



US 20120249474A1

(19) **United States**

(12) **Patent Application Publication**
PRATT et al.

(10) **Pub. No.: US 2012/0249474 A1**

(43) **Pub. Date: Oct. 4, 2012**

(54) **PROXIMITY AND FORCE DETECTION FOR
HAPTIC EFFECT GENERATION**

Publication Classification

(75) Inventors: **Susan Michelle PRATT**,
Caherconlish (IE); **Mark J.**
MURPHY, Kilmore (IE); **Eoin E.**
ENGLISH, Dromkeen (IE);
Adrian FLANAGAN, Raheen (IE)

(51) **Int. Cl.**
G06F 3/044 (2006.01)

(52) **U.S. Cl.** **345/174**

(73) Assignee: **ANALOG DEVICES, INC.**,
Norwood, MA (US)

(57) **ABSTRACT**

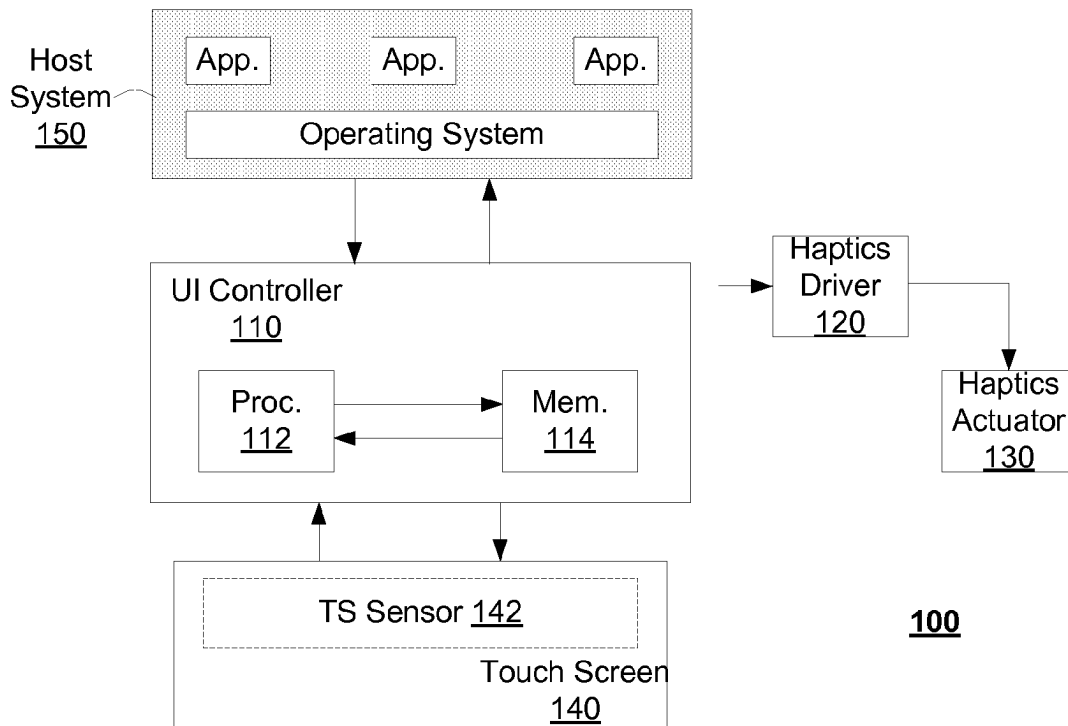
(21) Appl. No.: **13/434,623**

(22) Filed: **Mar. 29, 2012**

The present invention may provide a device including a haptic driver to drive a coupled actuator causing the actuator to generate a vibratory haptic effect. A touch screen may display a user interface and may include a sensor to detect user interaction with the touch screen within a predetermined range above the touch screen. A controller may calculate a proximity event based on the detected user interaction above the touch screen, and to control haptic driver operations according to the proximity event.

Related U.S. Application Data

(60) Provisional application No. 61/470,764, filed on Apr. 1, 2011.



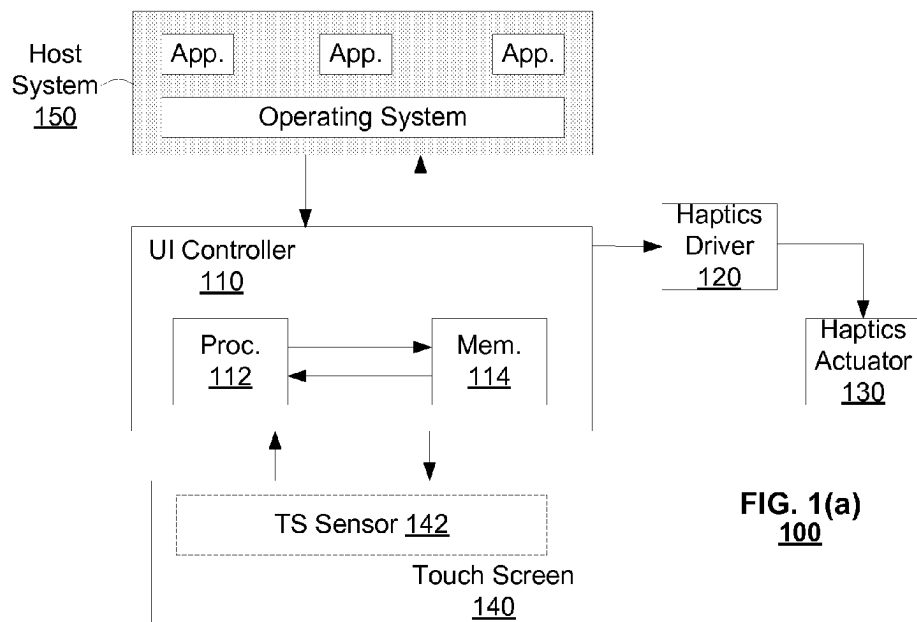


FIG. 1(a)
100

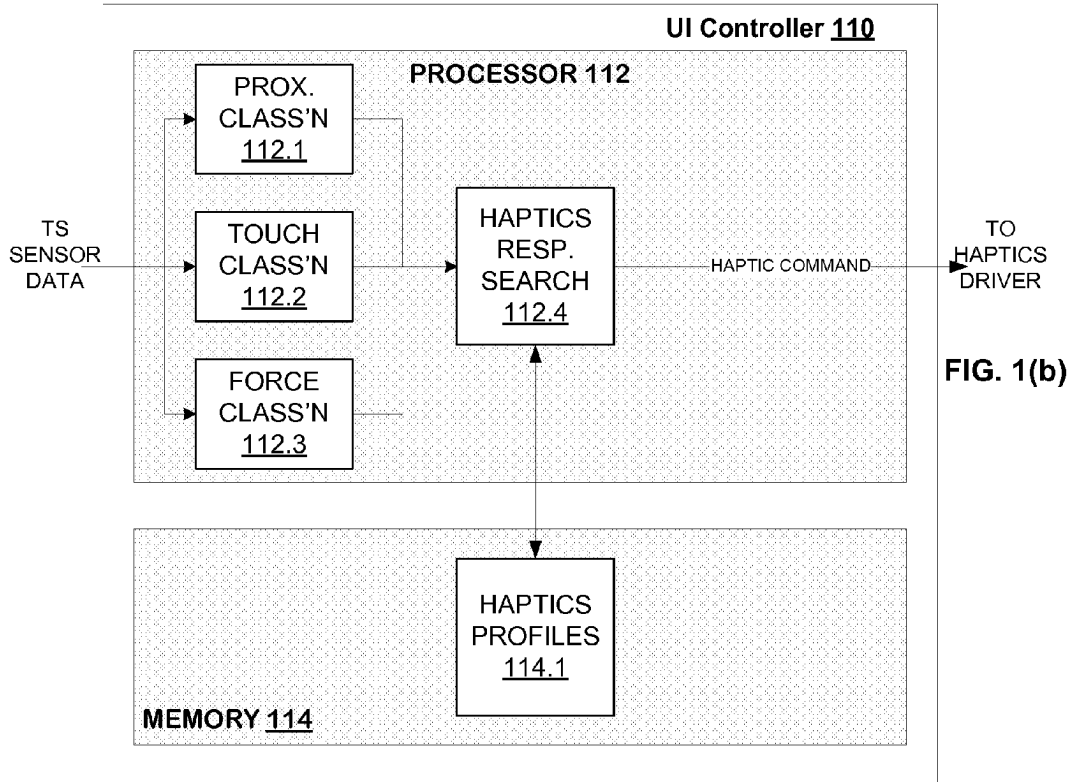


FIG. 1(b)

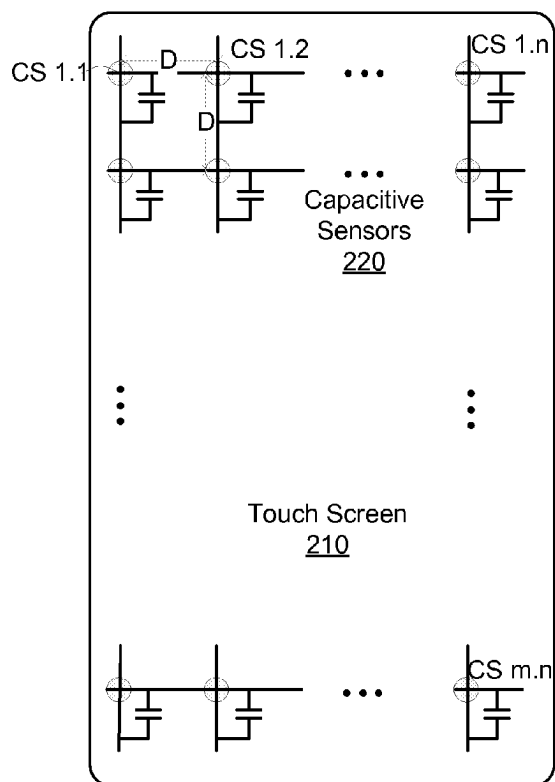


FIG. 2(a)

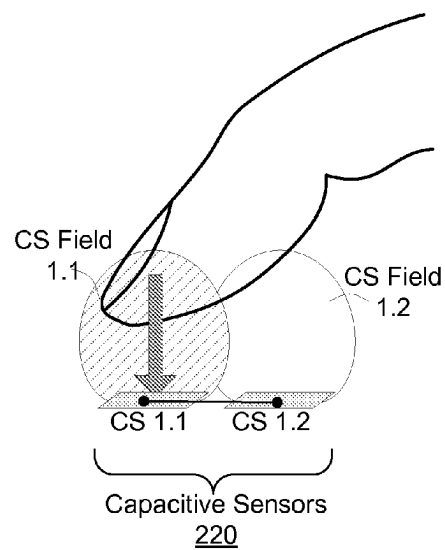


FIG. 2(c)

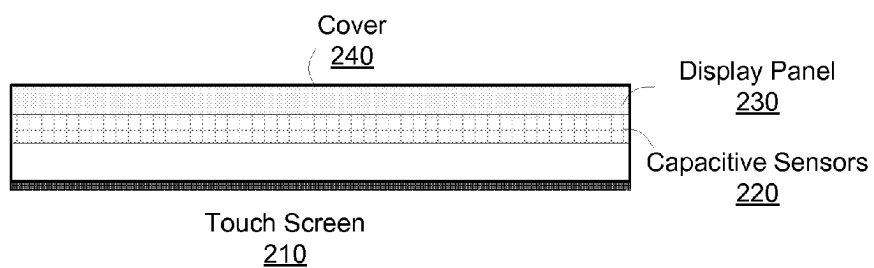


FIG. 2(b)

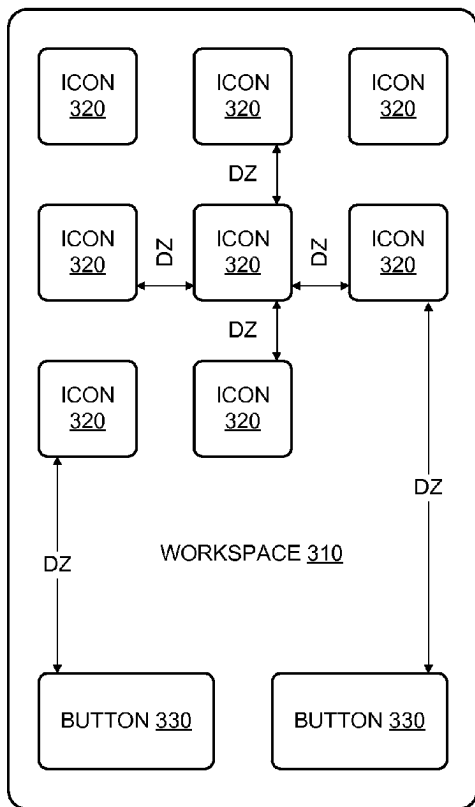


FIG. 3(a)

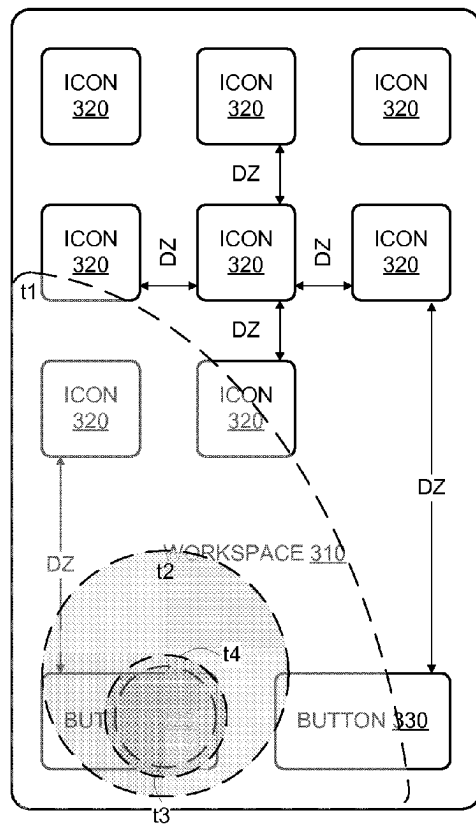
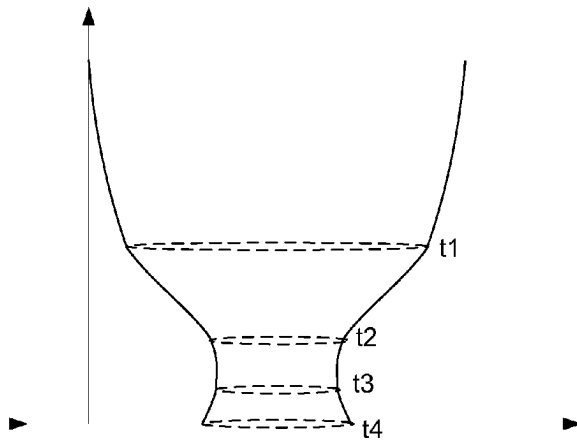


FIG. 3(b)

▲



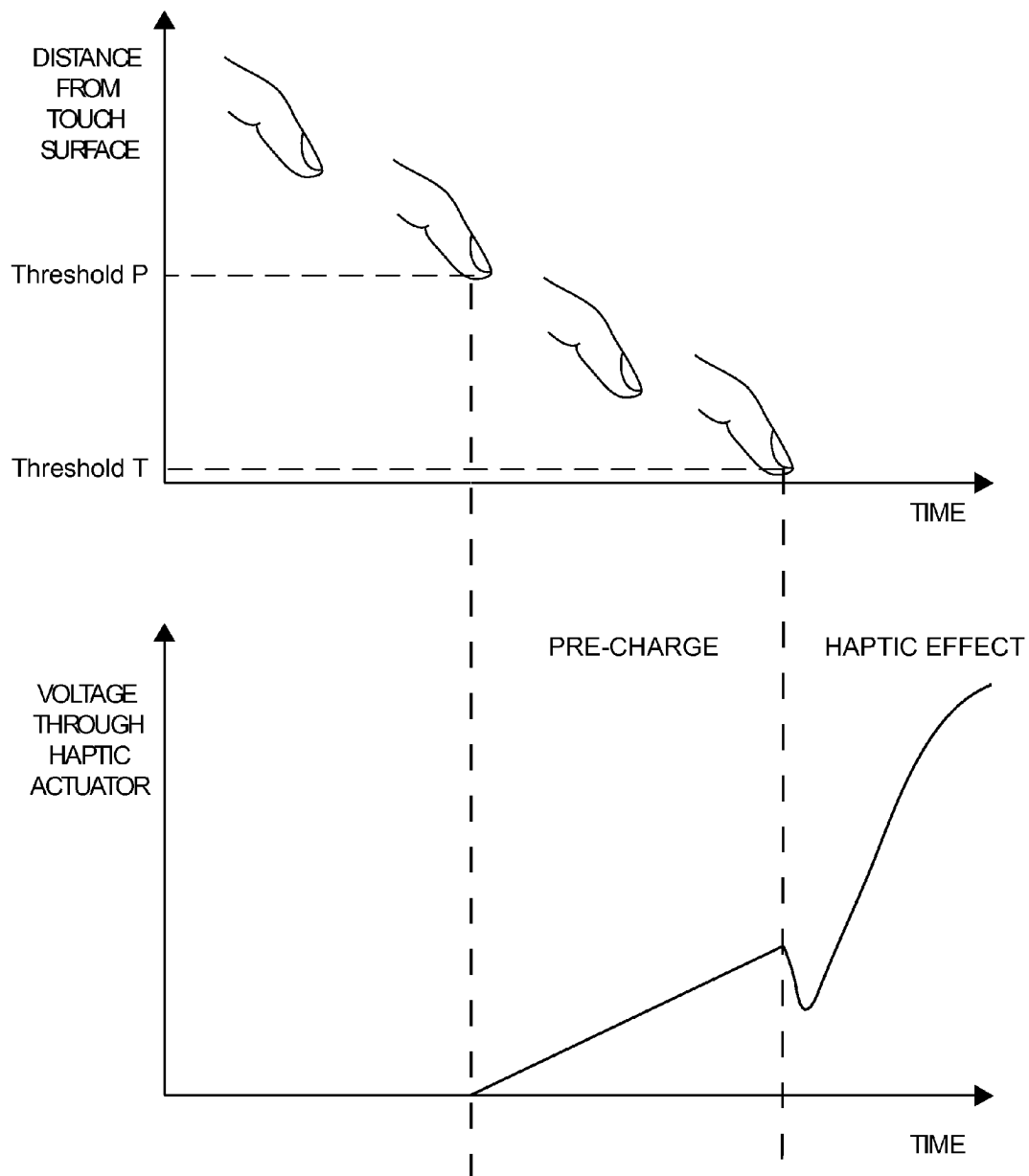


FIG. 4

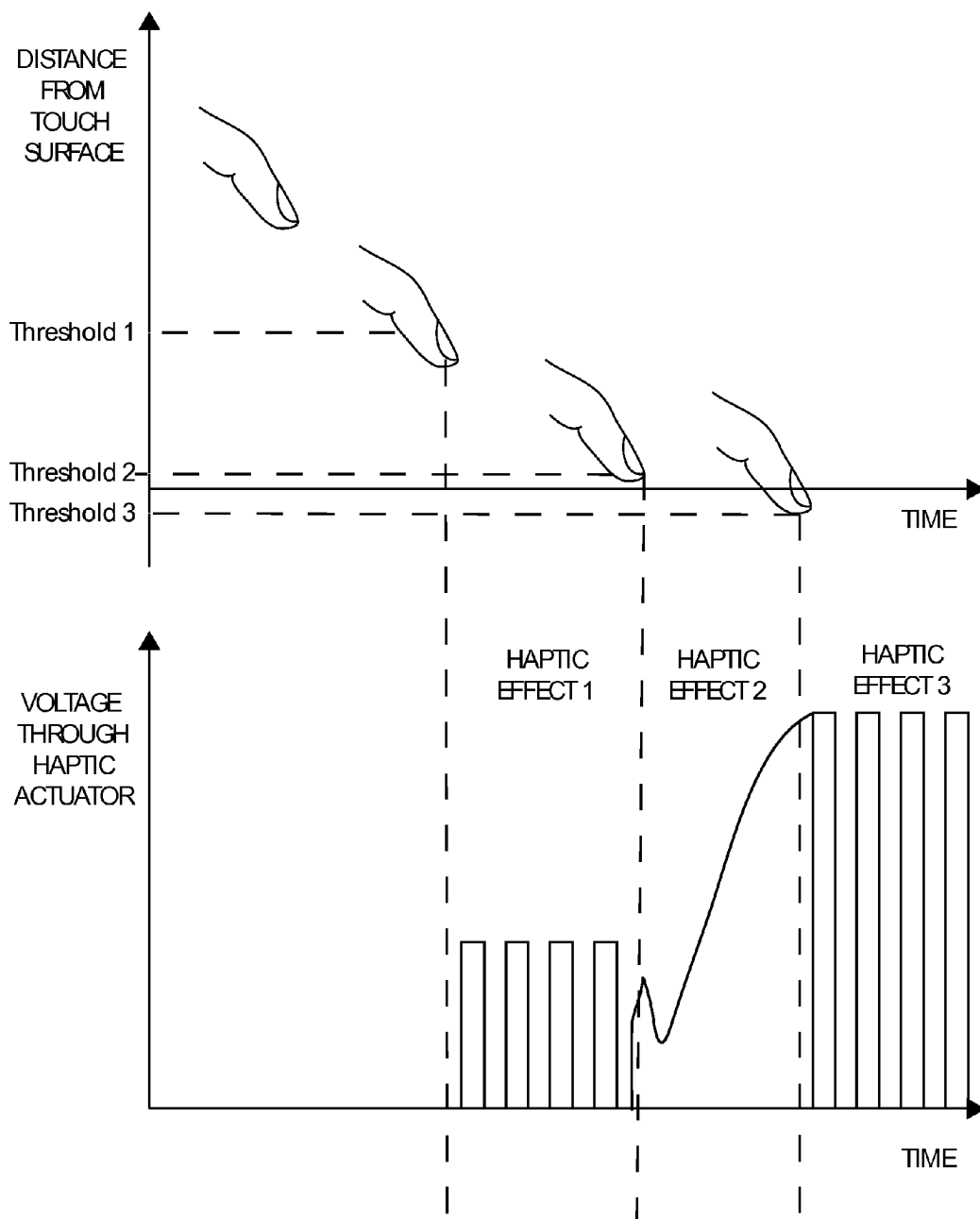
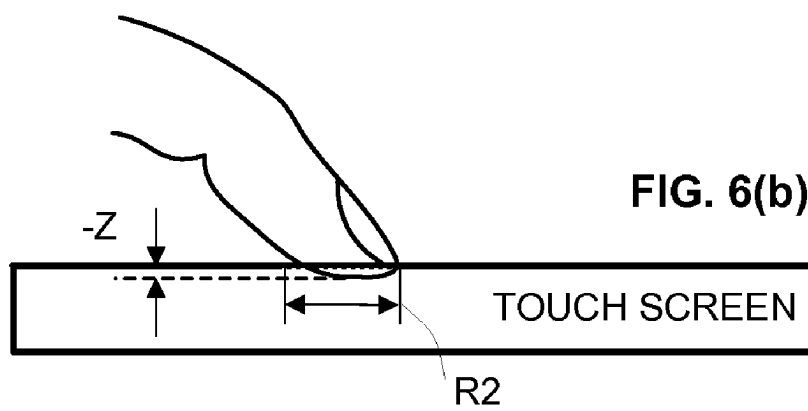
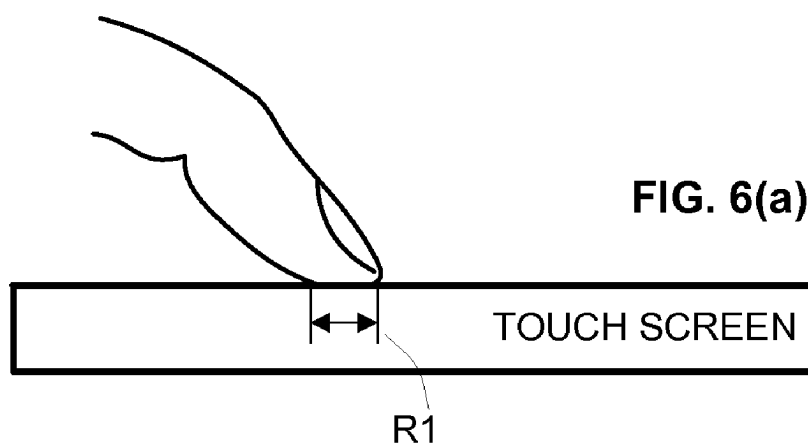


FIG. 5



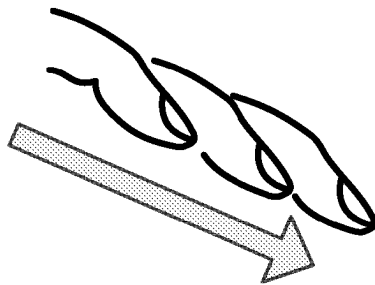


FIG. 7(a)

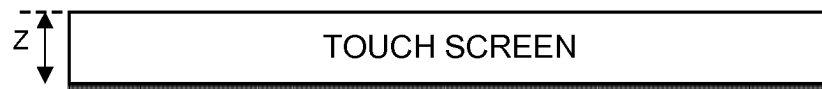


FIG. 7(b)



FIG. 7(c)



FIG. 7(d)



PROXIMITY AND FORCE DETECTION FOR HAPTIC EFFECT GENERATION

RELATED APPLICATIONS

[0001] This application claims priority to provisional U.S. Patent Application Ser. No. 61/470,764, entitled "Touch Screen and Haptic Control" filed on Apr. 1, 2011, the content of which is incorporated herein in its entirety.

BACKGROUND

[0002] The present invention relates to user interface control, in particular to haptic effect generation techniques based on proximity, touch, and/or force detection.

[0003] Haptics refers to the sense of touch. In electronic devices, haptics relates to providing a touch sensory feedback to the user. Electronic devices incorporating haptics may include cell phones, PDAs, gaming devices, etc. The user interacts with electronic devices through a user interface, such as a touch screen; however, the user often does not know if the user's desired function was recognized or is being performed by the electronic device. Thus, electronic devices generate a haptic feedback in the form of a vibro-tactile sensation (often, a simulated "click") to alert the user of the electronic device's performance. Stated differently, haptic feedback lets the user know what is going on with the electronic device. In a gaming electronic device, for example, haptics can provide a sensory stimuli according to game interactions.

[0004] For a user to accept haptics, the haptic response should follow closely in time with the user action. Thus, prolonged latency in the haptic response, which is the delay between the moment of user interaction and the corresponding haptics response, causes a disconnect between the touch and the haptic response. When the latency exceeds about 250 ms, the latency becomes noticeable to the user and it can be perceived as device error rather than an event that was triggered by the user's input. For example, a user may touch a first button on a touch screen and move onto another function of the device before feeling the haptic response to the first button. This temporal disconnect results in low user acceptance of haptics leading to a poor user experience.

[0005] Moreover, as electronic devices become more complex, user interaction with the device may expand to more than mere point touches on the screen. For example, user hovering his/her finger over a screen may constitute a type of user interaction or the force of the user touches may constitute different type of user interaction events depending on the amount of force. Thus, different haptic effects should complement these new types of user interaction events.

[0006] Therefore, the inventors recognized a need in the art for efficient haptic effect generation with reduced latency that complement different types of user interaction events.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] FIG. 1 is a simplified block diagram of a display device according to an embodiment of the present invention.

[0008] FIGS. 2(a)-(c) illustrate an integrated touch screen sensor grid according to an embodiment of the present invention.

[0009] FIGS. 3(a)-(b) illustrate a series of user interaction event detection according to an embodiment of the present invention.

[0010] FIG. 4 illustrates a haptic effect generation operation according to an embodiment of the present invention.

[0011] FIG. 5 illustrates a haptic effect generation operation according to an embodiment of the present invention.

[0012] FIGS. 6(a)-(b) illustrate a force detection operation according to an embodiment of the present invention.

[0013] FIGS. 7(a)-(d) illustrate a haptic bubble effect generation operation according to an embodiment of the present invention.

DETAILED DESCRIPTION

[0014] Embodiments of the present invention may provide a device including a haptic driver to drive a coupled actuator causing the actuator to generate a vibratory haptic effect. A touch screen may display a user interface and may include a sensor to detect user interaction with the touch screen within a predetermined range above the touch screen. A controller may calculate a proximity event based on the detected user interaction above the touch screen, and to control haptic driver operations according to the proximity event.

[0015] FIG. 1(a) is a simplified block diagram of a haptic-enabled display device 100 according to an embodiment of the present invention. The device 100 may include a user interface (UI) controller 110 with a processor 112 and a memory 114, a haptics driver 120, a haptics actuator 130, a touch screen 140 with a touch screen (TS) sensor 142, and a host system 150. The device 100 may be embodied as a consumer electronic device such as a cell phone, PDA, gaming device, etc.

[0016] Based on the TS sensor results, the UI controller 112 may calculate proximity, touch, and/or force user interaction events. Haptic generation, consequently, may be linked to these proximity, touch, and/or force events and thus may be significantly improved in terms of efficiency and precision. In an embodiment, latency may be improved by pre-charging the haptics actuator 130 based on detected proximity events such as location and/or rate of approach (i.e., velocity and/or acceleration). Therefore, a haptic effect may be generated faster upon an actual touch detected because of the pre-charged actuator. In another embodiment, haptic generation may be dynamically changed and/or adjusted based on detected proximity, touch, and/or force events.

[0017] The UI controller 110 may include the processor 112 and the memory 114. The processor 112 may control the operations of the UI controller 110 according to instructions stored in the memory 114. The memory 114 may also store haptic effect profiles associated with different feedback responses. Different user interaction events may be associated with different haptic effect profiles. The memory 114 may be provided as a non-volatile memory, a volatile memory such as random access memory (RAM), or a combination thereof.

[0018] The UI controller 110 may be coupled to the host system 150 of the device. The UI controller 110 may receive instructions from the host system 150. The host system 150 may include an operating system and application(s) that are being executed by the device 100. The host system 150 may represent processing resources for the remainder of the device and may include central processing units, memory for storage of instructions representing an operating system and/or applications, input/output devices such as display driver, audio drivers, user input keys and the like (not shown). The host system 150 may include program instructions to govern operations of the device and manage device resources on

behalf of various applications. The host system **150** may, for example, manage content of the display, providing icons and softkeys thereon to solicit user input thru the touch screen **140**. In an embodiment, the UI controller **110** may be integrated into the host system **150**.

[0019] The UI controller **110** may be coupled to the touch screen **140** and to the TS sensor **142** therein that measures different user interaction with the touch screen **140**. The touch screen **140** may also include an overlain display, which may be provided as a backlit LCD display with an LCD matrix, lenticular lenses, polarizers, etc.

[0020] FIG. 1(b) is a functional block diagram of the UI controller **110** according to an embodiment of the present invention. The processor **112** in the UI controller **110** may include a proximity classification module **112.1**, a touch classification module **112.2**, a force classification module **112.3** and a haptics response search module **112.4**. The memory **114** in the UI controller **110** may include haptics profiles data **114.1**. The data may be stored as look-up-tables (LUTs). The proximity classification module **112.1**, touch classification module **112.2**, and the force classification module **112.3** may receive the TS sensor data. Based on the TS sensor data, the classification module may calculate corresponding proximity, touch, and/or force event(s).

[0021] The proximity classification module **112.1** may calculate user proximity to a touch screen **140**, for example before contact is made, based on proximity associated TS sensor data. The proximity classification module **112.1** may calculate location (or locations for multi-touch user interactions) and time of the user movement (e.g., finger, stylus, pen, etc.) as it hovers over the touch screen **140**. The proximity classification module **112.1** may be complimentary to the type of touch screen **140** and TS sensor **142**. For example, if the touch screen **140** and the TS sensor **142** is provided as a capacitive touch screen and corresponding capacitive sensor (or grid of capacitive sensors), the proximity classification module **112.1** may calculate changes in respective capacitive fields for detecting proximity events. Further, the proximity classification module **112.1** may be programmed to differentiate between true positives for desired user proximity events and false positives for objects larger than a typical user interaction instrument (e.g., finger, pen, stylus).

[0022] The touch classification module **112.2** may calculate user touch(es) on the touch screen **140** and the touch(es) characteristics (e.g., icon selection, gesture, etc.). The touch classification module **112.2** may calculate the location (or locations for multi-touch user interactions) and time of the user touch. The touch classification module **112.2** may be complimentary to the type of touch screen **140** and TS sensor **142**. For example, if the touch screen **140** and the TS sensor **142** is provided as a capacitive touch screen and corresponding capacitive sensor (or grid of capacitive sensors), the touch classification module **112.2** may calculate changes in respective capacitive fields for detecting touch events. Further, the touch classification module **112.2** may be programmed to differentiate between true positives for desired user touch events and false positives for objects larger than a typical user interaction instrument (e.g., finger, pen, stylus).

[0023] The force classification module **112.3** may calculate an amount of force corresponding to a user touch on the touch screen **140**. The force classification module **112.3** may calculate how hard the user presses down and for how long with respect to a touch screen **140** contact. The force sensor **160** may be complimentary to the type of touch screen **170**. The

force classification module **112.3** may be complimentary to the type of touch screen **140** and TS sensor **142**. For example, if the touch screen **140** and the TS sensor **142** is provided as a capacitive touch screen and corresponding capacitive sensor (or grid of capacitive sensors), the force classification module **112.3** may calculate changes in respective capacitive fields for detecting force events. Further, the touch classification module **112.2** may be programmed to differentiate between true positives for desired user touch events and false positives for objects larger than a typical user interaction instrument (e.g., finger, pen, stylus).

[0024] The haptics response search module **112.4** may receive proximity, touch, and/or force events as calculated by the modules **112.1-112.3**, and may generate a haptic command based on the haptics profile data **114.1**. For example, the haptic response search module **112.4** may match the calculated proximity, touch, and/or force event data to a stored haptic profile and may generate a haptic command associated with the matched haptic profile.

[0025] Returning to FIG. 1(a), the UI controller **110** may receive the TS sensor results and may generate corresponding haptic commands accordingly based on stored haptic profiles. The UI controller **110** may be coupled to the haptic driver **120**. Based on the haptic command from the UI controller **110**, the haptic driver **120** may generate a corresponding drive signal. The drive signal may be an analog signal, and the drive signal may be a current or voltage signal.

[0026] The haptics driver **120** may be coupled to the haptics actuator **130**. The haptics actuator **120** may be embodied as piezoelectric elements, linear resonant actuators (LRAs), eccentric rotating mass actuators (ERMs), and/or other known actuator types. The haptics driver **120** may transmit the drive signal to the haptics actuator **130** causing it to vibrate according to the drive signal properties. The vibrations may be felt by the user providing a vibro-tactile sensory feedback stimuli.

[0027] In an embodiment, the haptics actuator **130** may include a mechanical system such as a motor that vibrates to generate the desired haptic effect. For example, the haptics actuator **130** may include a coil motor with a spring loaded mass and a permanent magnet. The coil motor may cause the spring loaded mass to vibrate to generate the haptic effect. The haptics actuator **130** may also include magnetic coils to generate the motion.

[0028] In an embodiment, a plurality of haptic actuators may be provided in the device to generate a plurality of haptic effects at different parts of the device. The haptic actuators may be driven by the haptic actuator **130** with the same drive signal with multiple drive signals.

[0029] FIG. 2 illustrates a simplified touch screen arrangement with capacitive sensors according to an embodiment of the present invention. FIG. 2(a) illustrates a capacitive sensor grid layout of the touch screen arrangement, FIG. 2(b) illustrates a cross-sectional view of the touch screen arrangement, and FIG. 2(c) illustrates capacitive fields of the capacitive sensor grid.

[0030] The touch screen arrangement may include a touch screen **210**, a plurality of capacitive sensors, and a cover **240**. The capacitive sensors **220** may be provided in a grid fashion that overlaps the display panel **230**. A cover **240** may protect the display panel. For example, the cover **240** may be provided as a glass cover.

[0031] The capacitive sensors **220** may be arranged in the grid with multiple columns and rows. The grid may include m

columns and n rows thus generating a $m \times n$ array (say, 11×15). The size of the array may be designed to accommodate different screen sizes and/or the desired accuracy/precision level of the touch screen. Cross points (CS) of the sensor grid may be placed a distance (D) apart from each other. In an embodiment, each cross point CS, for example, may be 5 mm apart from its neighboring cross points.

[0032] The capacitive sensors **220** may detect proximity events, touch events, and/or force events as will be described below. The array of capacitive sensors **220** may be scanned at a scanning frequency. The scanning frequency may be programmable. For example, the scanning frequency may be set to 100 or 120 Hz. In an embodiment, the scanning frequency, however, may be dynamically changed based on present conditions. For example, the scanning frequency may be dynamically changed based on a rate of approach as detected by the capacitive sensors **220** (e.g., $5 \times$ the rate of approach). Hence, the scanning frequency may increase as the rate of approach increases.

[0033] In a scan, each cross point CS (or each row or each column) may generate a bit code result, which may reflect a change from normal (i.e., without user presence) conditions with respect to proximity, touch, and/or force detection. For example, each CS may generate a 14 bit result. The code may be used to calculate the type, location, and/or other characteristics such as the rate of approach (velocity and/or acceleration), force, etc., of the user interaction.

[0034] For proximity detection, each CS may detect changes in its capacitive field as shown in FIG. 2(c). Further, a user's finger hovering over the touch screen may be sensed by multiple CSs. In the FIG. 2(c), CS **1.1** may detect a larger presence in its capacitive field as compared to its neighboring CS **1.2**. As a result, the code change of CS **1.1** may be higher than that of CS **1.2**. From the sensor results, data representing the X,Y,Z coordinates of the finger location may be generated. X,Y location may correspond to the location of the CS(s) in the grid that detected the presence (i.e., code change), and the Z location may correspond to the amount of change detected. Moreover, based on one or more sets of X,Y,Z coordinates, other characteristics such as the rate of approach may be calculated.

[0035] The capacitive sensors **220** may also detect location and time of actual touches. For touch detection, X,Y coordinates and the time of the touches may be generated based on the sensor results. In addition, other characteristics such as the type of touch (e.g., movement on the touch surface) may be calculated from one or more sets of scan results. Force detection by the capacitive sensors **220** may also be performed by the sensors as will be described below.

[0036] FIGS. 3(a)-(b) illustrate user interaction detection by the sensors. FIG. 3(a) illustrates a two-dimensional workspace **310** (i.e., UI map) without any user interaction in accordance with embodiments of the present invention. The workspace **310** is illustrated as including a plurality of icons **320** and buttons **330** that identify interactive elements of the workspace **310**. The workspace **310** may include other areas that are not designated as interactive. For example, icons **320** may be spaced apart from each other by a certain separation distance. Further, other areas of the display may be unoccupied by content or occupied with display data that is non-interactive. Thus, non-interactive areas of the device may be designated as "dead zones" (DZs) for purposes of user interaction (shown in gray in the example of FIG. 2(b)).

[0037] The sensors detection may become more localized as the user finger approaches and touches the screen. FIG. 3(b) illustrates the two-dimensional workspace **310** with a series of user interaction beginning from proximity detection to touch detection and then force detection. At time t_1 , the user's finger is a certain distance above the workspace **310** where workspace detection area for t_1 may be generalized to a significant left bottom corner. As the finger approaches, the detection may increase accuracy and may become more localized. At time t_2 , the user finger may be slightly above the button **330**. At time t_3 , the user finger may touch the screen at button **320**. And at time t_4 , the detection area may increase detecting a larger amount of force as compared to time t_3 touch. Based on the localization of the detection area, different haptic generation operations may be controlled and optimized.

[0038] FIG. 4 illustrates a pre-charging haptic generation operation according to an embodiment of the present invention. FIG. 4 includes two plots. The top plot shows a user finger approaching the touch surface in a distance versus time graph, and the bottom plot shows a corresponding voltage through the haptic actuator versus time graph. As the finger is approaching the touch surface, the device may be detecting the location of the finger via proximity sensing. The device, consequently, may be generating X,Y,Z coordinates based on the proximity results. Based on at least two sets of coordinates, the device may also calculate the rate of approach and/or the direction of the finger's movement. Hence, the device may anticipate the time and/or location of the touch.

[0039] At time t_1 , the device may detect the finger at a predetermined distance (Threshold P) from the touch surface. At this time t_1 , the device may initiate pre-charging the haptic actuator. The haptic actuator may be pre-charged according to a haptic profile for the anticipated time and location of the touch. At time t_2 , the device may detect the finger making contact with the touch surface via Threshold T. The device, consequently, may be generating X,Y,Z coordinates based on the touch results. At this time t_2 , the device may drive the haptic actuator with the a corresponding haptic effect voltage based on the haptic effect profile associated with the touch characteristics. Therefore, the device may generate the haptic effect faster upon touch screen contact because of pre-charging the haptic generating components, and thereby reducing latency between the user touching the screen and feeling the corresponding haptic feedback.

[0040] The Threshold P value may be programmable. In an embodiment, the Threshold P value may be dynamically adjustable based on finger movement characteristics. For example, the threshold P value may be directly proportional to the rate of approach. Hence, as the rate of approach increases, the Threshold P value increases and vice versa. As a result, the pre-charging time may be maintained independent of the rate of approach to allow sufficient time for pre-charging the haptic actuator to the desired voltage level.

[0041] In an embodiment, haptic selection may also be based on sensor measurements. For example, haptic effect types may be selected based on the rate of approach of the user's finger as it moves toward a touch screen—a first haptic effect may be selected in response to a relatively "fast" velocity and a second haptic effect may be selected in response to a relatively "slow" velocity.

[0042] In an embodiment of the present invention, different types of haptic events may be selected based in part on proximity, touch, and/or force events. For example, a set of dif-

ferent haptic effects may be generated based on different measured events such as the rate of approach, direction, location, force, etc. FIG. 5 illustrates a multi-haptic effect generation operation according to an embodiment of the present invention. FIG. 5 includes two plots. The top plot shows a user finger approaching the touch surface in a distance versus time graph, and the bottom plot shows a corresponding voltage through the haptic actuator versus time graph. FIG. 5 illustrates different haptic effect generation based on different user interaction events as detected by the sensor(s) via thresholds.

[0043] At time t_1 , the device may detect the finger at a predetermined distance, Threshold 1, from the touch surface. The device, consequently, may be generating X,Y,Z coordinates based on the proximity sensor results. At this time t_1 , the device may drive a haptic actuator to generate a first haptic effect according to a haptic profile associated with the finger location and/or movement characteristics (e.g., rate of approach).

[0044] At time t_2 , the device may detect the finger touching the touch surface with Threshold 2. The device, consequently, may be generating X,Y coordinates based on the touch sensor results. At this time t_2 , the device may drive the haptic actuator to generate a second haptic effect according to a haptic profile associated with the touch location and/or movement characteristics (e.g., type of contact).

[0045] At time t_3 , the device may detect the force of the finger contact crossing a predetermined level with Threshold 3. The device, consequently, may be generating X,Y,Z coordinates based on the force sensor results. At this time t_3 , the device may drive the haptic actuator to generate a third haptic effect according to a haptic profile for the finger touch location and/or movement characteristics (e.g., amount of force). The third haptic effect, for example, may be an alert to the user that he/she is pressing too hard on the touch screen. The same actuator or different actuators may be used to generate the first, second, and/or third haptic effects.

[0046] In an embodiment, the haptic effect selection for different interaction events such as proximity, touch, and/or force events may be dynamically changed based on user interaction history. For example, in a text entry application, different users may enter text at different rates. If a user touches a first letter and the device initiates a haptic effect, then the user's moves toward another letter, the device may recognize the approaching finger and terminate the first haptic effect sufficiently early before the second letter is touched so as to minimize blur between successive haptic effects.

[0047] FIG. 6 illustrates operation of a haptics enabled device to detect force applied to a touch screen for use in accordance with embodiments of the present invention. As illustrated in FIG. 6(a), a user may press the touch screen lightly with his/her finger. In this case, there may be a small deflection of the user's finger at the point of contact, which may be registered on the touch screen as an area of contact. In FIG. 6(a), the area may be considered as a circle having radius R. A touch sensor may derive a force at the point of contact from the calculated area. In FIG. 6(b), the user presses the touch screen with greater force, causing a greater amount of deformation in the user's finger. The user's finger, therefore, may register a greater area of contact than in the FIG. 6(a) case, which the touch sensor may use to derive a corresponding higher value of force.

[0048] In an embodiment, the force sensor may represent force as a distance value in the Z plane. The force sensor may

calculate an area of contact between the user and the touch screen and convert the value to a distance value in the Z plane. If, in the proximity and presence detection operations, distance values are represented as positive Z values, distance values representing user force may be represented as negative Z values. See, FIG. 6(b). In the example of FIG. 6(b), the negative Z value models a hypothetical depth of an operator's touch, based on deformation of the user's finger, rather than an actual depth of touch.

[0049] In an embodiment, haptic effects may be pre-charged and driven before the user touch based on proximity detection. The device, for example, may generate a "bubble" effect, which may correspond to stimulating a clicking functionality using haptic effects. FIGS. 4(a)-4(d) illustrate a bubble effect generation operation according to an embodiment of the present invention.

[0050] FIG. 7(a) illustrates a state of the touch screen prior to detection. As the user's finger approaches the touch screen, it enters the "field of view" of the touch screen and is identified by the proximity sensor. In response, a haptics driver may pre-charge a haptics actuator to cause the touch screen to deflect toward the user's finger by a predetermined amount, shown as ΔZ in FIGS. 7(a)-(d). Thus, the touch screen may be deflected toward the user's finger by the time it makes contact with the touch screen as shown in FIG. 7(b). When the touch sensor determines that the user's finger has made contact, it may initiate the haptic effect. As shown in FIG. 7(c), a mechanical button click may be simulated, for example, by removing the pre-charge effect and inducing the touch screen to return to its default level (shown as "Z"). After the retraction of the touch screen, the user's finger may fall to the surface of the screen at the Z level, shown in FIG. 7(d). The click effect may be induced by the user feeling mechanical resistance at the first point of contact with the screen deflected forward (FIG. 7(b)) and then at the second point of contact with the screen at the rest position (FIG. 7(d)). This effect simulates a mechanical compression effect (i.e., the bubble effect).

[0051] Of course, the proximity-based deflection operations are not limited to click effects. Vibration effects may be induced by deflecting the screen forward prior to initial contact, then oscillating the screen forward and backward after contact is made. A variety of different haptic effects may be used in connection with proximity detection operations.

[0052] The foregoing description refers to finger touches for illustration purposes only, and it should be understood that embodiments of the present invention are applicable for other types of user interaction such as with a pen, stylus, etc.

[0053] Those skilled in the art may appreciate from the foregoing description that the present invention may be implemented in a variety of forms, and that the various embodiments may be implemented alone or in combination. Therefore, while the embodiments of the present invention have been described in connection with particular examples thereof, the true scope of the embodiments and/or methods of the present invention should not be so limited since other modifications will become apparent to the skilled practitioner upon a study of the drawings, specification, and following claims.

[0054] Various embodiments may be implemented using hardware elements, software elements, or a combination of both. Examples of hardware elements may include processors, microprocessors, circuits, circuit elements (e.g., transistors, resistors, capacitors, inductors, and so forth), integrated

circuits, application specific integrated circuits (ASIC), programmable logic devices (PLD), digital signal processors (DSP), field programmable gate array (FPGA), logic gates, registers, semiconductor device, chips, microchips, chip sets, and so forth. Examples of software may include software components, programs, applications, computer programs, application programs, system programs, machine programs, operating system software, middleware, firmware, software modules, routines, subroutines, functions, methods, procedures, software interfaces, application program interfaces (API), instruction sets, computing code, computer code, code segments, computer code segments, words, values, symbols, or any combination thereof. Determining whether an embodiment is implemented using hardware elements and/or software elements may vary in accordance with any number of factors, such as desired computational rate, power levels, heat tolerances, processing cycle budget, input data rates, output data rates, memory resources, data bus speeds and other design or performance constraints.

[0055] Some embodiments may be implemented, for example, using a computer-readable medium or article which may store an instruction or a set of instructions that, if executed by a machine, may cause the machine to perform a method and/or operations in accordance with the embodiments. Such a machine may include, for example, any suitable processing platform, computing platform, computing device, processing device, computing system, processing system, computer, processor, or the like, and may be implemented using any suitable combination of hardware and/or software. The computer-readable medium or article may include, for example, any suitable type of memory unit, memory device, memory article, memory medium, storage device, storage article, storage medium and/or storage unit, for example, memory, removable or non-removable media, erasable or non-erasable media, writeable or re-writable media, digital or analog media, hard disk, floppy disk, Compact Disc Read Only Memory (CD-ROM), Compact Disc Recordable (CD-R), Compact Disc Rewritable (CD-RW), optical disk, magnetic media, magneto-optical media, removable memory cards or disks, various types of Digital Versatile Disc (DVD), a tape, a cassette, or the like. The instructions may include any suitable type of code, such as source code, compiled code, interpreted code, executable code, static code, dynamic code, encrypted code, and the like, implemented using any suitable high-level, low-level, object-oriented, visual, compiled and/or interpreted programming language.

We claim:

1. A device, comprising:
 - a haptic driver to drive a coupled actuator causing the actuator to generate a vibratory haptic effect;
 - a touch screen to display a user interface, wherein touch screen including a sensor to detect user interaction with the touch screen within a predetermined range above the touch screen; and
 - a controller to calculate a proximity event based on the detected user interaction above the touch screen, and to control haptic driver operations according to the proximity event.
2. The device of claim 1, wherein the proximity event includes a rate of approach.
3. The device of claim 1, wherein the sensor comprises a capacitive sensor grid that is scanned at a scanning frequency rate.

4. The device of claim 3, wherein the scanning frequency rate is dynamically adjusted based on prior calculated user interaction properties.

5. The device of claim 1, wherein the controller is configured to pre-charge the haptic driver to a voltage level based on the proximity event.

6. The device of claim 1, wherein the controller is further configured to calculate touch and force events.

7. The device of claim 6, wherein the device is configured to generate multiple haptic effects based different proximity, touch, and/or force events.

8. The device of claim 6, wherein force events are detected based on an area of user touch on the touch screen.

9. A method of generating haptic effects, comprising: detecting a user interaction above a touch surface within a predetermined range;

calculating location coordinates of the user interaction; calculating user interaction properties based on the location coordinates;

applying voltage through a haptic actuator based on the user interaction properties.

10. The method of claim 9, further comprises:

pre-charging the actuator to a first voltage level based on the user interaction properties;

detecting a user touch on the touch screen;

calculating touch location coordinates of the user touch;

driving the haptic actuator to generate a haptic effect based on the user touch from the first voltage level.

11. The method of claim 10, further comprises:

detecting an amount of force of the user touch;

driving the haptic actuator to generate a second haptic effect based on the amount of force.

12. The method of claim 11, wherein detecting the amount of force is proportional to an area of the user touch on the touch screen.

13. The method of claim 9, wherein the user interaction properties include a rate of approach.

14. The method of claim 9, wherein the detecting is performed by scan reads of a capacitive sensor grid at a scanning frequency.

15. The method of claim 14, wherein the scanning frequency rate is dynamically adjusted based on prior calculated user interaction properties.

16. A user interface controller, comprising:

a sensor input to receive sensor data related to user interaction above a touch screen within a predetermined range;

a memory to store program instructions and a plurality of haptic profiles;

a processor to calculate user interaction properties from the sensor data, to match a haptic profile from the memory to the user interaction properties, and to generate a haptic command associated with the haptic profile; and a

haptic driver output to send the haptic command.

17. The user interface controller of claim 16, wherein the user interaction properties include a rate of approach.

18. The user interface controller of claim 16, wherein the haptic command includes an instruction to pre-charge an actuator.

19. The user interface controller of claim 16, wherein the sensor data also includes also relates to a user touch on the touch screen and to an amount of force of the user touch.

20. The user interface controller of claim 19, wherein the processor is configured to generate multiple haptic commands based the sensor data.