antenna transmits and receives RF signals according to the IEEE 802.11 standard, including IEEE 802.11(a), (b), or (g). In another implementation, the antenna transmits and receives RF signals according to the BLUETOOTH standard. In the case of a cellular telephone, the antenna is designed to receive CDMA, GSM, AMPS, W-CDMA, or other known signals that are used to communicate within a wireless cell phone network. The transceiver 47 pre-processes the signals received from the antenna 43 so that they may be received by and further manipulated by the processor 56. The transceiver 47 also processes signals received from the processor 56 so that they may be transmitted from the display device 40 via the antenna 43.

[0085] In an alternative implementation, the transceiver 47 can be replaced by a receiver. In yet another alternative implementation, network interface 27 can be replaced by an image source, which can store or generate image data to be sent to the

processor 56. For example, the image source can be a digital video disc (DVD) or a hard-disc drive that contains image data, or a software module that generates image data.

[0086] The input device 48 allows a user to control the operation of the display device 40. In one implementation, input device 48 includes a keypad, such as a QWERTY keyboard or a telephone keypad, a button, a switch, a touch-sensitive screen, a pressure- or heat-sensitive membrane. In one implementation, the microphone 46 is an input device for the display device 40. When the microphone 46 is used to input data to the device, voice commands may be provided by a user for controlling operations of the display device 40.

[0087] The device will typically include host software such as an operating system and one or more application programs that are running on the one or more processors 56 in the device. These host programs define what is to be displayed on the array 58. The processor 56 will generally include an internal memory (not shown) for storing image data, and includes electronic processing circuitry configured to process this image data as defined by one or more software or firmware programs running on the processor 56. In alternate implementations, the memory may be a physically separate component from the processor 56.

[0088] In some implementations, the host software configures the processor to perform a variety of image processing steps. For example, the host processor may configure the processor to store image data to the internal or external memory (not shown) of the device 40. When utilizing external memory, the processor will be in communication with the external memory via an address and data bus (also not shown). A processor with internal memory or a processor in communication with an external memory, along with host software running on processor 56 represents one means for storing image data to an electronic memory.

[0089] The host software may also configure the processor to perform other image processing steps, for example, dithering, quantization, and line doubling steps discussed later. In some implementations, the host software further configures the processor to communicate with the display controller 60 to determine what specific image data is displayed. The host software may also control line multiplying of the

image data before the data is transferred to display controller 60. In other implementations, display controller 60 may control line multiplying.

[0090] Although the host software determines what information is displayed, direct control over the pixels of the array is generally allocated to a display controller 60 and driver circuits 62. Although illustrated as two blocks in Figure 10, these two functions are often part of one controller integrated circuit, as is shown, for example, in Figure 2. As described above, the driver circuits 62 generate and apply the segment and common waveforms of, for example, Figure 5A, in accordance with the display data and line strobe timing required to place the pixels of the array in the state desired by the host software.

[0091] As the host receives and/or generates pixel data for display, it stores that data in a frame buffer 64. The host may have direct access to these memory locations, or it may access them through the display controller 60. The frame buffer 64 may be incorporated into the display controller 60. The display controller 60 reads the memory locations that constitute the frame buffer, and places the data into the correct format and timing to operate the driver circuits 62.

[0092] As noted above, in some displays, the time required to write data to the display elements can place constraints on the overall rate at which the display can be written to. If each common line is separately addressed, the write time necessary for each line will determine the overall frame write time. In some implementations, an increased refresh rate or frame rate of the display may be desired, and may be more important than the resolution or color range of the display for a good visual appearance to a user. In some implementations, driver circuitry and display arrays which are capable of presenting high resolution images with a wide color range may be utilized in a variety of different "modes" of strobing the common lines of the array. These modes may be designed to reduce one or both of the resolution and the color range and in turn increase the potential refresh rate of the display and/or save power consumption by strobing multiple lines of the array at the same time.

[0093] In some implementations, the resolution can be effectively reduced by simultaneously applying the same waveforms across common lines corresponding to display elements of the same color. For example, if a write waveform is simultaneously

applied across red common lines 112a and 112b to address those common lines, the data pattern written to the interferometric modulators along common line 112a will be identical to the data pattern written to the interferometric modulators along common line 112b. If write waveforms are simultaneously applied across green common lines 114a and 114b, and then across blue common lines 116a and 116b, the data pattern written to pixel 130a will be identical to the data pattern written to pixel 130b, causing pixel 130a to display the same color as pixel 130b. Although the term "simultaneously" is used throughout this discussion for the purposes of conciseness, the voltage waveforms need not be perfectly synchronized. As discussed above with respect to Figure 5B, the write waveform may include an overdrive or address voltage during which the potential difference across a display element is sufficient to result in data being written to that display element given an appropriate segment voltage. So long as there is sufficient overlap between the overdrive or address voltages of the write waveforms applied across the common lines and the data signals applied across the segment lines that actuation of the display elements on all of the addressed common lines will occur, the write waveforms and data signals are considered to be applied simultaneously.

[0094] In comparison to a write process in which each common line is individually addressed, data has been written to pixels 130a and 130b in as little as half the time it would have taken to write separate data to pixels 130a and 130b, at the cost of decreased resolution. If this line multiplying process is applied to the remainder of the common lines in the display, the frame write time is considerably reduced.

[0095] Figure 11 is an example of a flowchart illustrating a frame write process 200 which reduces the overall frame write time through the use of line multiplication. This particular frame write process may represent only a portion of the complete frame write, and may occur at the beginning, middle, or end of the complete frame write. Thus, image data may already have been written to one or more common lines within the frame. In block 202, a pair or group of common lines to be simultaneously addressed is identified.

[0096] In block 204, a plurality of data signals are applied along segment lines. Simultaneously, in block 206 a first write waveform is simultaneously applied to at least two common lines in the array to address the waveforms. Such a write waveform

may include, for example, a positive or negative overdrive or address voltage appropriate for the common lines being addressed, as described with respect to Figure 5B above. Hold voltages may be simultaneously applied to multiple common lines not being addressed, and reset voltages may be applied to common lines prior to addressing the common lines. When the write waveform is applied along a pair or group of common lines to be addressed, the application of properly selected data signals along the segment lines will not result in an accidental actuation or accidental release of display elements along common lines not being addressed.

[0097] Although the flowchart of Figure 11 illustrates block 204 as taking place before block 206, the desired actuation will occur so long as there is sufficient overlap between the write waveform and the plurality of data signals to allow all the electromechanical devices sufficient time to actuate or release in accordance with the applied data signals. The frame write time can thus be reduced by maximizing the overlap between the write waveform of block 206 and the data signals of block 204, and block 204 and 206 can occur in either order so long as there is overlap between the application of the signals.

[0098] In block 208, a determination is made as to whether any additional pairs or groups of common lines are to be simultaneously addressed. If so, the process returns to block 202 to select an appropriate pair or group of common lines to simultaneously address. If not, the process moves to further blocks which could include a termination of the frame write process if all necessary common lines have been addressed, or could include individual addressing of certain common lines. In addition, simultaneous addressing of pairs or groups of common lines may be interspersed with individual addressing of common lines, depending on the nature of the data to be written. For example, if a portion of the image data written to a display includes text or another still image, and another portion of the data includes a video which can be displayed at a lower resolution and which is located vertically between sections of text or still image, the portions of the display located above the video can be written by individually addressing those common lines, the portions of the display including the video can be written at a lower resolution by utilizing a line multiplying write process, and the write

process may return to individual addressing of the common lines of the display for the portion of the display located below the video.

[0099] The particular method of line multiplication discussed above can apply identical write waveforms to common lines in adjacent pixels, although other pairs of common lines may be simultaneously addressed in other implementations. Furthermore, even if the line multiplying method is used to simultaneously apply write waveforms to common lines in adjacent pixels, all of the lines in a given pair or group of pixels need not be written before writing lines in other groups of pixels. In some implementations, multiple pairs or groups of common lines of the same color can be addressed before addressing common lines of another color. For example, red common lines 112a and 112b may be simultaneously addressed, followed by a subsequent write process which simultaneously addresses red common lines 112c and 112d. Because different voltage waveforms may be used to address common lines of different color display elements, utilize the write waveform appropriate for a particular color for multiple pairs or groups of common lines before addressing common lines of another color. In some implementations, any number of pairs or groups of common lines of a given color may be addressed before addressing common lines of another color. For example, in some implementations 5 pairs or groups of common lines of a given color may be addressed before common lines of another color are addressed, although larger or smaller numbers of pairs or groups may be used, as well.

[0100] In addition, although the simultaneous application of substantially identical waveforms to two common lines is discussed herein, further increases in refresh rate or frame write or reductions in power usage may be achieved by simultaneously applying substantially identical waveforms to more than two common lines.

[0101] In some methods of updating data on a display, charge buildup on particular display elements may be reduced by altering the polarity of the write waveforms applied to the common line. In one implementation, which may be referred to as frame inversion, a given frame is fully addressed using write waveforms of a particular polarity, and a subsequent frame is fully addressed using write waveforms of the opposite polarity. In other implementations, however, the polarity of write waveforms may be altered during a single frame write. In another implementation, which may be referred to

as line inversion, the polarity of the write may be altered after addressing each line, and the polarity used to address a particular line will be changed in subsequent frames. If the display is being updated in a substantially linear fashion, this may result in adjacent lines being addressed by write voltages having opposite polarities. Thus, in some implementations, it may be advantageous to utilize a given write waveform having a given polarity to write to, for example, every other red common line with a positive polarity for some number of common lines, before writing to the skipped red common lines with a negative polarity.

[0102] Polarity inversion within a frame can be applied to a write process in which line multiplying is used as well. In one implementation, red lines 112c and 112d may be addressed using the opposite polarity of that used to address red lines 112a and 112b within a given frame write. In an implementation, such as the one described above, where a write waveform with a given polarity is used for multiple sequential addressing operations, red lines 112a and 112b may be addressed using a first polarity, and red lines 112c and 112d may be skipped while some number of additional pairs or groups of red lines are written using the first polarity. After some number of pairs or groups have been addressed using the first polarity, red lines 112c and 112d may be addressed using the opposite polarity.

[0103] If polarity inversion is utilized, addressing a certain number of lines of one color using a first polarity need not be followed by addressing a certain number of lines in the same color using the opposite polarity. In some other implementations, positive red write processes may be followed by, for example, negative blue write processes, or positive green write processes.

[0104] The above description sets forth the methods and benefits of writing identical data to multiple common lines simultaneously. Turning now to Figures 12–22, methods of processing image data to produce the multiplied lines will be described.

[0105] Figure 12 illustrates an example 12x16 array of pixel data. The image data 142 can be arranged as a row-column array of pixel data, with each element of pixel data designated $P_{row,column}$ at each location in the array. In this example, there are 12 rows and 16 columns of pixel data. Of course, in an actual display more rows and columns will generally be provided, with 768 rows of pixels and 1024 columns of pixels being one

common implementation. Each pixel data element P may be formed of three different color sub-pixel data elements, which may include red, green, and blue sub-pixel data elements. Thus, as shown in Figure 12, pixel data $P_{2,2}$ is made up of red sub-pixel data $R_{2,2}$, green sub-pixel data $G_{2,2}$, and blue sub-pixel data $B_{2,2}$. A display device that displays this image will therefore have 12 rows of red display elements, 12 rows of green display elements, and 12 rows of blue display elements interleaved with each other. This is the same format as shown in the physical display of Figure 9, where each row of pixels includes three "sub-rows," one of each color. The pixel data at each location may be of any number of tone-levels, e.g., 2 bits per color, 4 bits per color, 6 bits per color, 8 bits per color, or any other value.

[0106] One implementation for creating doubled lines is to substitute the even rows of data with copies of the odds row of data. Figure 13 illustrates an example line multiplied array derived from the array of Figure 12. In Figure 13, the six odd rows of pixel data from Figure 12 are used to fill the entire twelve rows of image data. Row 1 data is used in both row 1 and row 2, row 3 data is used in both row 3 and row 4, etc. A person having ordinary skill in the art will readily appreciate that the original odd rows of data could be substituted with copies of the even rows of data instead. Another implementation is to average the data of adjacent common color rows, and use that data for both of the rows that were averaged together. In this case, rows 1 and 2 would each contain the average of original rows 1 and 2. As discussed above, this line doubling enables the application of simultaneous waveforms across multiple common lines, thus increasing the maximum possible refresh rate or frame rate.

[0107] Further shown in Figure 13 is an expanded portion 144 of several pixels, showing the red, green, and blue subpixel values. In the expanded view 144 of five pixels of the array, the pixel boundaries are marked with solid lines 145 and dashed lines 147 for purposes of illustration. For the illustrated set of pixels, the data for all three colors is copied above and below each dashed line. The image data changes across solid line pixel boundaries. Thus, the line doubling essentially turns the original square pixels into rectangular pixels with a long side extending between the solid lines 145 in the direction of the doubling. Because of this loss of resolution in the line multiplied image of Figure 13, visual artifacts are created, especially near the edges of objects in the

image where brightness transitions occur. Display of text is especially susceptible to visual artifacts caused by line doubling in this manner.

[0108] Digital image processing techniques may be employed to mitigate or eliminate the adverse effects of line doubling. These techniques may be implemented by host software executing, for example, on processor 56 of Figure 10, or similarly on processor 21 of Figure 2. Figure 14 shows an example system block diagram illustrating a visual display device including a plurality of imaging pipeline components. Processor 1410 of device 1400 is connected to working memory 1420 and nonvolatile memory 1440. Nonvolatile memory 1440 stores host software including multiple modules as shown. Some of the host software modules implement digital processing techniques that may improve the quality of a display utilizing line doubling.

[0109] Modules stored in nonvolatile memory 1440 include half-height module 1450, dithering module 1460, quantization module 1470, and line multiplying module 1480. The image processing modules comprise instructions that configure the processor 1410 to perform digital image processing tasks. These tasks may include reducing a full height image to a half height image, dithering of the half height image, quantizing the dithered image, and line doubling the quantized image. Instructions in the image processing modules may also configure the processor to read and write the working memory 1420 to store and load image data as it is processed through the various steps of a digital image pipeline. Instructions in the line multiplying module 1480 may also update an electronic display with image data stored in the working memory 1420. The line multiplying module 1480 may also utilize instructions in the operating system module 1490 discussed below to interface either directly or indirectly with the electronic display.

[0110] Another module stored in nonvolatile memory 1440 include a frame rate setting module 1495. This module includes instructions that configure the processor to determine whether the display frame stream is dynamic or static in nature, and adapts the frame rate of the display as desired. This module also adapts the image pipeline processes to select a dithering method to be used on the frames before they are displayed on an electronic display, and determines whether or not to line double the frame. Thus, the frame rate setting module 1495 provides for a variable frame rate display system, and

therefore comprises one means for determining a display mode based on whether a current frame stream is static or dynamic.

[0111] Processor 1410 also executes instructions contained in operating system module 1490 to manage the hardware and software resources of device 1400. Instructions in operating system module 1490 may also provide a consistent interface to the instructions in the digital image processing modules and other modules.

[0112] Processor 1410 is also connected to display controller I/O. This may include a direct connection to display control hardware, as illustrated by display controller 60 of device 40 of Figure 10, or other means to control a display. One or more of the modules located in the nonvolatile memory 1440 of this implementation may include instructions that cause the processor to communicate with the display controller so as to display data on an electronic display as described above.

[0113] Figure 15 shows the data flow in one implementation of a variable frame rate display system. The variable rate display system may be used to adapt the frame rate of a display based on a characterization of the information to be displayed. For example, in one implementation, the frame rate of a H.264 video stream may exceed the maximum frame rate of the display when the display is in single line addressing mode. To match the frame rate of the H 264 video stream, the frame rate setting module may select a fast frame update mode. The fast frame update mode may include multi-line addressing to achieve a faster frame rate. The fast frame update mode may also result in effectively halving the display resolution.

[0114] In other implementations, a touch screen input may provide for rapid display changes. For example, a "pinch to zoom" input may display a series of rapid images as an image is resized. The pinch input may place the display into a low resolution, fast update mode. The lower resolution, faster update mode may provide for a more smooth sense of motion as the image is zoomed with the touch input. When the zoom operation is complete, the display may return to a full resolution mode.

[0115] The data flow begins at the far left of Figure 15 with a series of display frames. The display frames may be generated by software running locally with display system 1500 or may be received from an external source. The display frames are ordered in time series, with the current frame 1545 displayed first, and frames higher in the figure

to be displayed progressively later. In some implementations, the series of display frames may be part of a video stream. Alternatively, in some implementations, the series of display frames may be repetitions of a static image to be displayed.

[0116] The frame rate setting module 1530 may examine two or more frames 1545 and 1560 to determine an appropriate frame rate for electronic display 1550. The frame rate setting module may also utilize other information to determine an appropriate frame rate. For example, the frame rate setting module 1530 may set the frame rate based on the type of video stream being displayed. As described earlier, a H 264 video stream may cause the frame rate setting module to select a particular frame rate.

[0117] In some implementations, the system may compare the current frame to a previous frame or a subsequent frame to characterize the current frame as static or dynamic. In some implementations, if the two frames are different, the current frame is characterized as dynamic. If the two frames are equivalent, the current frame is characterized as static. The frame rate setting module 1530 may then select a frame rate based at least in part on the characterization of the current frame. For example, if the current frame is characterized as static, a slower frame rate with higher resolution may be selected. If the current frame is characterized as dynamic, a faster frame rate with lower resolution may be selected.

[0118] Other implementations may examine more than two frames when characterizing the current frame as static or dynamic. For example, as shown by the frames grouped by dotted line 1590 of Figure 15, the frame rate setting module may look ahead several frames into the input frame stream to characterize the current frame. In some implementations, if any of the frames examined are different, the current frame may be characterized as dynamic. In these implementations, if all the frames are equivalent, the current frame may be characterized as static.

[0119] Considering more than two frames may enable improved image quality in some implementations. For example, flicker caused by unnecessary transitions in frame rate may be reduced by examining enough frames to detect transient events in the frame stream that should not be used as a basis for frame rate changes.

[0120] Other implementations may avoid unnecessary flicker by maintaining a minimum "settle time" before a mode switch. With this approach, once a display mode

switch has occurred, no additional switching is done for some minimum time, regardless of the nature of the frame stream. An appropriate settle time may be chosen to avoid both the display of lower resolution still images for a prolonged period, or "jerky" motion effects resulting from a frame rate that is too slow to adequately convey motion.

[0121] Once the desired display mode or frame rate is known, the frame rate setting module 1530 may determine whether the current frame is necessary to maintain the desired frame rate. If the frame is necessary, frame rate setting module 1530 may forward the frame to either the line multiplying image pipeline 1510 or the non-line multiplying image pipeline 1515. In some implementations, the input frame rate provided by the series of display frames at the far left of the dataflow diagram may exceed the display frame rate even for the fastest frame rate supported by the image pipeline. In this case, the frame rate setting module 1530 will discard some frames, although the rate at which frames are discarded may vary with display mode. Additionally, if the frame stream is characterized as static by the frame rate setting module 1530, the frame rate of the display may be lower than the frame rate provided by the series of display frames. In these circumstances, unnecessary frames may be discarded before processing them further in the image pipeline.

[0122] Once the frame rate of the input stream has been matched to the frame rate of the display, the frame rate setting module 1530 will transfer dynamic frames to be displayed at a fast update rate to the line multiplying image pipeline 1510. Static frames to be displayed at a slower update rate will be transferred to the non line multiplying image pipeline 1520. Each image pipeline may process the frames differently. For example, the dithering methods utilized by each pipeline may be different. Additionally, the line doubling image pipeline may create a half height version of the image and line multiplying the image. This may not be performed in the non line doubling image pipeline.

[0123] In some implementations, static frames are transferred to a simple non line multiplying image pipeline 1515. This image pipeline dithers and quantizes the frame, and then displays the frame on the display 1550 at a time defined by the static frame rate. When the selected image pipeline completes processing of the frame, the frame is displayed on electronic display 1550. By selecting which image pipeline will

process a frame, the frame rate setting module determines the dithering method used for the frame.

[0124] Figure 16 shows the data flow of image data as it is processed by one implementation of a line multiplying image processing pipeline. This data flow corresponds to that illustrated within the line multiplying image pipeline 1510 of Fig. 15. At the far left of Figure 16 is a source image, typically dimensioned to be compatible with a target electronic display. The source image may be stored in an electronic memory, such as the memory 25 of system 30 in Figure 2 or the processor internal memory (not shown) or external memory (not shown) of device 40 in Figure 10, or the working memory 1420 of Figure 14.

[0125] In an implementation of an image pipeline utilizing line multiplying, such as image pipeline 1510 of Fig 15, this source image is typically first reduced to a half height image. As described above, adjacent lines of the source image may be averaged to produce one line of the half height image (b). Alternatively, in other implementations, alternate lines of source image (a) may be selected to form half height image (b), with the remaining lines simply omitted from image (b). While some implementations may create physical copies of the half height image, other implementations may choose to virtualize the half height image via well known processing methods. For example, by utilizing object oriented programming methods, an image object may provide both full height and half height image accessors, while only maintaining one physical copy of image data.

[0126] Regardless of the specific implementation details chosen by a particular implementation, once a half height image is provided, some implementations of host software may dither the half height image, as illustrated by Figure 16 (c). Dithering reduces the banding effects caused by quantization, and is the intentional addition of noise to an image to improve display characteristics. Dithering is discussed in more detail below.

[0127] Quantization is performed in some implementations of host software, and is illustrated by the transformation of the dithered image of Figure 16(c) to the quantized image of Figure 16(d). Quantization is necessary in some implementations to adapt the color depth of the source image data to a color depth supported by the target

display. For example, some implementations of the IMOD displays render two bits per sub-pixel of color information, corresponding to 64 total colors. However, the source image may support up to 24 bits of color information, typically arranged as eight bits per pixel. When the electronic display's color depth is lower than a source image data's color depth, the host software's image pipeline utilizes a quantization process to convert the color depth of the source image to one compatible with the target display.

[0128] While quantization is often necessary, it may produce banding effects in the displayed image. These effects, known as quantization error, arise when adjacent pixels with similar values are mapped to an identical new pixel value. Dithering improves image quality by adding noise to the image that disrupts the visual patterns that would otherwise result from the quantization error caused when similarly valued pixels are rounded during the quantization process. A host program dithering method is dependent on the dither noise signal. The dither noise signal can be represented by a dither mask. To apply the noise signal to the target image, in one implementation, instructions in the host program "tile" the dither mask to match the dimensions of the target image. Therefore, the dithering module 1460 represents one means for dithering a frame using a dithering method to produce a dithered frame.

[0129] Figure 17 is a flow chart illustrating one implementation of a line multiplying image pipeline. Process 1700 of Figure 17 may be implemented by instructions within the line multiplying image pipeline 1510 of figure 15, and the image processing modules of Figure 14, to include the half height module 1450, dithering module 1460, quantization module 1470, or the line doubling module 1480. Process 1700 begins at start state 1705 and then moves to block 1710 where instructions implementing block 1710 cause a source image to be stored to an electronic memory. In some implementations, the electronic memory may include the working memory 1420 of Figure 14, or the memory 25 of Figure 1. Process 1700 then moves to block 1720 where a half height image is generated. In one implementation, the half height image may be generated by instructions in the half-height module 1450 of Figure 14. Next, process 1700 moves to block 1730. In block 1730, a half height image is dithered. The dithering may be performed in one implementation by the instructions of dithering module 1460 of Figure 14. Next, process 1700 moves to block 1740. Here, the dithered image is

quantized. Quantization of the half height image may be performed in one implementation by the instructions of quantization module 1470 of Figure 14. Process 1700 then moves to block 1760 where the quantized image is line multiplied. Line multiplying may be performed in one implementation by instructions contained in the line multiplying module 1480 of Figure 14. Process 1700 then moves to end state 1770.

[0130] Figure 18 shows three example noise signals and their corresponding discrete Fourier transforms used to dither images in various implementations. The top row of masks, labeled (a)(1), (a)(2), and (a)(3), illustrate three noise masks that may be used to dither images. These masks have a noise value at each position in the mask. The shading of each square in the noise mask indicates the amplitude of the noise value at that position in the mask, with lighter shading denoting a larger value. The dithering process first tiles the noise mask on the image being dithered. Then, the dithering process adds the noise value at each position in the tiled mask to the value of the corresponding pixel in the image being dithered, with the result overwriting the pixel value in the target image.

[0131] Below each noise mask in the top row of Figure 18 is a graphical representation of the discrete Fourier transform of the mask's noise signal, illustrating the distribution of frequency components of the noise mask around zero frequency. Figure 18, image (a)(2) illustrates the discrete Fourier transform of the noise signal shown in image (a)(1), consisting of white noise. As shown, the large and small frequency component amplitudes for the noise mask of Figure 18(a)(1) are distributed across the two dimensional frequency spectrum.

[0132] Figure 19 shows examples of a variety of images produced by different implementations of image processing components. Image (a) represents an original source image, while image (b) shows the image after a quantization process without any dithering performed on the image beforehand. The banding effects are created by quantization error, which arises when adjacent pixels with similar values are mapped to an identical new pixel value.

[0133] Utilization of the noise signal represented by Figure 18, image (a)(1) to dither an image prior to quantization results in a visual appearance similar to that shown in Figure 19 (c). While the banding caused by quantization errors has been

eliminated, use of the white noise signal of Figure 18 image (a)(1) results in image artifacts caused by the low frequency components of the noise signal. This is especially noticeable in the sky portion of the image.

[0134] By utilizing a noise signal with primarily high frequency components, the artifacts evident in Fig. 19, image (c) can be reduced. Figure 18, image (b)(1) is a noise signal including primarily high frequency components. This is illustrated by its discrete Fourier transform, image (b)(2). The dark region in the center of image (b)(2) illustrates the low amplitude of low frequency components around zero frequency in the noise mask of Figure 18(b)(1).

[0135] Applying the noise signal of Figure 18, image (b)(1) to the image of Figure 19 (a) prior to quantization results in an improvement compared to Figure 19, image (c), which utilized the white noise signal of Figure 18, image (a)(1). These results can be seen in Figure 19 (d). Dithering with high frequency noise (also known as blue noise) achieves these results by taking advantage of characteristics of the human visual system. The human visual system is less sensitive to high frequency noise than low frequency noise. By dithering with high frequency noise, the visual artifacts introduced by the quantization process in pushed into the higher frequencies, where the human visual system is less sensitive.

[0136] However, in implementations using image processing that performs dithering and line multiplying, as illustrated by process 1700 of Figure 17, there are additional challenges to ensure high quality images. These challenges cannot be met by use of the high frequency noise mask described above, because the high frequency dither noise signal of Fig. 18, image (b)(1) does not account for additional visual effects caused by line multiplying.

[0137] The dithering operation presents special challenges when line multiplying, partially because it must be performed before line multiplying. Since dithering introduces random noise to an image, dithering of a full height line multiplied image would cause each line of the image to become unique.

[0138] Therefore, in implementations where it is desirable to have identical multiplied lines to realize performance advantages, host programs may perform the dithering process at a point in the image pipeline before the line multiplying occurs.

However, when an image is dithered before it is line multiplied, the subsequent multiplying operation creates effects in the dithered image. These effects reduce image quality. For example, because line multiplying scales an image in one dimension, it also reduces the frequency of any noise signal along the multiplied dimension of the image by one half.

[0139] This lower frequency noise has undesirable visual effects, illustrated by image (e) of Figure 19. While the non-line multiplied image of image (d) exhibits high image quality as a result of the high frequency noise mask, the line multiplied image of image (e) exhibits reduced visual quality. The lower frequency noise resulting from the multiplying process in the vertical dimension allows the human visual system to perceive quantization errors.

[0140] To compensate for the effect of the line doubling on the dithering noise signal, the dither noise signal can be tailored to provide improved results after the line doubling operation occurs. Since the image is multiplied in only one dimension, the dithering may introduce a noise signal with fewer low frequency components in the multiplied dimension, while maintaining frequency components similar to that of the signal represented by the discrete Fourier transform of Fig. 18 (b)(2) in the non multiplied dimension. Dithering the reduced height image with a noise signal meeting these characteristics will result in a line multiplied image having an appropriate noise signal in both dimensions. A dither noise signal with such characteristics and its discrete Fourier transform are illustrated by Fig. 18, image (c)(1) and (c)(2). Fig. 18, image (c)(2) illustrates that the noise mask of Fig. 18, image (c)(1) has a two dimensional frequency spectrum that is radially asymmetric around zero frequency. The dark, elliptically shaped portion of image (c)(2) illustrates increased suppression of low frequency components along the vertical dimension of the noise mask represented by image (c)(1) as compared to the horizontal dimension of noise mask (c)(1).

[0141] The results of dithering with the noise signal of Figure 18, image (c)(1) are illustrated in Figure 19, image (f). Relative to image (e) which was produced with the symmetric high frequency noise signal of Figure 18, image (b)(1), Figure 19 image (f) illustrates a reduction in visible effects. This improvement is the result of the high frequency noise that remains in both the horizontal and vertical dimension of the

image after the lines have been multiplied. The asymmetric noise signal, while producing an asymmetric noise pattern in the non-multiplied image, results in symmetric high frequency noise in the multiplied image.

[0142] Figure 20 is a flow chart illustrating one implementation of a variable frame rate image processing system with dynamic dither selection. Some portions of process 2000 may be implemented in some implementations by instructions in the frame rate setting module 1495 of Figure 14. Process 2000 begins at start block 2010 and then moves to decision block 2020, where a determination is made as to whether the current frame should be treated as a dynamic frame or a static frame. For example, the current frame may be determined to be a static frame if it is equivalent to a previous frame or frames. The current frame may be determined to be a dynamic frame if it is different from a previous frame or frames. If the frame is dynamic, process 2000 moves to decision block 2070. In decision block 2070, process 2000 determines whether the current frame is necessary to maintain the current frame rate for the dynamic frame stream. This decision may be based on how recently the last display update was performed, the frame rate of the input frame stream, or other factors. If the current frame is not required to maintain the current display frame rate, process 2000 moves to block 2060, and the current frame is discarded. Process 2000 then moves to end state 2095.

[0143] If the current frame is required to maintain the current display frame rate, process 2000 moves to block 2075, where the current frame is converted into a half height image in the illustrated implementation. Block 2075 may be performed by instructions included in the half height module 1450 of Figure 14. Process 2000 then moves to block 2080, and the current frame is dithered with a noise signal tailored for the half height image. Some implementations may utilize a noise signal with fewer low frequency components in one dimension, while maintaining frequency components similar to that of the signal represented by the discrete Fourier transform of Figure 18 (b)(2) in another dimension.. Block 2080 may be performed by instructions included in the dithering module 1460 of Figure 14. Process 2000 then moves to block 2085, where the half height frame is quantized to match the color depth of the frame with the color depth of the target display. Block 2085 may be performed by instructions included in the quantization module 1470 of Figure 14. Process 2000 then moves to block 2090 where

the quantized frame is line multiplied. Instructions implementing block 2090 may be included in the line multiplying module 1480 of Figure 14.

[0144] Returning to decision block 2020, if the current frame is part of a static frame stream, process 2000 moves from block 2020 to decision block 2030. Whether the current frame is required to maintain the target frame rate for a static frame stream is determined in decision block 2020. If the frame is not required, process 2000 moves to block 2060 and the frame is discarded. Process 2000 then moves to end block 2095.

[0145] Returning to decision block 2030, if the current frame is needed to maintain the target frame rate, process 2000 moves to block 2040 where the current frame is dithered with a noise signal tailored for the static frame. Some implementations may use a noise signal with high frequency components symmetrically distributed around zero frequency. Dithering of the frame may be implemented in some implementations by instructions included in the dithering module 1460 of Figure 14. Process 2000 then moves to block 2050. In block 2050, the frame is quantized to match the frame color depth to the target display color depth. Quantization of the frame may be performed by instructions included in the quantization module 1470 of Figure 14. Process 2000 then moves to end state 2095.

[0146] Figure 21 shows the selection of a dither noise signal as a frame stream transitions between a dynamic and static character. In Figure 21, a frame stream advances from left to right, with the oldest frames to the left and newest frames to the right. Each frame is identified with a letter, where frames labeled with the same letter are assumed to be characterized as similar or equivalent. The vertical bars 2130 and 2140 indicate a display mode switch, as determined by the characterization of the frame stream to the immediate right of the bar.

[0147] As illustrated, the two frames to the left labeled "A" are static frames. Because the two "A" frames are part of a frame stream characterized as static, a slower update rate without line multiplication is used, and a noise signal with a symmetric high frequency mask 2150 is used in this implementation to dither those frames.

[0148] The character of the stream transitions to a dynamic stream to the right of vertical bar 2130. This is the result of a change in the frame image pattern illustrated by double arrow 2110 from "A" to "B", "C", "D", and "E". Because the frame stream

has become dynamic, a higher frame rate is used to adequately convey motion on the display in the illustrated implementation. As such, the image pipeline transitions into a line multiplying mode, and a dither noise signal with asymmetric frequency components 2160 is used for frames in the middle region of the illustrated frame stream. Finally, after vertical line 2140, the frame stream is again characterized as static. The image pipeline stops line multiplying, slows the frame update rate, and a noise signal with a symmetric noise mask 2170 is used to dither the frames.

[0149] Figures 22A and 22B show examples of system block diagrams illustrating a display device 40 that includes a plurality of interferometric modulators. The display device 40 can be, for example, a cellular or mobile telephone. However, the same components of the display device 40 or slight variations thereof are also illustrative of various types of display devices such as televisions, e-readers and portable media players.

[0150] The display device 40 includes a housing 41, a display 30, an antenna 43, a speaker 45, an input device 48, and a microphone 46. The housing 41 can be formed from any of a variety of manufacturing processes, including injection molding, and vacuum forming. In addition, the housing 41 may be made from any of a variety of materials, including, but not limited to: plastic, metal, glass, rubber, and ceramic, or a combination thereof. The housing 41 can include removable portions (not shown) that may be interchanged with other removable portions of different color, or containing different logos, pictures, or symbols.

[0151] The display 30 may be any of a variety of displays, including a bi-stable or analog display, as described herein. The display 30 also can be configured to include a flat-panel display, such as plasma, EL, OLED, STN LCD, or TFT LCD, or a non-flat-panel display, such as a CRT or other tube device. In addition, the display 30 can include an interferometric modulator display, as described herein.

[0152] The components of the display device 40 are schematically illustrated in Figure 19. The display device 40 includes a housing 41 and can include additional components at least partially enclosed therein. For example, the display device 40 includes a network interface 27 that includes an antenna 43 which is coupled to a transceiver 47. The transceiver 47 is connected to a processor 21, which is connected to

conditioning hardware 52. The conditioning hardware 52 may be configured to condition a signal (e.g., filter a signal). The conditioning hardware 52 is connected to a speaker 45 and a microphone 46. The processor 21 is also connected to an input device 48 and a driver controller 29. The driver controller 29 is coupled to a frame buffer 28, and to an array driver 22, which in turn is coupled to a display array 30. A power supply 50 can provide power to all components as required by the particular display device 40 design.

[0153] The illustrated display device 40 can include additional components associated therewith. For example, in one implementation, the display device 40 includes a network interface 27 that includes an antenna 43 which is coupled to a transceiver 47. The transceiver 47 is connected to a processor 56, which is connected to conditioning hardware 52. The conditioning hardware 52 may be configured to condition a signal (e.g. filter a signal). Conditioning hardware 52 generally includes amplifiers and filters for transmitting signals to the speaker 45, and for receiving signals from the microphone 46. Conditioning hardware 52 may be discrete components within the display device 40, or may be incorporated within the processor 56 or other components.

[0154] The conditioning hardware 52 is connected to a speaker 45 and a microphone 46. The processor 56 is also connected to an input device 48 and a driver controller 29. A power supply (not shown) provides power to all components as required by the particular display device 40 design. The power supply can include a variety of energy storage devices as are well known in the art. For example, in one implementation, the power supply is a rechargeable battery, such as a nickel-cadmium battery or a lithium ion battery. In another implementation, the power supply is a renewable energy source, a capacitor, or a solar cell, including a plastic solar cell, and solar-cell paint. In another implementation, the power supply is configured to receive power from a wall outlet.

[0155] The network interface 27 includes the antenna 43 and the transceiver 47 so that the display device 40 can communicate with one ore more devices over a network. In one implementation the network interface 27 may also have some processing capabilities to relieve requirements of the processor 56. The antenna 43 is any antenna for transmitting and receiving signals. In one implementation, the antenna transmits and receives RF signals according to the IEEE 802.11 standard, including IEEE 802.11(a), (b), or (g). In another implementation, the antenna transmits and receives RF signals

according to the BLUETOOTH standard. In the case of a cellular telephone, the antenna is designed to receive CDMA, GSM, AMPS, W-CDMA, or other known signals that are used to communicate within a wireless cell phone network. The transceiver 47 pre-processes the signals received from the antenna 43 so that they may be received by and further manipulated by the processor 56. The transceiver 47 also processes signals received from the processor 56 so that they may be transmitted from the display device 40 via the antenna 43.

[0156] In some implementations, the transceiver 47 can be replaced by a receiver. In addition, the network interface 27 can be replaced by an image source, which can store or generate image data to be sent to the processor 21. The processor 21 can control the overall operation of the display device 40. The processor 21 receives data, such as compressed image data from the network interface 27 or an image source, and processes the data into raw image data or into a format that is readily processed into raw image data. The processor 21 can send the processed data to the driver controller 29 or to the frame buffer 28 for storage. Raw data typically refers to the information that identifies the image characteristics at each location within an image. For example, such image characteristics can include color, saturation, and gray-scale level.

[0157] The processor 21 can include a microcontroller, CPU, or logic unit to control operation of the display device 40. The conditioning hardware 52 may include amplifiers and filters for transmitting signals to the speaker 45, and for receiving signals from the microphone 46. The conditioning hardware 52 may be discrete components within the display device 40, or may be incorporated within the processor 21 or other components.

[0158] The driver controller 29 can take the raw image data generated by the processor 21 either directly from the processor 21 or from the frame buffer 28 and can re-format the raw image data appropriately for high speed transmission to the array driver 22. In some implementations, the driver controller 29 can re-format the raw image data into a data flow having a raster-like format, such that it has a time order suitable for scanning across the display array 30. Then the driver controller 29 sends the formatted information to the array driver 22. Although a driver controller 29, such as an LCD controller, is often associated with the system processor 21 as a stand-alone Integrated

Circuit (IC), such controllers may be implemented in many ways. For example, controllers may be embedded in the processor 21 as hardware, embedded in the processor 21 as software, or fully integrated in hardware with the array driver 22.

[0159] The array driver 22 can receive the formatted information from the driver controller 29 and can re-format the video data into a parallel set of waveforms that are applied many times per second to the hundreds, and sometimes thousands (or more), of leads coming from the display's x-y matrix of pixels.

[0160] In some implementations, the driver controller 29, the array driver 22, and the display array 30 are appropriate for any of the types of displays described herein. For example, the driver controller 29 can be a conventional display controller or a bi-stable display controller (e.g., an IMOD controller). Additionally, the array driver 22 can be a conventional driver or a bi-stable display driver (e.g., an IMOD display driver). Moreover, the display array 30 can be a conventional display array or a bi-stable display array (e.g., a display including an array of IMODs). In some implementations, the driver controller 29 can be integrated with the array driver 22. Such an implementation is common in highly integrated systems such as cellular phones, watches and other small-area displays.

[0161] In some implementations, the input device 48 can be configured to allow, e.g., a user to control the operation of the display device 40. The input device 48 can include a keypad, such as a QWERTY keyboard or a telephone keypad, a button, a switch, a rocker, a touch-sensitive screen, or a pressure- or heat-sensitive membrane. The microphone 46 can be configured as an input device for the display device 40. In some implementations, voice commands through the microphone 46 can be used for controlling operations of the display device 40.

[0162] The power supply 50 can include a variety of energy storage devices as are well known in the art. For example, the power supply 50 can be a rechargeable battery, such as a nickel-cadmium battery or a lithium-ion battery. The power supply 50 also can be a renewable energy source, a capacitor, or a solar cell, including a plastic solar cell or solar-cell paint. The power supply 50 also can be configured to receive power from a wall outlet.

[0163] In some implementations, control programmability resides in the driver controller 29 which can be located in several places in the electronic display system. In some other implementations, control programmability resides in the array driver 22. The above-described optimization may be implemented in any number of hardware and/or software components and in various configurations.

[0164] The various illustrative logics, logical blocks, modules, circuits and algorithm steps described in connection with the implementations disclosed herein may be implemented as electronic hardware, computer software, or combinations of both. The interchangeability of hardware and software has been described generally, in terms of functionality, and illustrated in the various illustrative components, blocks, modules, circuits and steps described above. Whether such functionality is implemented in hardware or software depends upon the particular application and design constraints imposed on the overall system.

[0165] The hardware and data processing apparatus used to implement the various illustrative logics, logical blocks, modules and circuits described in connection with the aspects disclosed herein may be implemented or performed with a general purpose single- or multi-chip processor, a digital signal processor (DSP), an application specific integrated circuit (ASIC), a field programmable gate array (FPGA) or other programmable logic device, discrete gate or transistor logic, discrete hardware components, or any combination thereof designed to perform the functions described herein. A general purpose processor may be a microprocessor, or, any conventional processor, controller, microcontroller, or state machine. A processor may also be implemented as a combination of computing devices, e.g., a combination of a DSP and a microprocessor, a plurality of microprocessors, one or more microprocessors in conjunction with a DSP core, or any other such configuration. In some implementations, particular steps and methods may be performed by circuitry that is specific to a given function.

[0166] In one or more aspects, the functions described may be implemented in hardware, digital electronic circuitry, computer software, firmware, including the structures disclosed in this specification and their structural equivalents thereof, or in any combination thereof. Implementations of the subject matter described in this

specification also can be implemented as one or more computer programs, i.e., one or more modules of computer program instructions, encoded on a computer storage media for execution by, or to control the operation of, data processing apparatus.

[0167] If implemented in software, the functions may be stored on or transmitted over as one or more instructions or code on a computer-readable medium. The steps of a method or algorithm disclosed herein may be implemented in a processor-executable software module which may reside on a computer-readable medium. Computer-readable media includes both computer storage media and communication media including any medium that can be enabled to transfer a computer program from one place to another. A storage media may be any available media that may be accessed by a computer. By way of example, and not limitation, such computer-readable media may include RAM, ROM, EEPROM, CD-ROM or other optical disk storage, magnetic disk storage or other magnetic storage devices, or any other medium that may be used to store desired program code in the form of instructions or data structures and that may be accessed by a computer. Also, any connection can be properly termed a computer-readable medium. Disk and disc, as used herein, includes compact disc (CD), laser disc, optical disc, digital versatile disc (DVD), floppy disk, and blu-ray disc where disks usually reproduce data magnetically, while discs reproduce data optically with lasers. Combinations of the above should also be included within the scope of computer-readable media. Additionally, the operations of a method or algorithm may reside as one or any combination or set of codes and instructions on a machine readable medium and computer-readable medium, which may be incorporated into a computer program product.

[0168] Various modifications to the implementations described in this disclosure may be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other implementations without departing from the spirit or scope of this disclosure. Thus, the disclosure is not intended to be limited to the implementations shown herein, but is to be accorded the widest scope consistent with the claims, the principles and the novel features disclosed herein. The word "exemplary" is used exclusively herein to mean "serving as an example, instance, or illustration." Any implementation described herein as "exemplary" is not necessarily to be construed as

preferred or advantageous over other implementations. Additionally, a person having ordinary skill in the art will readily appreciate, the terms "upper" and "lower" are sometimes used for ease of describing the figures, and indicate relative positions corresponding to the orientation of the figure on a properly oriented page, and may not reflect the proper orientation of the IMOD as implemented.

[0169] Certain features that are described in this specification in the context of separate implementations also can be implemented in combination in a single implementation. Conversely, various features that are described in the context of a single implementation also can be implemented in multiple implementations separately or in any suitable subcombination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a subcombination or variation of a subcombination.

[0170] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous. Moreover, the separation of various system components in the implementations described above should not be understood as requiring such separation in all implementations, and it should be understood that the described program components and systems can generally be integrated together in a single software product or packaged into multiple software products. Additionally, other implementations are within the scope of the following claims. In some cases, the actions recited in the claims can be performed in a different order and still achieve desirable results.

CLAIMS

What is claimed is:

1. A method of updating an electronic display, comprising:

storing a current frame in an electronic memory;

determining a display mode based on whether a current frame stream is static or dynamic;

selecting a dithering method based at least in part on the display mode;

dithering image data derived from the current frame using the dithering method to produce a dithered frame; and

updating the electronic display with displayed image data derived from the dithered frame.

2. The method of claim 1, wherein the current frame stream comprises two frames.

3. The method of claim 1, wherein the current frame stream comprises more than two frames.

4. The method of claim 1, further comprising updating the electronic display by line multiplying if the current frame stream is dynamic.

5. The method of claim 1, wherein if the current frame stream is dynamic the dithering method selected utilizes a noise signal that has a radially asymmetric frequency spectrum around zero frequency.

6. The method of claim 1, wherein if the current frame stream is static the dithering method selected utilizes either Floyd-Steinberg error-diffusion or a dither noise signal having a radially symmetric frequency spectrum around zero frequency.

7. The method of claim 1, wherein the lines are multiplied twice.

8. The method of claim 1, further comprising quantizing the dithered frame.

9. The method of claim 1, wherein the current frame has a color depth greater than the color depth of the displayed image data.

10. A display apparatus, comprising:

a display;

a display controller coupled to the display;

a processor, coupled to the display controller, wherein the processor is configured to

determine a display mode based on whether a current frame stream is static or dynamic,

select a dithering method based at least in part on the display mode,

dither image data using the dithering method to produce a dithered frame, and

control the display controller to display image data derived from the dithered frame.

11. The apparatus of claim 10, wherein the processor is further configured to line multiply image data if the current frame is dynamic.

12. The apparatus of claim 10, wherein the processor is further configured to quantize the dithered frame to produce a quantized frame.

13. The apparatus of claim 11, wherein the line multiplying is line doubling.

14. The apparatus of claim 10, further comprising a memory device that is configured to communicate with the processor.

15. The apparatus as recited in claim 10, further comprising a driver circuit configured to send at least one signal to the display.

16. The apparatus as recited in claim 10, further comprising a controller configured to send at least a portion of the image data to the driver circuit.

17. The apparatus as recited in claim 10, further comprising an image source module configured to send the image data to the processor.

18. The apparatus as recited in claim 17, wherein the image source module includes at least one of a receiver, transceiver, and transmitter.

19. The apparatus as recited in claim 10, further comprising an input device configured to receive input data and to communicate the input data to the processor.

20. A display method comprising dithering image data using a dithering method based at least in part on a frame update rate.

21. The method of claim 20, wherein the dithering method utilizes a noise signal that has a radially asymmetric frequency spectrum around zero frequency when the frame update rate is above a threshold.

22. The method of claim 20, wherein the dithering method utilizes either Floyd-Steinberg error-diffusion or a dither noise signal having a radially symmetric frequency spectrum around zero frequency when the frame update rate is below a threshold.

23. The method of claim 20, further comprising writing identical display data to multiple lines of a display simultaneously.

24. A display apparatus comprising:

means for storing a current frame in an electronic memory;

means for determining a display mode based on whether a current frame stream is static or dynamic;

means for selecting a dithering method based at least in part on the display mode;

means for dithering image data derived from the current frame using the dithering method to produce a dithered frame; and

means for updating an electronic display with image data derived from the dithered frame.

25. The display apparatus of claim 24, further comprising means for quantizing the dithered frame to produce a quantized frame.

26. A non-transitory, computer readable storage medium having instructions stored thereon that cause a processing circuit to perform:

storing a current frame in an electronic memory;

determining a display mode based on whether a current frame stream is static or dynamic;

selecting a dithering method based at least in part on the display mode;

dithering image data derived from the current frame using the dithering method to produce a dithered frame; and

updating an electronic display with image data derived from the dithered frame.

27. The computer readable storage medium of 25 wherein the current frame stream comprises one frame.

28. The computer readable storage medium of claim 25, wherein the current frame stream comprises multiple frames.

29. The computer readable storage medium of claim 25, further comprising instructions causing a processing circuit to quantize the dithered frame data to produce a quantized frame.

30. The computer readable storage medium of claim 25, wherein if the current frame stream is dynamic the dithering method selected utilizes a noise signal that has a radially asymmetric frequency spectrum around zero frequency.

31. The computer readable storage medium of claim 25, wherein if the current frame stream is static the dithering method selected utilizes either Floyd-Steinberg error-diffusion or a noise signal having a radially symmetric frequency spectrum around zero frequency.

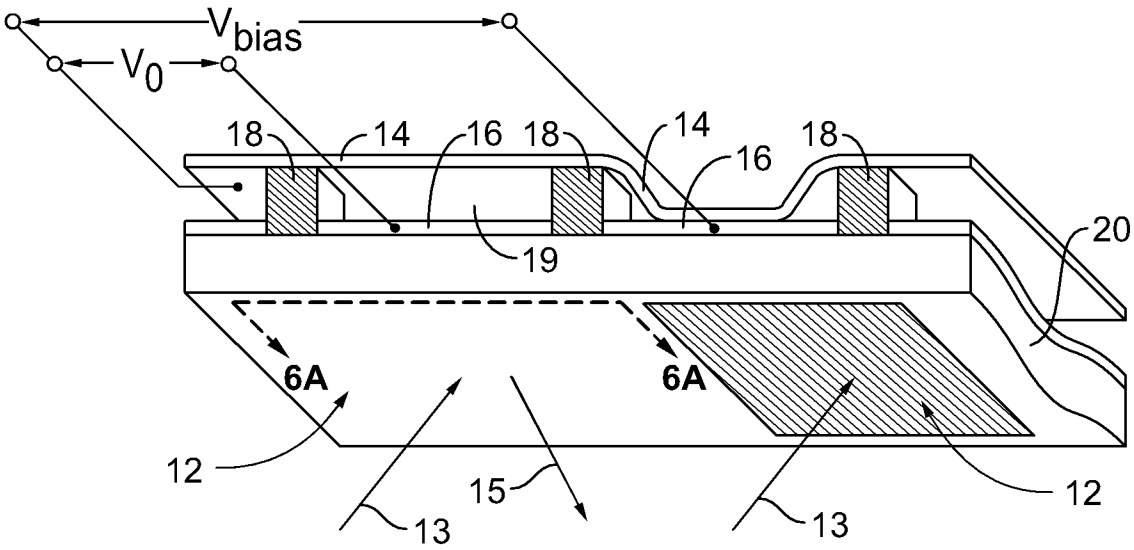


Figure 1

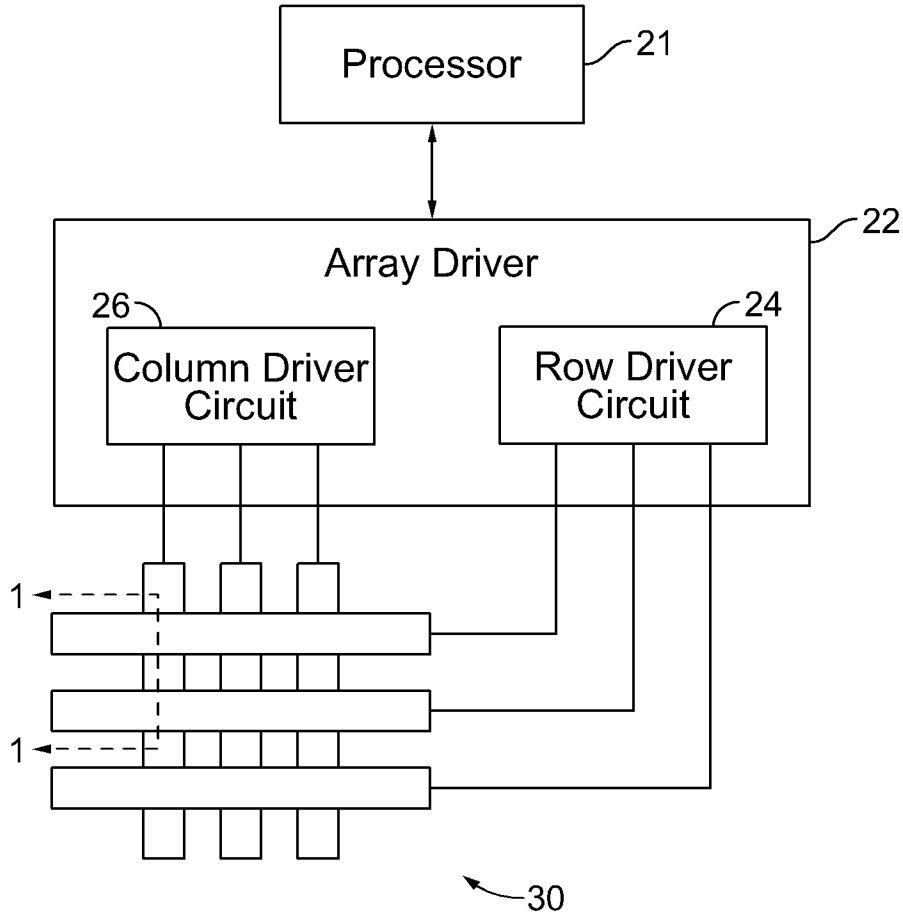


Figure 2

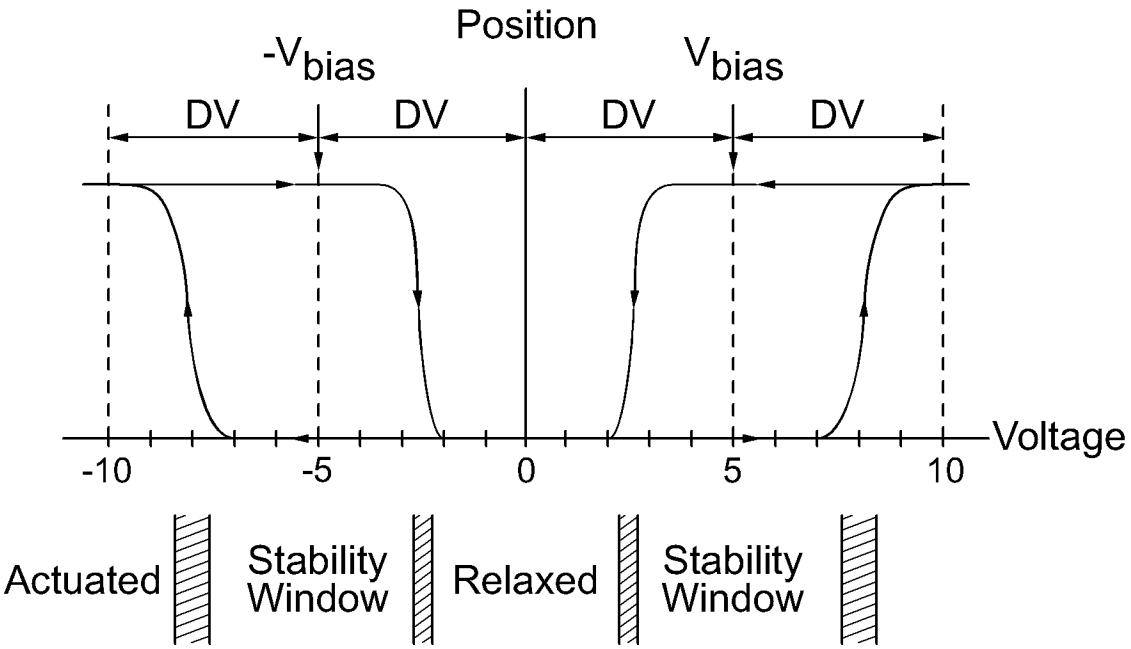


Figure 3

Common Voltages

Segment Voltages	VC_{ADD_H}	VC_{HOLD_H}	VC_{REL}	VC_{HOLD_L}	VC_{ADD_L}
VS_H	Stable	Stable	Relax	Stable	Actuate
VS_L	Actuate	Stable	Relax	Stable	Stable

Figure 4

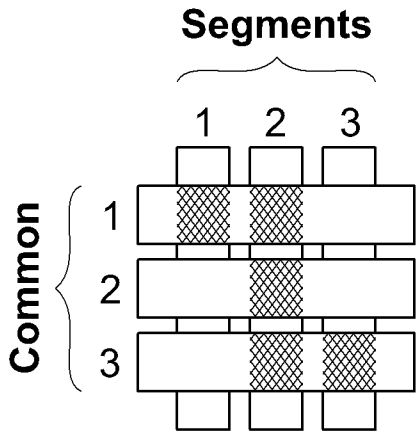


Figure 5A

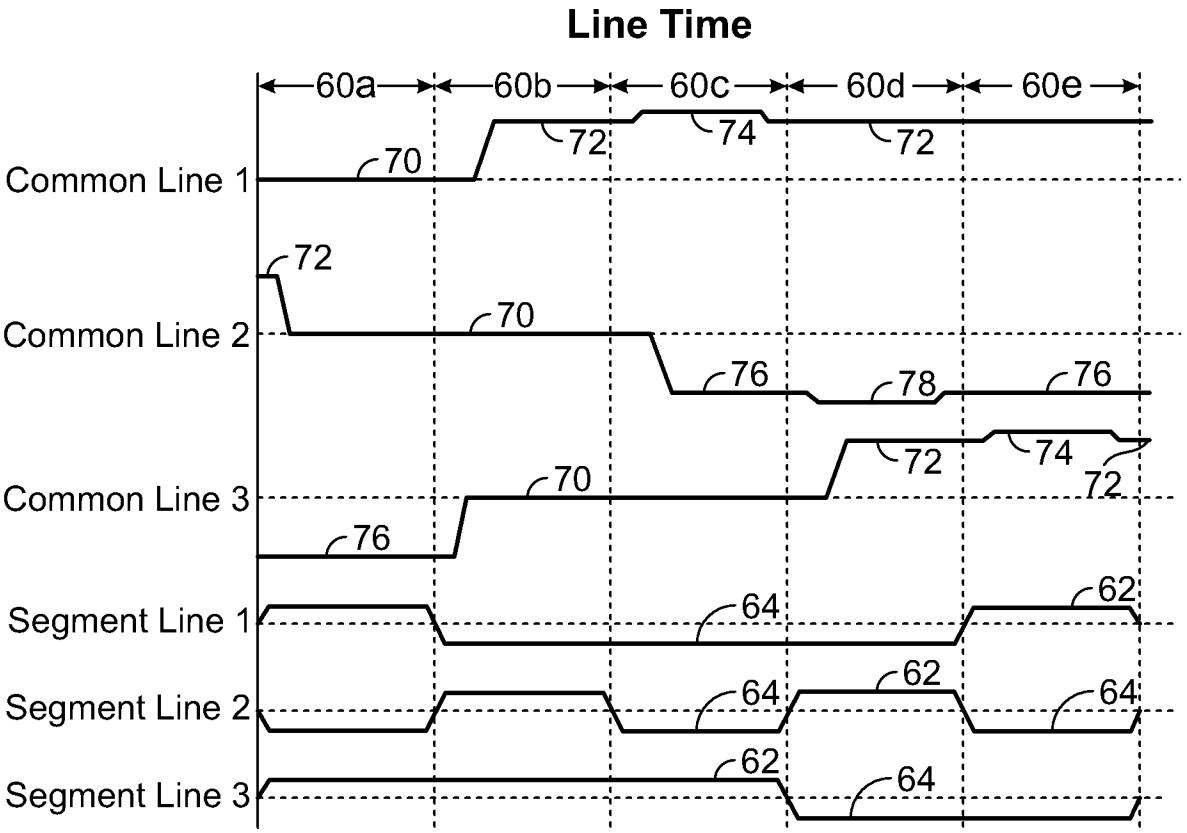


Figure 5B

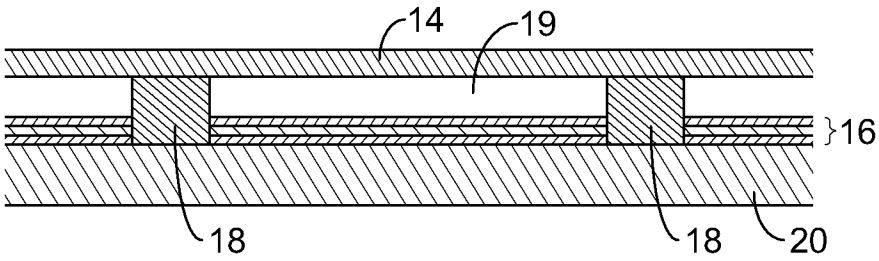


Figure 6A

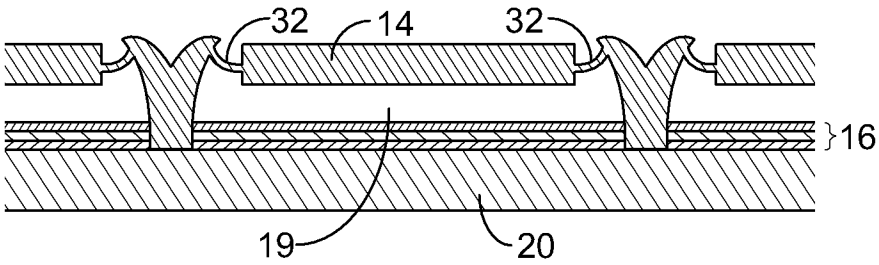


Figure 6B

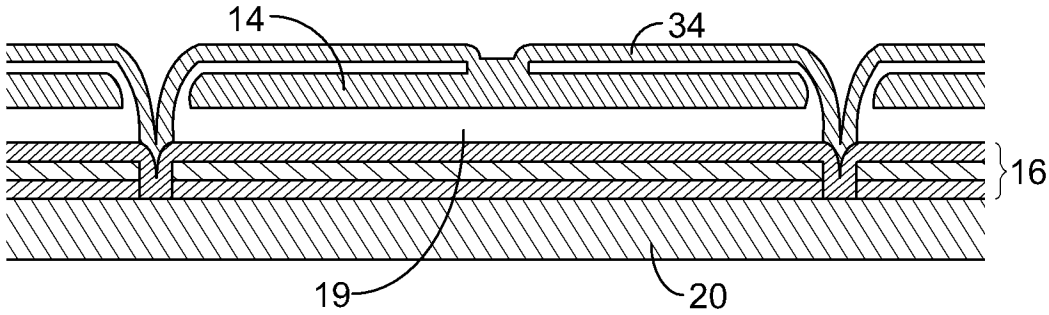


Figure 6C

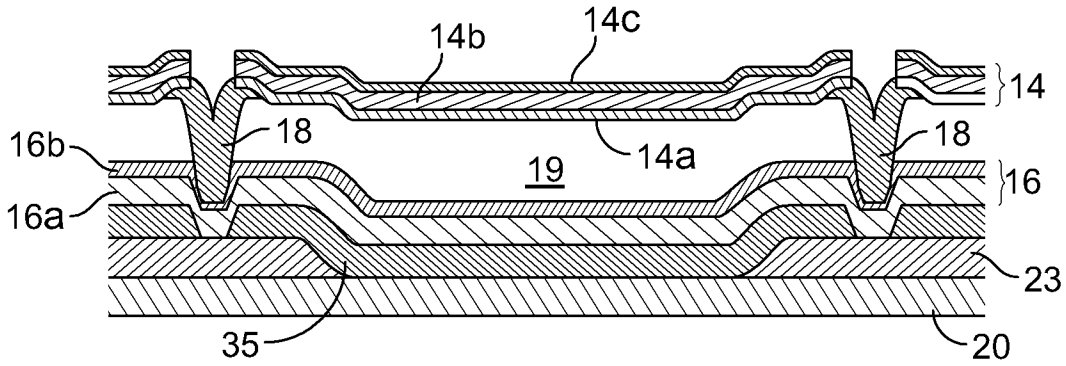


Figure 6D

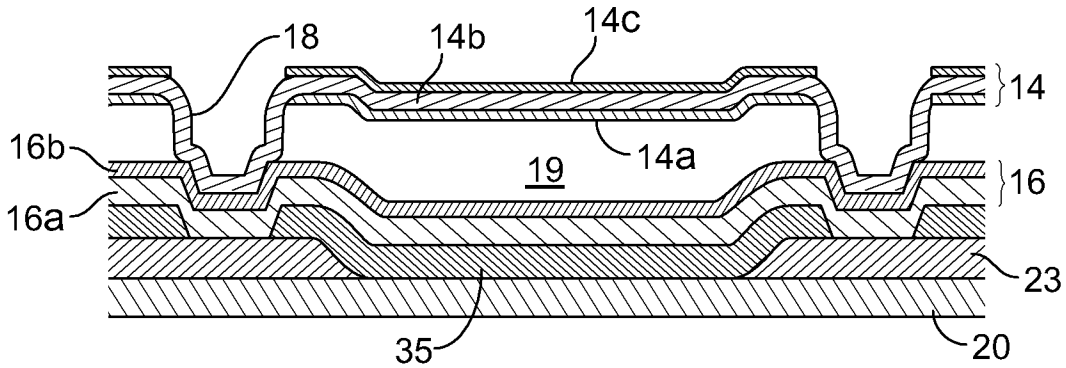


Figure 6E

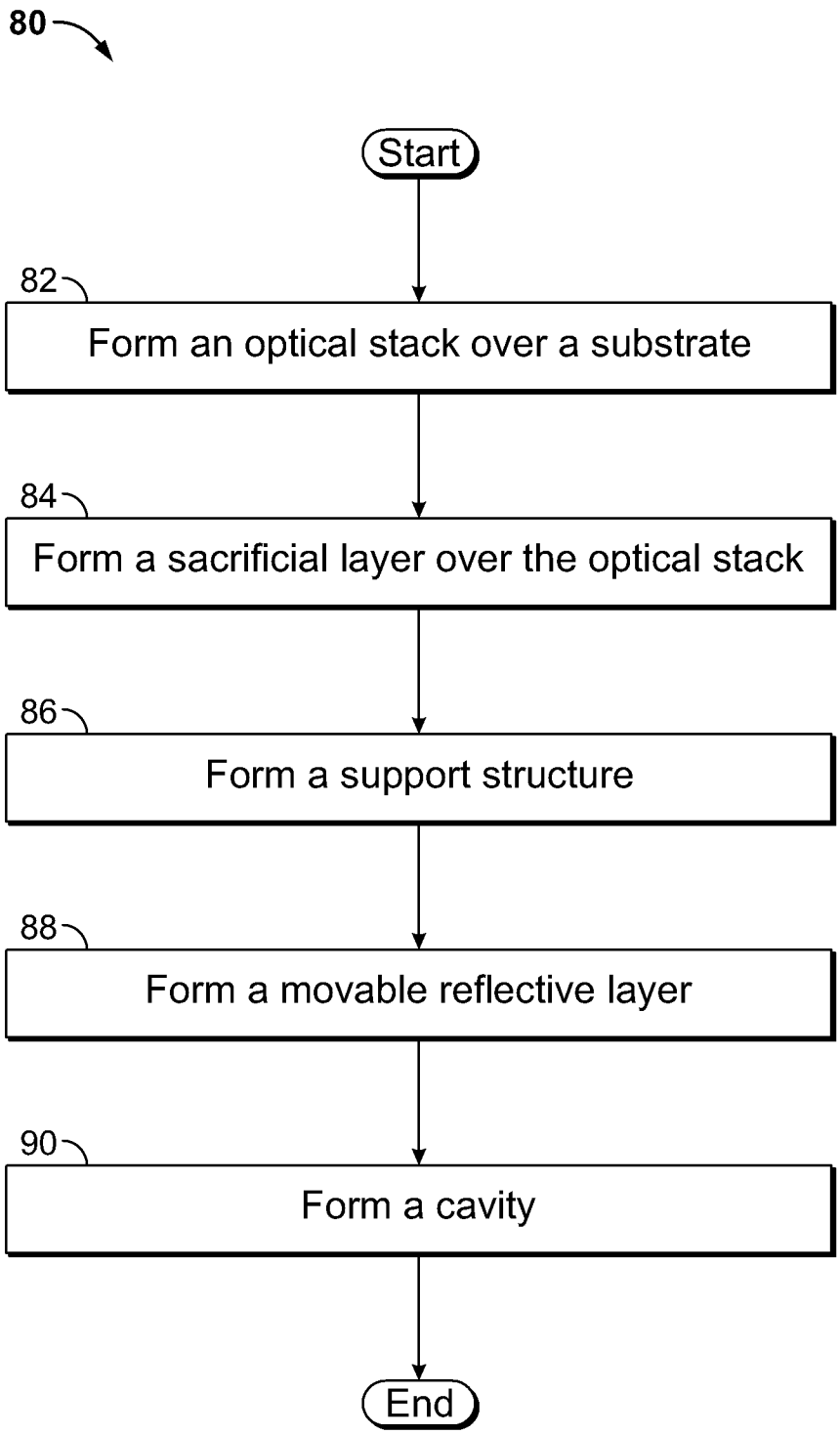

80
Start
82
Form an optical stack over a substrate
84
Form a sacrificial layer over the optical stack
86
Form a support structure
88
Form a movable reflective layer
90
Form a cavity
End

Figure 7

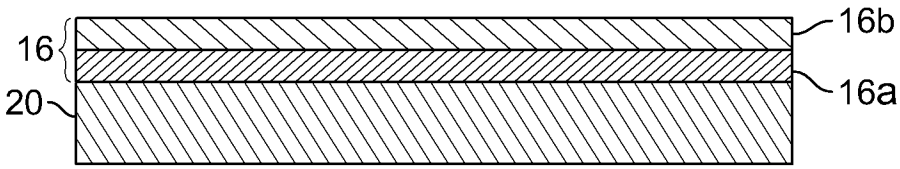


Figure 8A

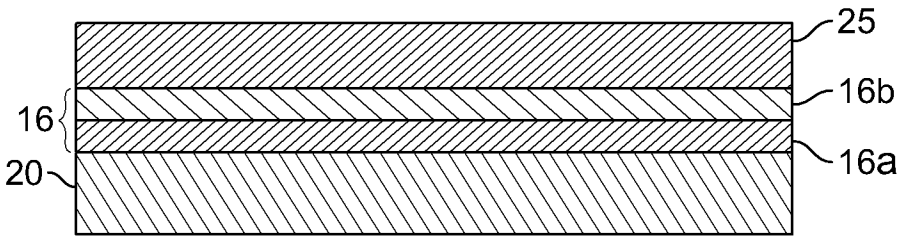


Figure 8B

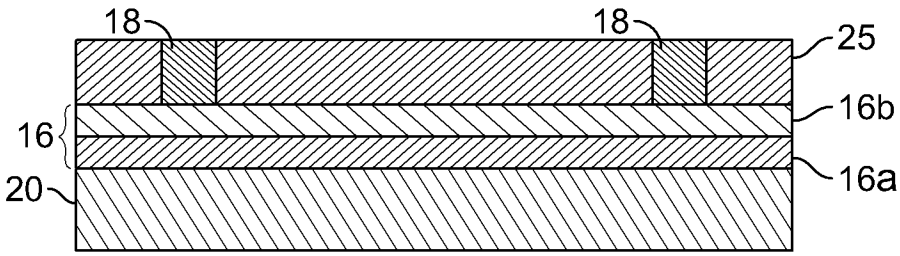


Figure 8C

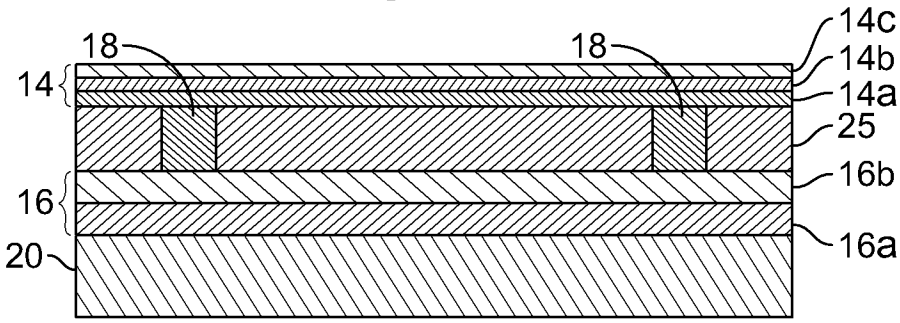


Figure 8D

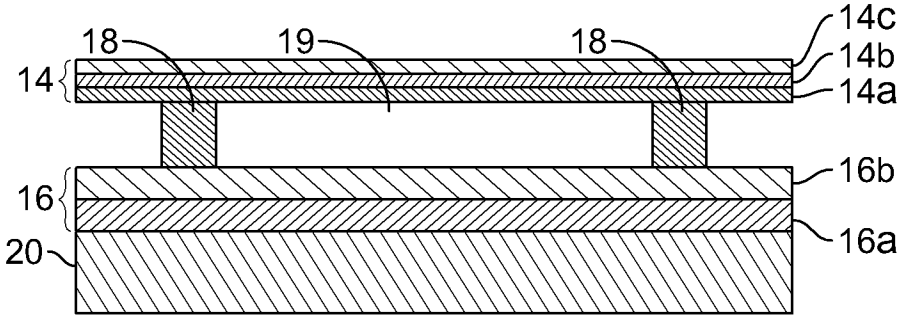


Figure 8E

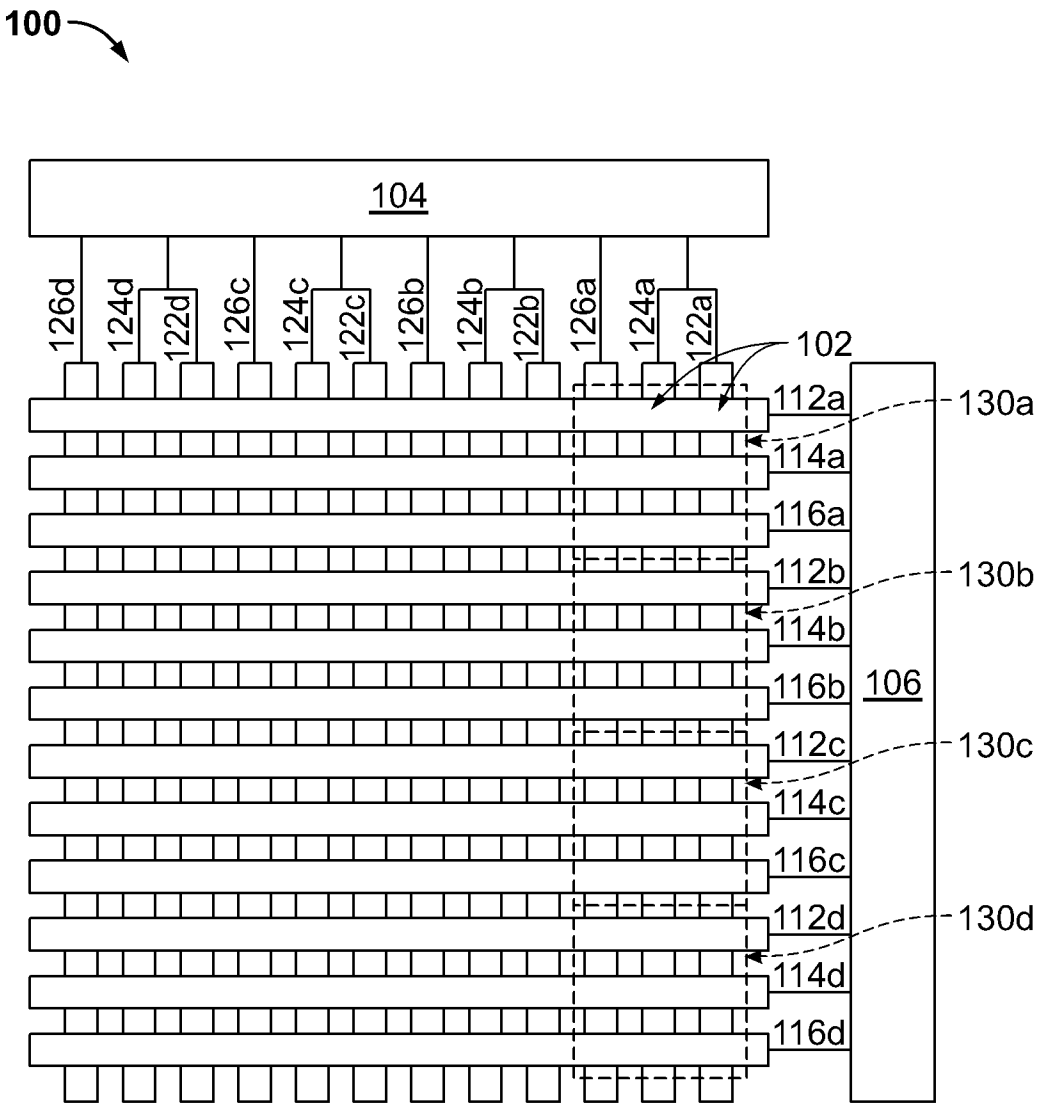

100
104
126d
124d
122d
126c
124c
122c
126b
124b
122b
126a
124a
122a
102
112a
114a
116a
112b
114b
116b
112c
114c
116c
112d
114d
116d
106
130a
130b
130c
130d

Figure 9

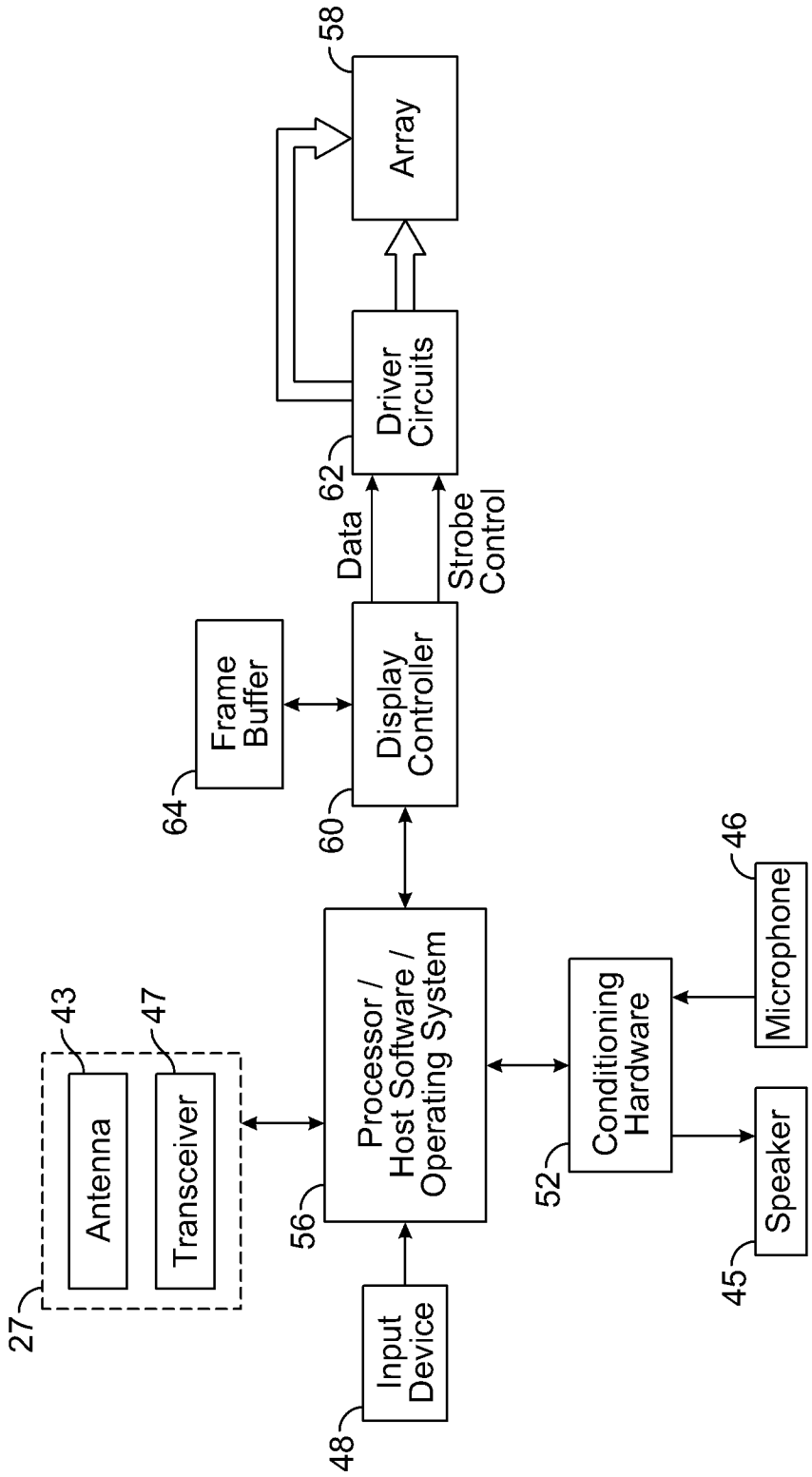

40
58
Array
Driver Circuits
62
Data
Strobe Control
Frame Buffer
64
Display Controller
60
Processor / Host Software / Operating System
56
Antenna
43
Transceiver
47
27
Input Device
48
Conditioning Hardware
52
Microphone
46
Speaker
45

Figure 10

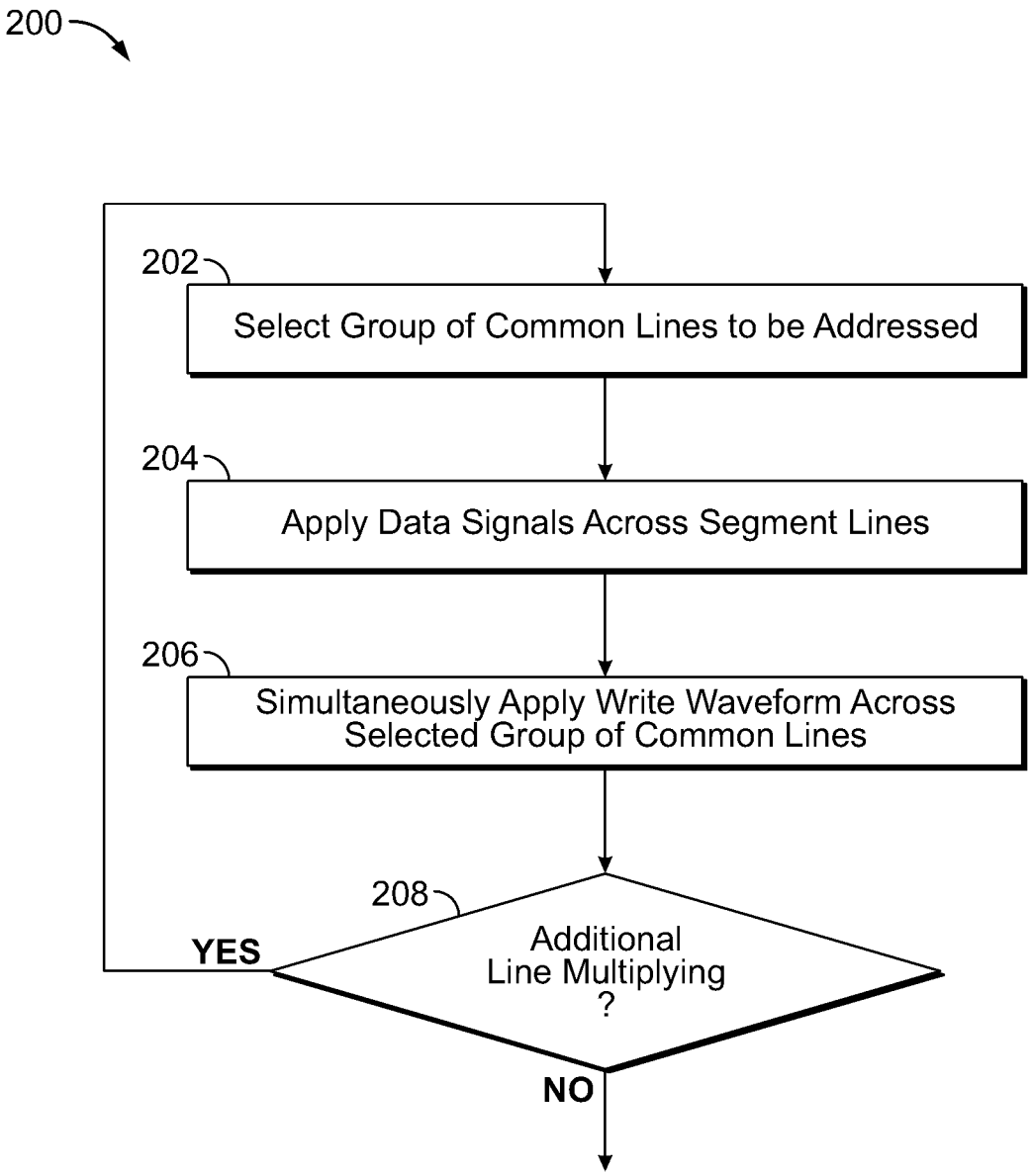

200
202
Select Group of Common Lines to be Addressed
204
Apply Data Signals Across Segment Lines
206
Simultaneously Apply Write Waveform Across Selected Group of Common Lines
208
Additional Line Multiplying ?
YES
NO

Figure 11

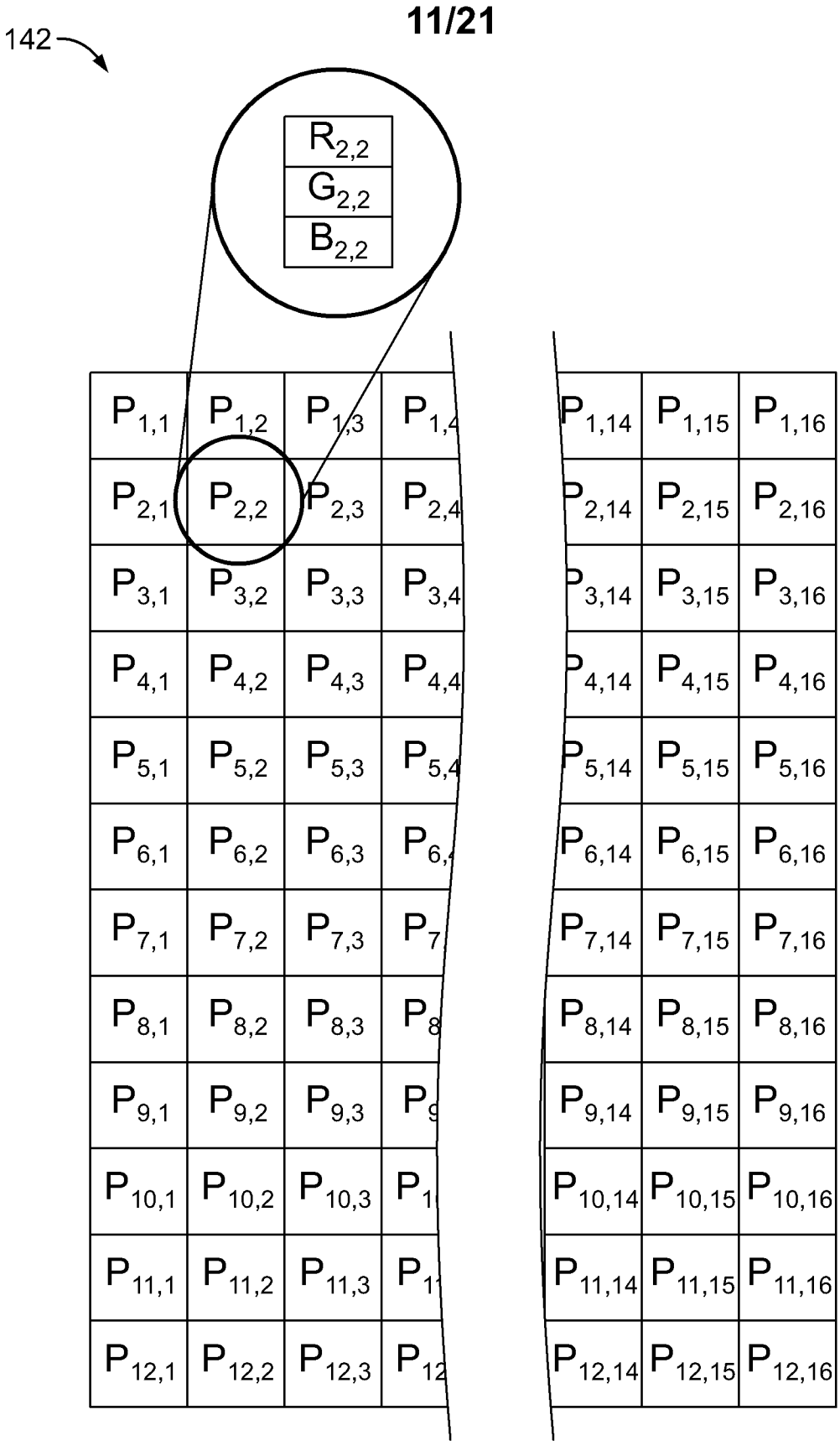

142
R2,2
G2,2
B2,2
P1,1 P1,2 P1,3 P1,4 P1,14 P1,15 P1,16
P2,1 P2,2 P2,3 P2,4 P2,14 P2,15 P2,16
P3,1 P3,2 P3,3 P3,4 P3,14 P3,15 P3,16
P4,1 P4,2 P4,3 P4,4 P4,14 P4,15 P4,16
P5,1 P5,2 P5,3 P5,4 P5,14 P5,15 P5,16
P6,1 P6,2 P6,3 P6,4 P6,14 P6,15 P6,16
P7,1 P7,2 P7,3 P7 P7,14 P7,15 P7,16
P8,1 P8,2 P8,3 P8 P8,14 P8,15 P8,16
P9,1 P9,2 P9,3 P9 P9,14 P9,15 P9,16
P10,1 P10,2 P10,3 P1 P10,14 P10,15 P10,16
P11,1 P11,2 P11,3 P11 P11,14 P11,15 P11,16
P12,1 P12,2 P12,3 P12 P12,14 P12,15 P12,16

Figure 12

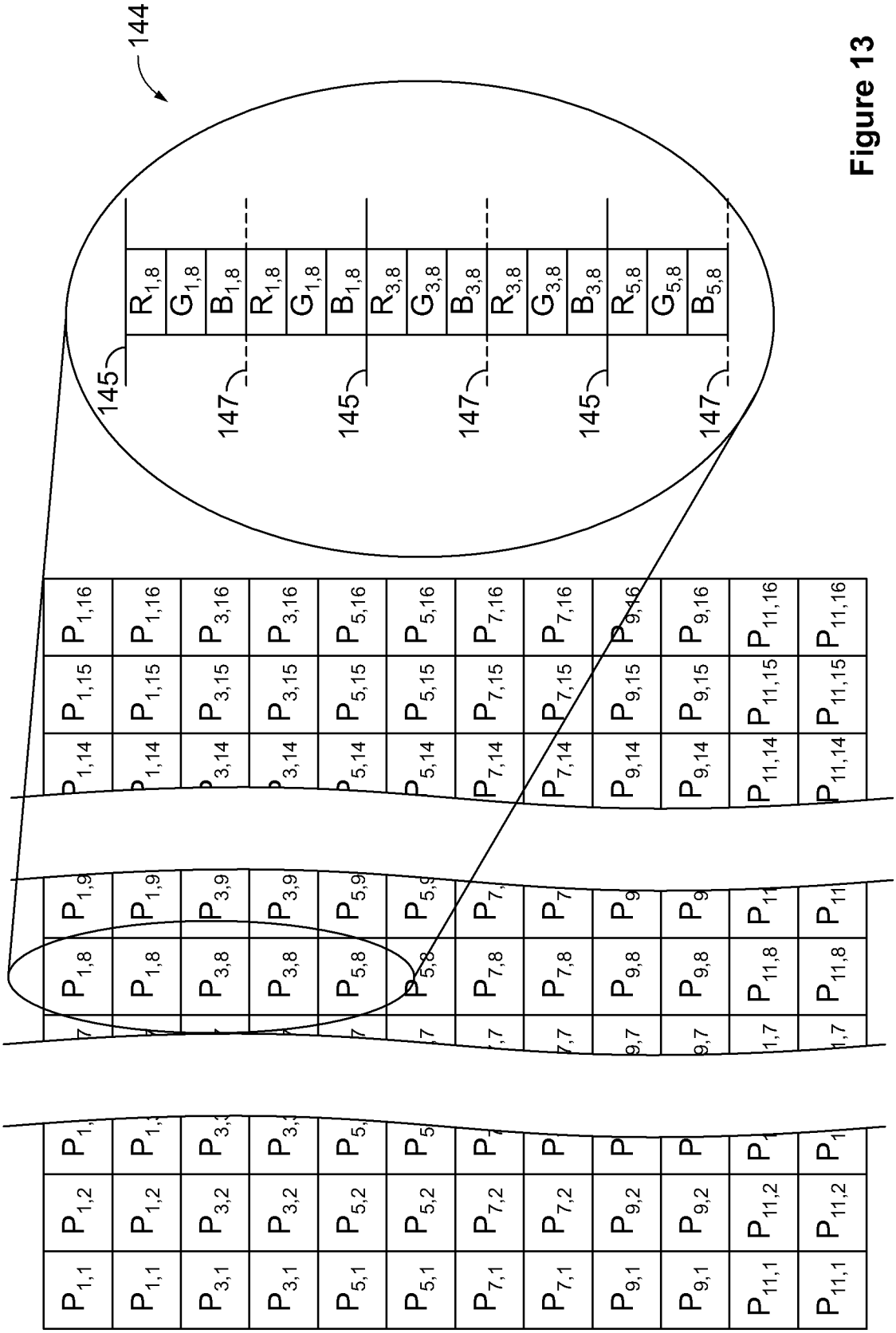

144
145
147
145
147
145
147
$R_{1,8}$ $G_{1,8}$ $B_{1,8}$ $R_{1,8}$ $G_{1,8}$ $B_{1,8}$ $R_{3,8}$ $G_{3,8}$ $B_{3,8}$ $R_{3,8}$ $G_{3,8}$ $B_{3,8}$ $R_{5,8}$ $G_{5,8}$ $B_{5,8}$
$P_{1,16}$ $P_{3,16}$ $P_{5,16}$ $P_{7,16}$ $P_{9,16}$ $P_{11,16}$
$P_{1,15}$ $P_{3,15}$ $P_{5,15}$ $P_{7,15}$ $P_{9,15}$ $P_{11,15}$
$P_{1,14}$ $P_{3,14}$ $P_{5,14}$ $P_{7,14}$ $P_{9,14}$ $P_{11,14}$
$P_{1,8}$ $P_{3,8}$ $P_{5,8}$ $P_{7,8}$ $P_{9,8}$ $P_{11,8}$
$P_{1,2}$ $P_{3,2}$ $P_{5,2}$ $P_{7,2}$ $P_{9,2}$ $P_{11,2}$
$P_{1,1}$ $P_{3,1}$ $P_{5,1}$ $P_{7,1}$ $P_{9,1}$ $P_{11,1}$

Figure 13

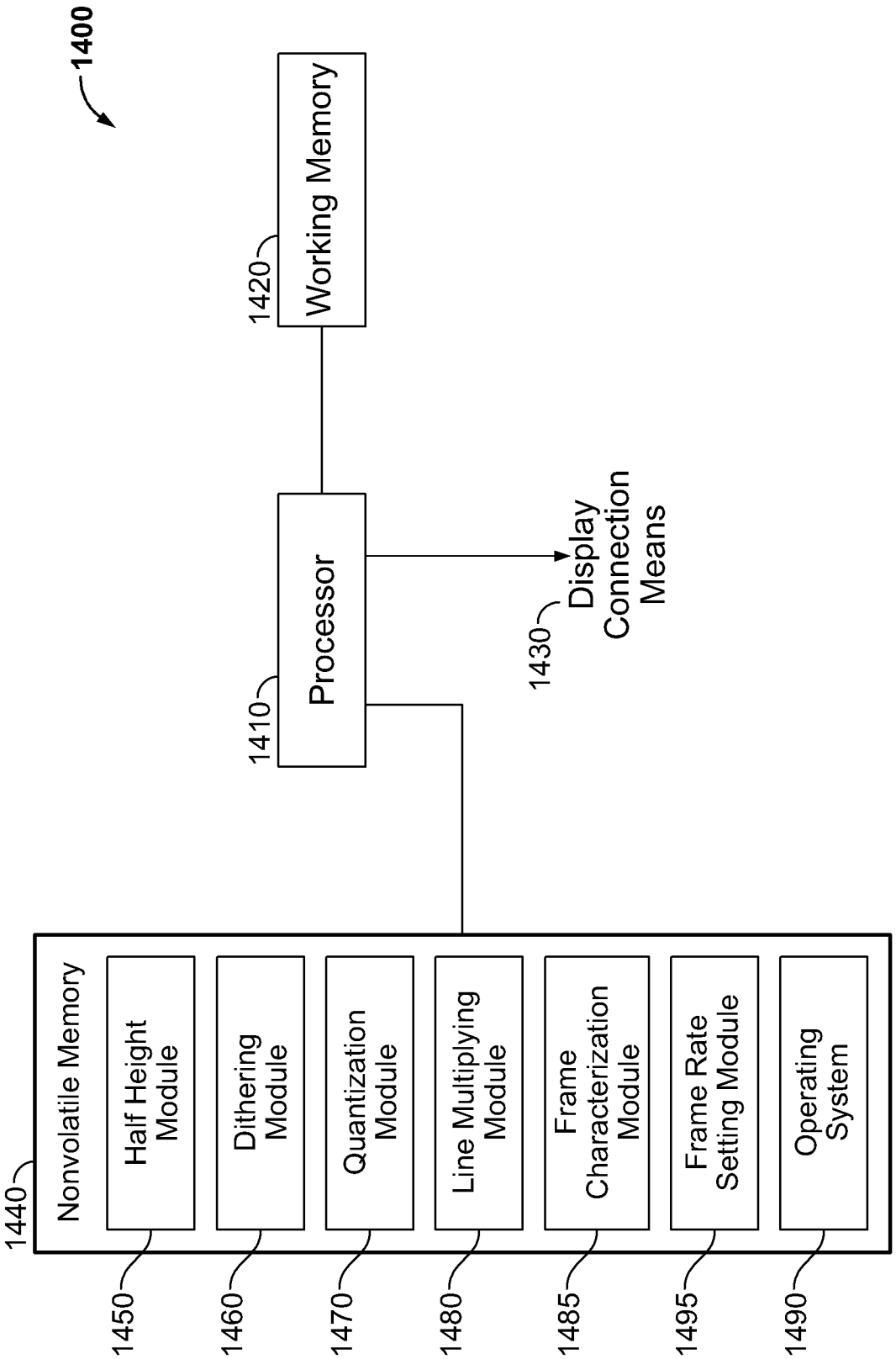

1400
1420
Working Memory
1410
Processor
1430
Display Connection Means
1440
Nonvolatile Memory
Half Height Module
Dithering Module
Quantization Module
Line Multiplying Module
Frame Characterization Module
Frame Rate Setting Module
Operating System
1450
1460
1470
1480
1485
1495
1490

Figure 14

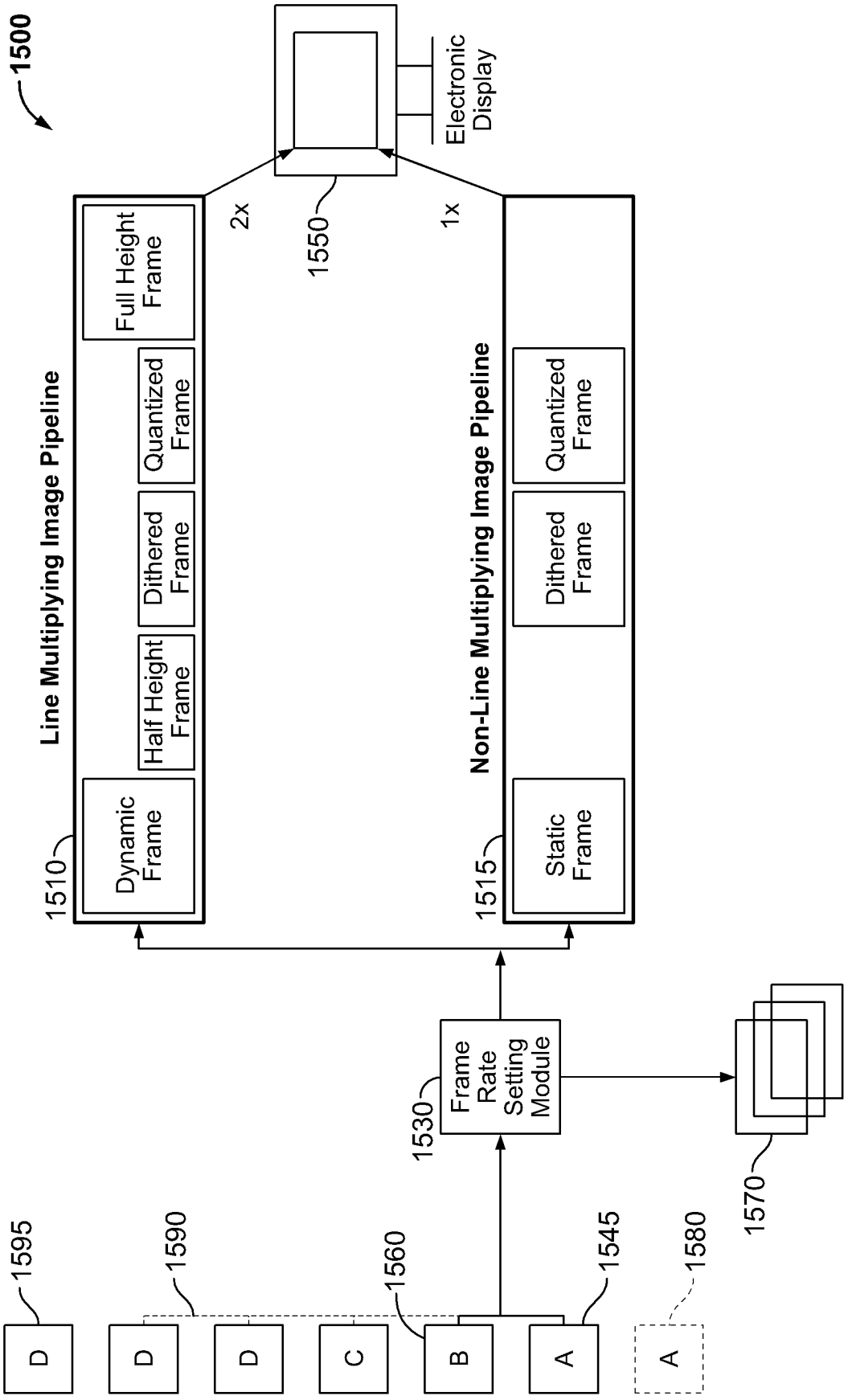

1500
Electronic Display
2x
1550
1x
Line Multiplying Image Pipeline
Full Height Frame
Quantized Frame
Dithered Frame
Half Height Frame
Dynamic Frame
1510
Non-Line Multiplying Image Pipeline
Quantized Frame
Dithered Frame
Static Frame
1515
Frame Rate Setting Module
1530
1570
D
D
D
C
B
A
A
1595
1590
1560
1545
1580

Figure 15

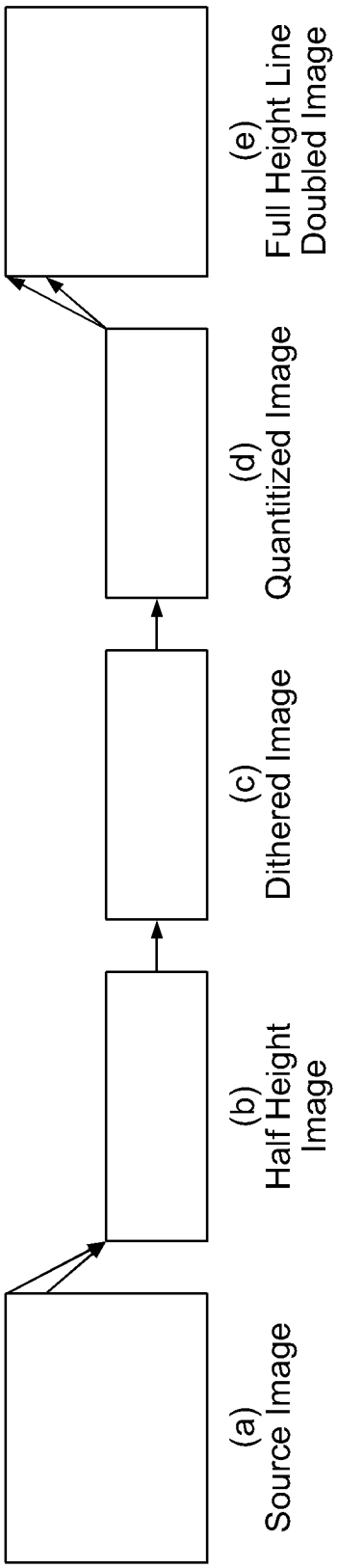

(a)
Source Image
(b)
Half Height
Image
(c)
Dithered Image
(d)
Quantitized Image
(e)
Full Height Line
Doubled Image

Figure 16

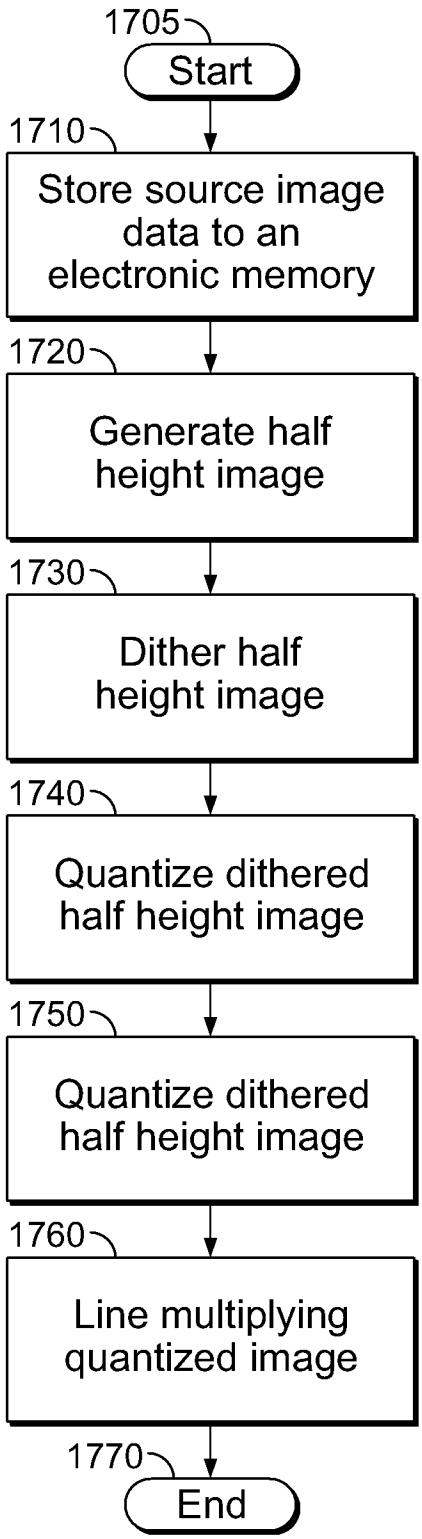

1705
Start
1710
Store source image data to an electronic memory
1720
Generate half height image
1730
Dither half height image
1740
Quantize dithered half height image
1750
Quantize dithered half height image
1760
Line multiplying quantized image
1770
End

Figure 17

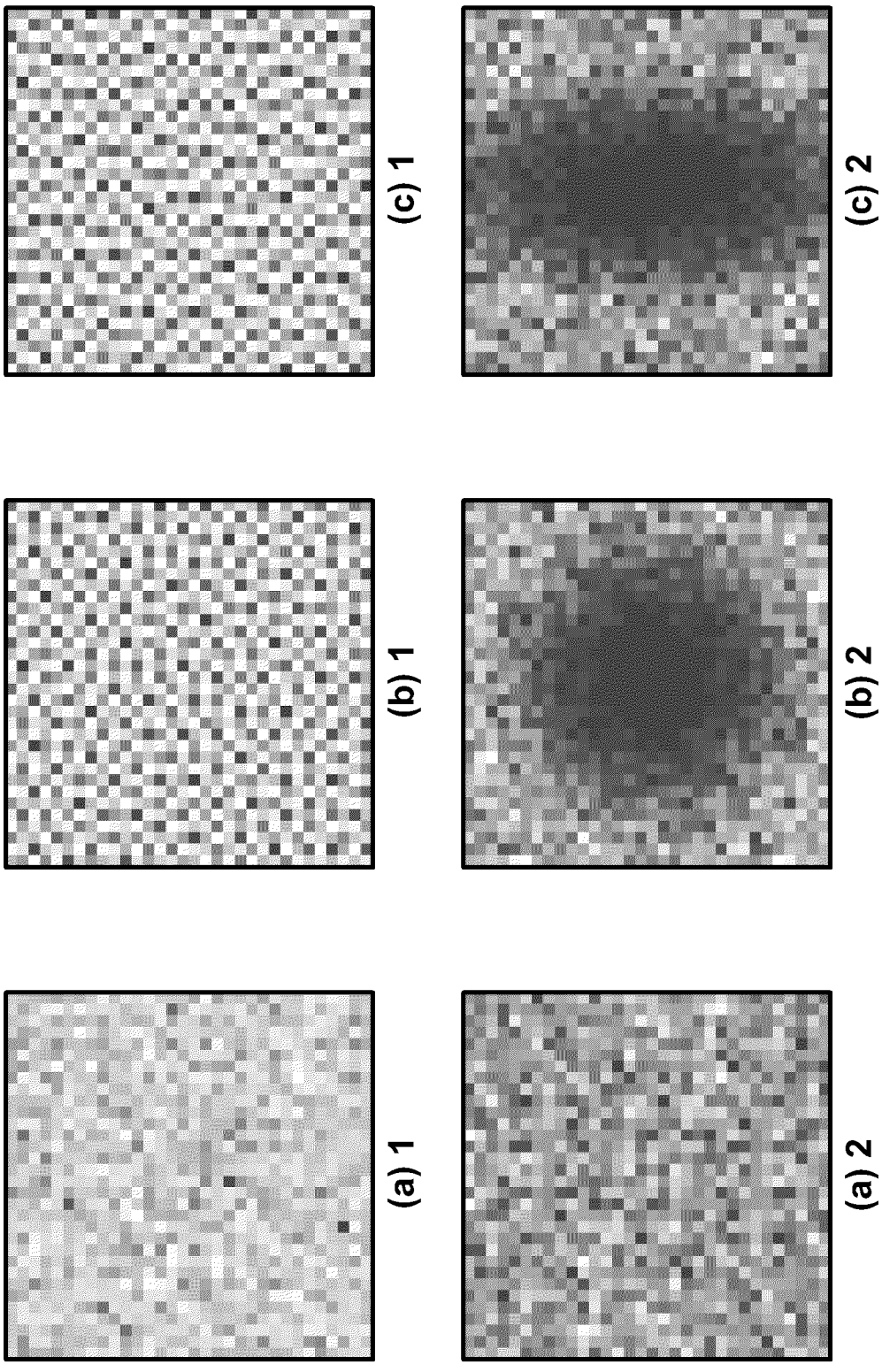


Figure 18

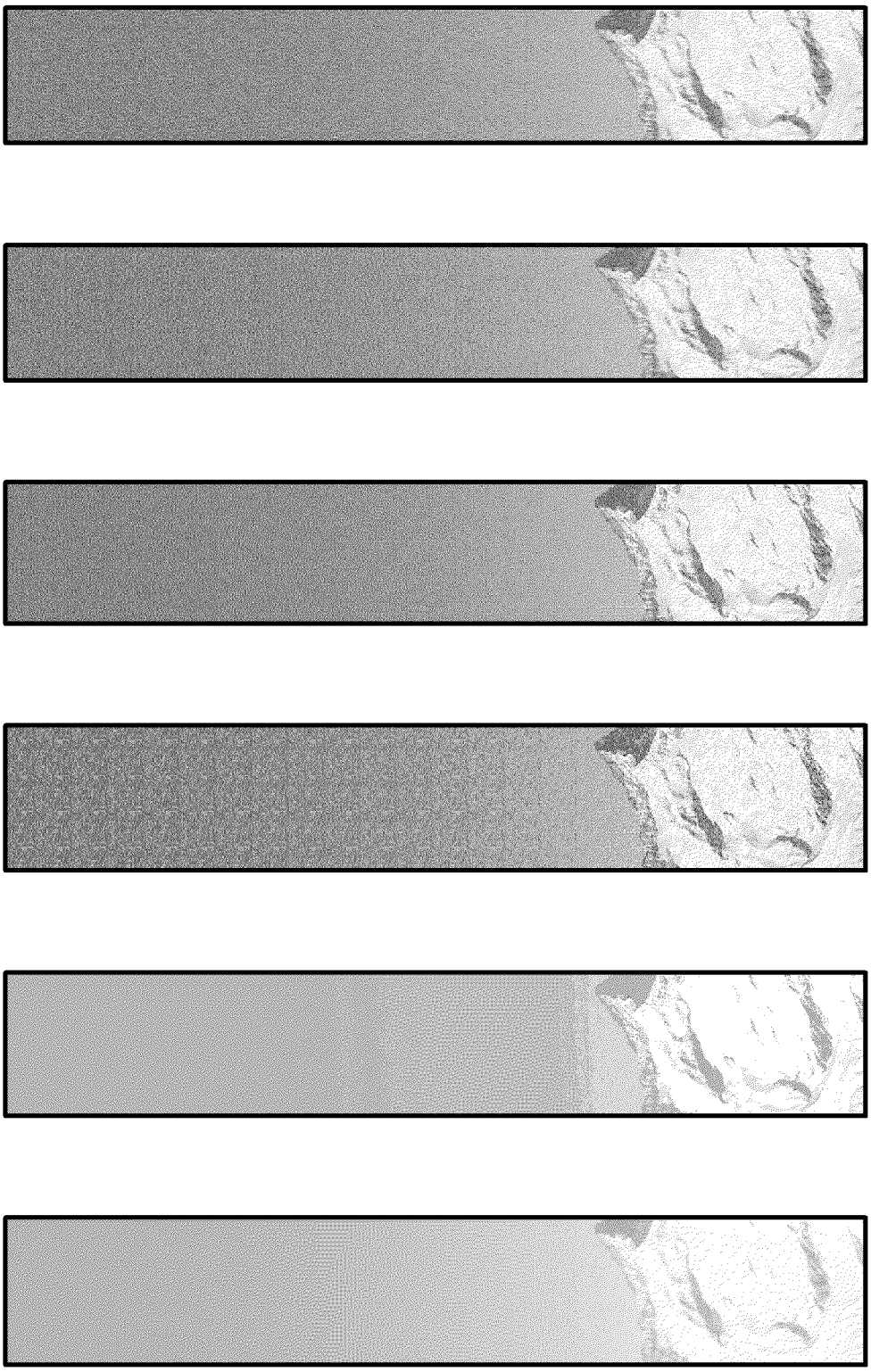


Figure 19

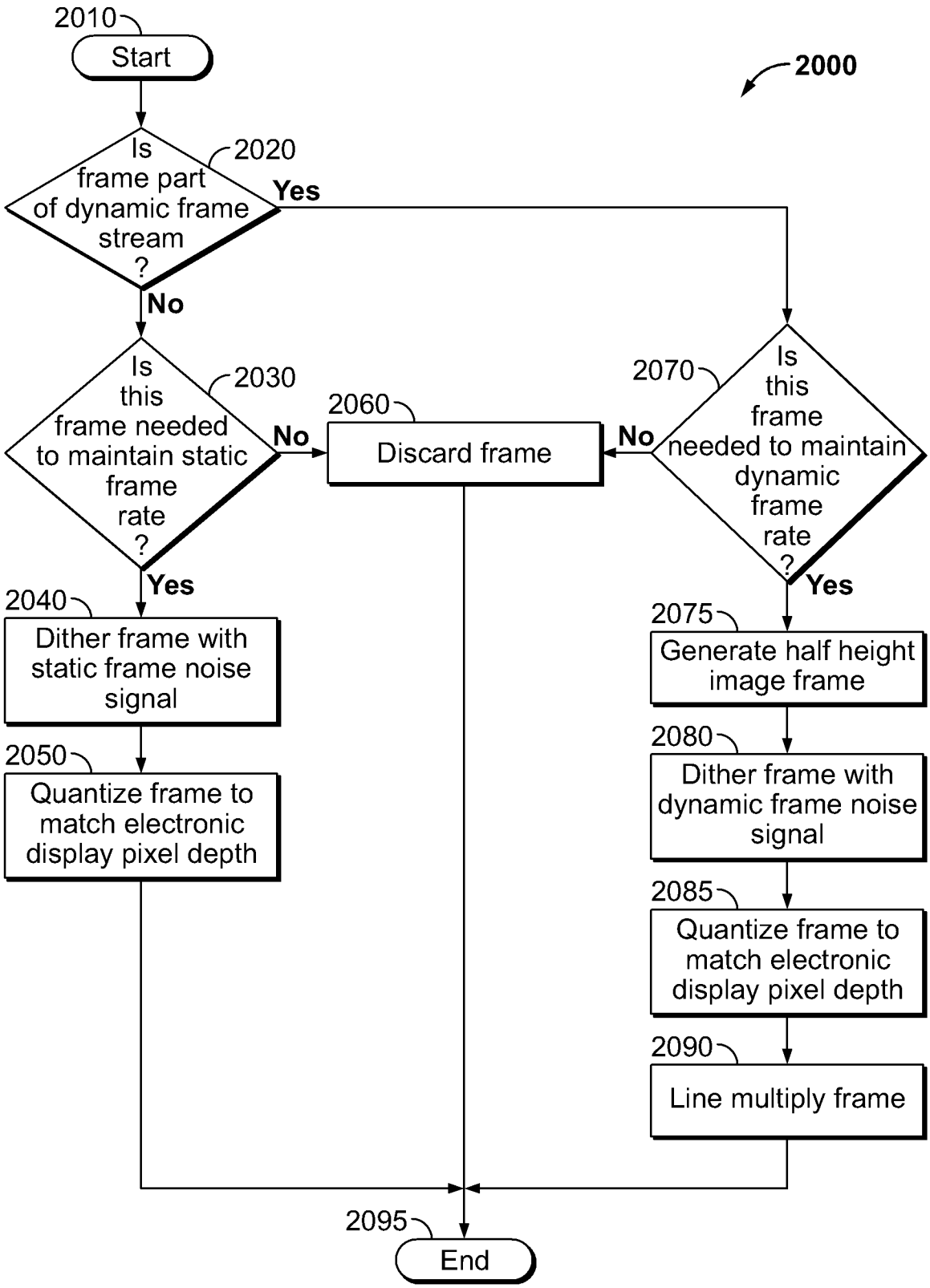

2010
Start
2000
Is frame part of dynamic frame stream ?
2020
Yes
No
Is this frame needed to maintain static frame rate ?
2030
No
2060
Discard frame
No
2070
Is this frame needed to maintain dynamic frame rate ?
Yes
Yes
2040
Dither frame with static frame noise signal
2050
Quantize frame to match electronic display pixel depth
2075
Generate half height image frame
2080
Dither frame with dynamic frame noise signal
2085
Quantize frame to match electronic display pixel depth
2090
Line multiply frame
2095
End

Figure 20

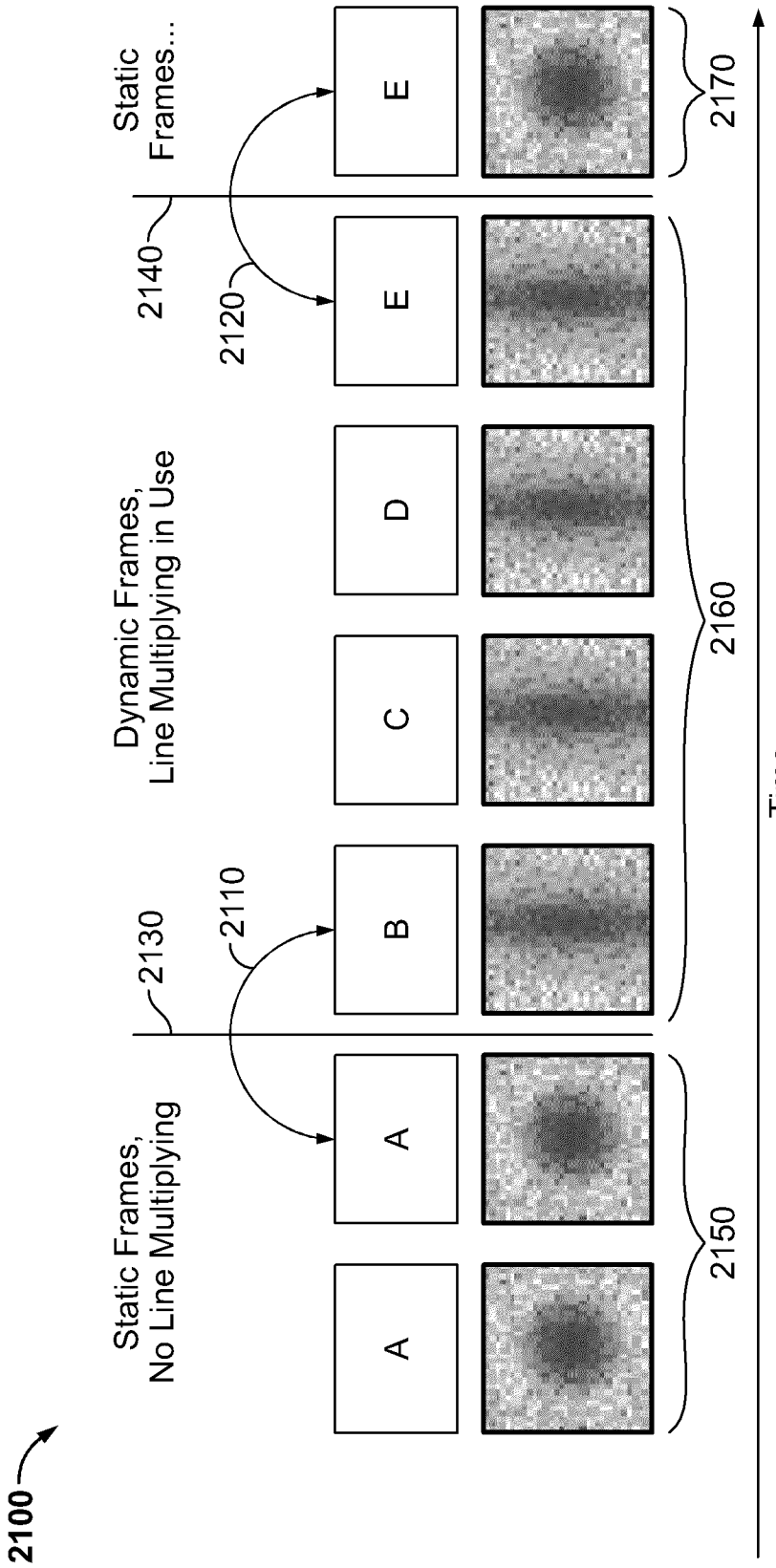

2100
Static Frames, No Line Multiplying
Dynamic Frames, Line Multiplying in Use
Static Frames...
A
A
B
C
D
E
E
2110
2120
2130
2140
2150
2160
2170
Time

Figure 21

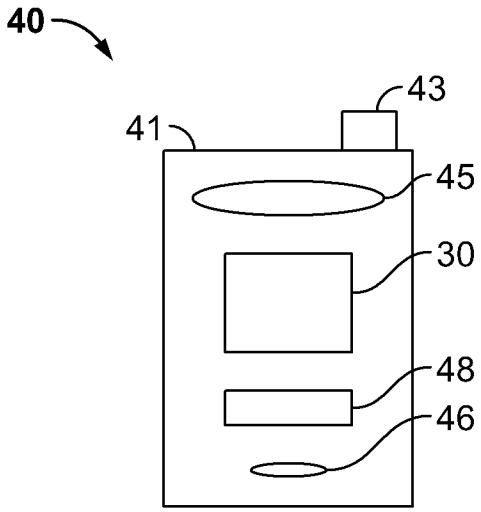


Figure 22A

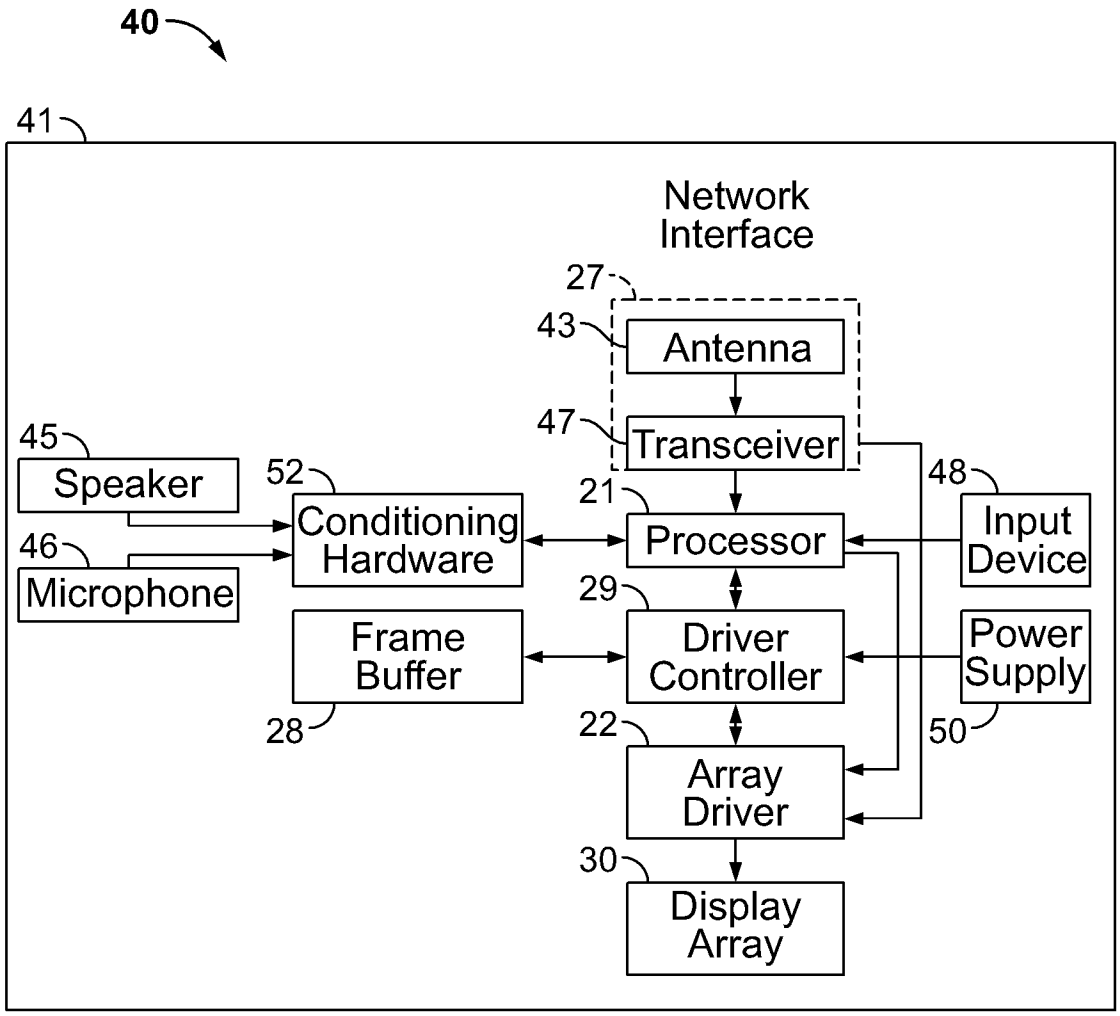


Figure 22B